SESSION

AGENT TECHNOLOGIES AND APPLICATIONS

Chair(s)

TBA

An Autonomous Agent with an Evolved, Affect-based Decision System

Paul G. Joseph[1]**,Haim Levkowitz**[1]

[1]Department of Computer Science, University of Massachusetts, Lowell, MA, USA

Abstract - *We investigate the modeling of affect with the goal to create an agent that behaves "emotionally" closer to humans. Our previous work showed evidence that modeling the fundamental emotions of the mammalian brain offered the basis of a flexible decision system for an autonomous agent. Earlier work hard-coded the parameters used to model affect; now, we let the parameters evolve instead. Agents learn dynamically from interactions with their environment and use this learning in a speedy, general-purpose decision system. The agent's interactions with its environment and consequent impact on its well-being promise an affect-based approach to semantics. We subject an agent to a "Turing-like test for affect" in which it has to pass a hostile predator to reach its goal. We compare our approach with others; the strict basis of our approach is mainstream neuroscience. Agents independently assign affect valuations that provide a basis for semantic understandings.*

Keywords: *autonomous agent, affect, decision system, mammalian brain, semantics*

1. Introduction

This paper describes continued work in a new kind of decision system—one based on emotions or "affect" and whose implementation is directly modeled on and constrained by those systems in the mammalian brain that are presently thought to handle affect. The underlying premise is that humans (and all mammals) emotionally evaluate actions in terms of how these will influence their well-being and that consequently, being able to reproduce these affect-based evaluations in autonomous agents will provide for an additional method of making and/or supporting decisions. We intentionally constrain ourselves to the underlying empirical evidence from mainstream neuroscience and adopt the "philosophical stance" that when presented with two approaches, we limit ourselves to that which has a sounder basis in the underlying biological empirical evidence.

The approach offers the promise of advancing understanding and capability in three areas: autonomous decision-making, semantics, and affect modeling. Using a search heuristic in which the external environment modifies internal affect with the aim of maximizing internal positive affect provides the basis of an autonomous decision engine, which, because of its explicitly non-cognitive bias, is fast enough for use in real-time applications. The affect generated in the agent by interacting with objects in its environment in terms of the impact of these on the agent's well-being, promises an affect-based approach to semantics. The modeling of affect will also advance understanding of how the mammalian brain handles affect and makes affect-based decisions, thus advancing research in cognitive and neuroscience. In addition, modeling emotions in a single agent and in groups of agents will be useful to researchers in other fields including ethology and social sciences..

Previous work [18, 19, 20] identified the fundamental emotions as understood by the neuroscientist Dr. Jaak Panksepp and created a decision system based on these fundamental emotions. The approach showed promise—creating a fast decision system that seemed to be flexible and useful to an autonomous, intelligent agent. However, the parameters of this system were "hard coded" with no real justification for their values other than that trial and error showed that they worked. In this paper, we allow the autonomous agent to evolve its parameters based on its interaction with its environment. We then study the resulting system with a "Turing-like test for affect."

2. Review of previous work

Previous work [18, 19, 20], proposed that a major part of decision-making involves emotion-based evaluations of objects and ideas. We proposed an architecture that follows closely findings in neuroscience. Our goal is to articulate and validate the concept that non-conscious software agents can develop a sense of value of the objects with which they interact. This evaluation is in terms of how these objects influence the agent's well-being, i.e., its ability to function effectively. A key measure of well-being is the agent's affective state.

Such an approach promises the ability to create systems that can interact with real-world objects with some understanding of what these objects mean that is over and above analytical understandings typical of standard computing. The promise is that we can then use this extended understanding to demonstrate behavior and adaptability typically seen in humans and animals. These behaviors include rapid decision capabilities with autonomously driven goal setting. Such an affect-based approach also offers promise in providing a new approach to semantics and other problems, including, eventually, even the ability to make analogies and be creative [19].

The basis of our earlier work [18] was the neuroscience model of affect by Dr. Jaak Panksepp. Panksepp [30] postulates the existence of seven core (fundamental) emotional systems that map directly to seven physical neuronal systems that he had identified in the mammalian brain which "…generate well-organized emotion sequences that can be evoked by localized electrical stimulation of the brain…". He called these systems the SEEKING, RAGE, FEAR, PANIC, LUST, CARE, and PLAY systems, with the latter three qualified as serving socioemotional needs. Panksepp distinguishes his emotions from "basic drives" such as for example "hunger." Per Pank-

sepp, basic drives form the physical/bodily substrate that forms the basis of emotions.

Following from the theory of core/fundamental emotions, any emotion is some combination of the core or fundamental emotions. For example, the fear felt by a teenager on his first date would be viewed as some combination of (perhaps) PANIC, CARE, PLAY, SEEK, and LUST; resentment or indignation as combinations of RAGE, CARE, SEEK. The SEEK response in particular is associated with a large release of dopamine and a sense of well-being and is the key driver of emotions like "joy", "satisfaction", "contentment", "enjoyment", etc., variations of the theme of gratification of the SEEKING system and the attainment and benefit of the goal sought.

Initial work implemented the Panksepp model in the Pac-Man game [11] with Pac-Man coded using Python and with a heuristic to make it behave as a utility reflex agent [37] that maximizes its overall wellbeing. The heuristic was a simple dynamical systems model that used the Panksepp variables, with Pac-Man conveniently modeling prey and the Ghost, a predator. The results were promising and justified further research.

We then did a detailed review of other approaches to emotions [2, 3, 15, 23] and meta-analyses on affect processing in the mammalian brain [21, 24, 35]. Greene [15] describes his theory that the human brain has two modes of responding to situations. One, a set of efficient automatic responses driven by emotions, the other a manual mode used in response to non-standard situations that involves significant cognitive evaluation. LeDoux [23] posits that the former typically operates in timeframes of about 10 ms, while the latter in timeframes closer to 500 ms. Consequently, we use the 10 ms number as a heuristic: our explicitly non-cognitive model should takes no more than about 10 ms to compute a decision on an ordinary single CPU laptop. Eventually, the brain evaluates decisions from this layer in conjunction with those from an independent analytical layer in the ventromedial-prefrontal-cortex through a process not fully understood [15].

In our review of the mainstream neuroscience literature on emotions, we found no support for appraisal theories of emotions. The basis of cognitive appraisal theory is the James-Lange theory of emotions. It holds that we extract emotions primarily from our cognitive appraisals of events, and that emotions follow from the actions and feelings that result from these appraisals. Mainstream neuroscience no longer considers the James-Lange theory and cognitive appraisal theory to be valid [21, 35]. Today's mainstream neuroscience models are premised on the understanding that the most important contributions to affect come from the autonomic nervous system and that affect is largely facilitated by the limbic system [21, 24, 35]. Panksepp's model is explicitly non-cognitive.

Based on our review and meta-analyses we decided to focus initially on that subset of Panksepp's emotions that appear to have general acceptance in mainstream neuroscience. Specifically, these are SEEKING, FEAR, and RAGE. We did not implement PANIC as a fundamental emotion as it appears that the meta-analyses view it more as based on FEAR, rather than an emotion in its own right. We also did not implement any of

Panksepp's "more sophisticated, special purpose, socio-emotional systems," i.e., PLAY, LUST, and CARE.

Every module of the brain is made up of neurons whose activations are known to usually activate exponentially (see for example Armony et al. 1997, Greenfield, S. 2000). This motivated us to use exponential relationships in this initial attempt at modeling the different neural systems. These equations are tentative; "place-holders" until research from the neurosciences provide us with the basis for more accurate relationships. Nonetheless, it appears that even with these heuristic approximations, a useful decision system with seemingly realistic behavior is possible.

As a first approximation, we based the equation we used to model FEAR, on the distance to the threat. If the Manhattan distance to the threat is greater than some threshold value, then the prey takes no evasive action (shows no FEAR). Let this threshold distance be X_F. However once this threshold is crossed, then the prey exhibits FEAR behavior (avoidance), and we need a function to model this FEAR. Following from our earlier discussion on neurons activating exponentially, we choose an exponential function as an initial, approximate model for FEAR—Pac-Man's FEAR increases from zero at distance [greater than or equal to] X_F from a ghost, to infinity at a distance of zero from the ghost, as follows:

$$FEAR = e^{(\frac{X_F}{X})} \text{ for } X < X_F \text{, otherwise } FEAR = 0 \quad (1)$$

Likewise, we use a similar approach for SEEKING. Starvation studies (Owen and Hanson 2004) indicate that humans experience hunger pangs a few hours after a meal. These continue for the first three days without food, at which point, the body switches to start consuming body fat. Once this happens, the average human feels no further feelings of hunger for approximately five weeks. After this, with body fat consumed, the system starts consuming muscle. The person then experiences extreme hunger that signals that they must find food or else, their body will start consuming body muscle and brain matter to survive. Rather than model this complex behavior in detail, we again chose as a first approximation an approach similar to that used with the FEAR dimension, except time based. If the prey last consumed food at time T_S, then the $SEEKING$ value for Pac-Man goes up exponentially for values of time T greater than T_S, until $T - T_S$ reaches another value T_D, at which time the prey dies from starvation. Likewise, we assumed that $SEEKING$ varies exponentially in proportion X_D of the inverse distance X to the food, as:

$$SEEKING = e^{(\frac{T_D}{T_D - (T - T_S)_f})} * e^{(\frac{X_D}{X})} \quad (2)$$

As before, we used Panksepp's description of the RAGE system to model RAGE as FEAR multiplied by a function for entrapment. In the absence of more explicit rules, we simply write RAGE as a function of FEAR and entrapment, with entrapment being a simple summation of the minimum distance of

the prey to boundaries in the x and y directions (B_X, B_Y respectively), and coming into play only when this value is below some threshold value X_E, i.e.:

$$RAGE = FEAR * entrapment$$

$$entrapment = e^{\frac{X_E}{(\min(B_X) + \min(B_Y))}} \, for < X_E \quad (3)$$

The prey's affective state is hence contained in the dimensions of FEAR, SEEKING, RAGE and time, i.e., its affective evaluation of the predator and its food, are represented by the values of RAGE, FEAR, and SEEKING that result from its interacting with them and its own need for survival and well-being. At any given time, these values represent the prey's semantic understanding of the predator. The overall affective state of the prey at any given time is the net of the positive affect of SEEKING and the negative affects of FEAR and RAGE:

$$PREY_{overall\_affect} = SEEKING - FEAR - RAGE \quad (4)$$

This research (Joseph and Levkowitz 2012a) made a convincing case that adaptive behavior was possible with just the simpler three emotions. Once we understand these better, we plan to incorporate the more complex emotions not presently considered. Hence, we view this model only as the start of a program that will grow to include the more sophisticated emotions as knowledge of these become more available.

In this earlier work though, the values of T_D, X_F, X_D, X_E in the equations were "hard-coded." Since the results with this hard coding were promising, we felt justified in going on to the next stage and evolving these parameter values.

3. Evolving the parameters of the equation

The goal of Evolutionary Programming (EP) is to model evolution with learning. Given our underlying philosophy of constraining ourselves to empirically evidenced biology based approaches, EP appeared to be the natural choice to use over other rational optimization methods in order to determine optimal affect values with which to replace the simple hard-coded values used in our previous work. There are several approaches to evolutionary programming (see for example [34]) and we used the approach known as "mutation." Since the focus of this research was not evolutionary programming, we simply used a standard approach known as "meta-EP," which uses self-adaptation of the parameters with a real-valued representation.

Mutation changes a chromosome represented as $< x_1,...,x_n, \sigma_1,...,\sigma_n >$ into $< x_1',...,x_n', \sigma_1',...,\sigma_n' >$ where:

$$\sigma_i' = \sigma_i(1 + \alpha.N(0,1))$$
$$x_i' = \bar{x}_i + \sigma_i'.N(0,1) \quad (5)$$

and where $N(0,1)$ is the result of a random selection from a Gaussian distribution with a zero mean and a standard deviation of 1. In our case, each instance of the autonomous agent was treated as a chromosome, with each of the affect parameters T_D, X_F, X_D, X_E corresponding to the x_i, and the corre-

sponding standard deviation σ_i and mean $\bar{x}_i$ for T_D, X_F, X_D, X_E calculated from the population of surviving autonomous agents. As is typically the case for mutation (Poli, Langdon, and McPhee 2008.), we used $\alpha = 0.2$. We also enforced a standard requirement that:

$x_i < \varepsilon_0 \Rightarrow x_i' := \varepsilon_0$ where, typically, $\varepsilon_0 = 1$.
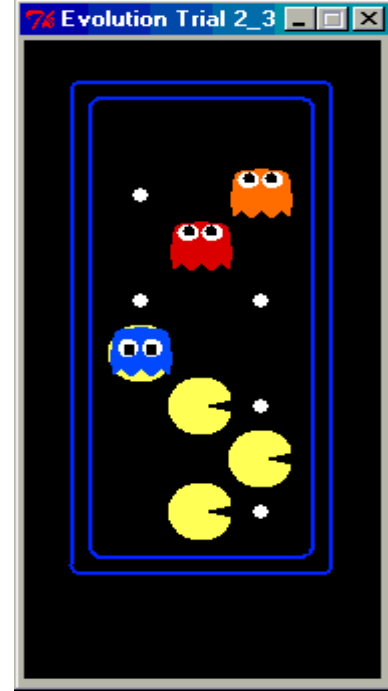

Figure 1: Setup used to evolve agent parameters

The general concept was to start with a certain number of autonomous agents (prey) and predators. When a predator "killed" an instance of prey, we regenerated a replacement autonomous agent according to the mutation scheme above. We tried various configurations of environment and numbers of autonomous agents and predators, starting with some initial "seed values" for the affect parameters. Figure 1 shows one such typical environment—the yellow dots are the autonomous agents (prey), the other colors are the predators.

Two well-known issues that can arise in this type of action-selection problem are "dithering" and choices that take the agent into local minima. In dithering, the prey moves back and forth between the same two positions, and in local minima, prey become "paralyzed" even if the predator is no where near. We used two approaches to minimize either possibility. First, in this version of the model, the predator has a degree of randomness built into it each time it chooses the direction of the next step to take. Second, when presented with two equally optimal alternatives, the prey randomly chooses between either. Together these two approaches seemed sufficient to guard against both dithering and local minima. Further, none of the "tracks" used by the prey for any given run ever fully matched that from any other run. In future work we plan to provide the prey with the ability to actively target and hunt down the prey with no random behavior.

Results were as expected from standard concepts of evolutionary programming: affect parameters changed and evolved depending on the number of predators and prey, different space and food configurations of the environment, and with differing degrees of rates of convergence and stability, and depending on the population size. In short, as expected, we learned nothing new about evolution *per se.* Rather, we found that the agents were able to use this scheme to adapt to their environment, as was to be expected. The basic finding was that EP is a useful technique to use, allowing autonomous agents to learn and adapt to their environment. We found that average computing time on a low end 2.3GHz Intel CPU for 1000 runs of the model was 7.08 seconds, in line with our LeDoux based heuristic of 10 ms., i.e., the model is fast enough for use in real-time applications.
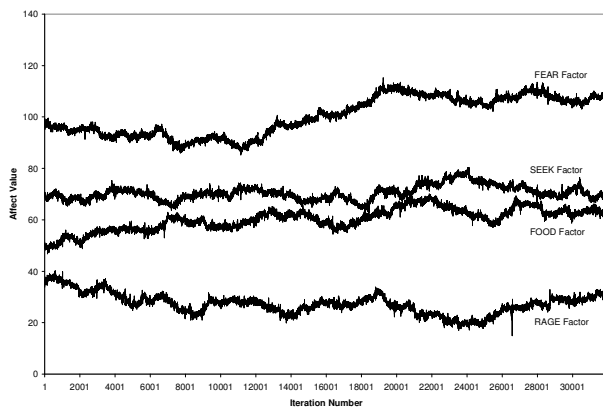

Figure 2: Stability of evolved affect parameters

Figure 2 shows typical evolution of the parameters, in this case, for the example in Figure 1. Here, initial values chosen based on the results of previous EP experiments showed stability. As shown, parameters did not significantly change after about 30,000 iterations, i.e., the system had stabilized.

4. A "Turing-like test" for affect

The well-known Turing test [41] opens with the words: "I propose to consider the question, 'Can machines think?'" Turing goes on to say that, "thinking" is difficult to define and so chooses to "replace the question by another, which is closely related to it and is expressed in relatively unambiguous words." His question was, "Are there imaginable digital computers which would do well in the imitation game?" Turing felt that it was possible to answer this question with an empirical test—the now famous "Turing Test."

Likewise, the question for our purposes is not whether machines can feel emotions but rather, whether they behave as if they feel emotions and are able to use these behaviors to survive. The mechanistic approach presented here via the equations implies that "consciousness" or "subjective feels" are not required to leverage the ability of emotions to form the basis of a decision system such as the one described above. Further, following Turing's approach of comparing empirical behavior of the system to real life behavior, in theory the model should allow for non-intuitive behavior observed in real life by ethologists—that of a prey animal moving towards a predator if its af-

fect state so disposes it. In other words, a prey is observed to move towards the predator if the prey's SEEK value was high enough to override its FEAR and RAGE, and there was food in the vicinity of the predator.
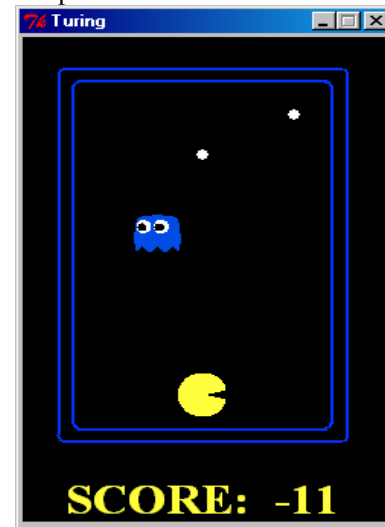

Figure 3: Setup for a "Turing test for affect"

Figure 3 shows a setup we used to test the ability of our affective systems model to demonstrate this kind of behavior. We programmed a predator (blue) to march only horizontally, choosing at random whether to go to the left or to the right, and thereby creating a guarded boundary. To pass through this guarded boundary, the prey must expose itself to considerable danger. We programmed the prey (yellow) with the emotional system. All the food is on the side of the predator away from the prey and the prey has to cross the predator's line of marching to get to the food.

Initially, we see that the prey stays as far away from the predator as per its value of FEAR. However, as the time since it has last eaten increases, its SEEK value goes up and it begins to move to the food, despite the presence of the predator. As the predator comes closer to the food, the SEEK value also goes up and directs it towards the food. At some point the SEEK overrides the FEAR and the prey moves towards and crosses the line of action of the predator. The position of the second piece of food is in the corner: SEEK must exceed RAGE (a function of closeness to enclosing walls) before the prey will consume this food.

We found that the values of the parameters as evolved in the environment of (say) Figure 1 were far from suitable for such a task. The prey was never able to summon up enough of SEEK to overcome its FEAR. We had to evolve these values specifically for the Turing test environment before we saw the expected outcome of the prey overriding its FEAR, making it to its food, and thereby surviving. The evolved value of FEAR for this more aggressive situation where Pac-Man had to pass the ghost was about 25 times less than in the previous case. RAGE (a measure of the prey being trapped by the boundaries) was about five times smaller, signifying that Pac-Man had evolved more aggressive behavior and so less likely to not approach Pac-Man or to stay away from the boundaries of its environment. Likewise, the evolved values of the prey indicated

as much as an order of magnitude reduction in its SEEK factor, and a five fold reduction in its FOOD factor, both reductions indicating a hardier profile, with reduced dependence on food.

This is as expected—parameter values evolved in one particular environment must be different from those evolved in a significantly different (more hostile) environment. The conclusion was that the evolutionary process did work as expected; the model could evolve to exhibit the wide range of behavior found in nature: a very fearful prey, or a hardier, less fearful prey, depending on the environment it evolved in.

The predator uses a random algorithm to determine the direction (left or right) of its next step. Consequently, while for any given iteration number, the specific values from the model across many thousands of runs are different, the general pattern of affect is the same for each run.

For the initial part of the run the prey remained where FEAR and RAGE were lowest. As time since eating increased, SEEK increased; the prey began to explore wider regions of its domain despite increases in both RAGE and FEAR. Finally, as time (SEEK) increased, the total of SEEK exceeded the sum of RAGE and FEAR and the prey crossed the boundary "patrolled" by the predator, and despite increasing RAGE, approached the boundary, and ate the food. Figure 4 shows this switch in NET affect for one "Turing test" in which there were at least 10 reversals of affect (and hence direction) before Pac-Man crossed the predator.

Heider and Simmel (1944) studied the reactions of human observers watching animations of moving triangles and spheres. These observers invariably projected life like behavior onto these abstract moving objects, concocting elaborate stories, including romantic interactions, between triangles and squares. Consequently, rather than rely on subjective interpretations by human observers to judge the realism of the predator/prey behavior we chose to determine this with more objective criteria, that is available for this kind of a test—specifically, whether Pac-Man demonstrated behaviors typically observed in predator-prey interactions (see [8] for concept). Specifically, these were 1) the actual survival rate, i.e., the measure of the overall success of the prey behavior 2) retreat/advance in the face of the prey 3) lateral avoidance ("dodging") behavior to get past the prey and 4) "freezing" behaviors when movement by the prey would put it in the immediate path of the predator (not the same as the "paralysis in the absence of the predator" described earlier).

On average, over 1,000 runs, we found that the success rate of the evolved prey in getting past the predator was above 80%. The prey would move retreat/advance with respect to the predator a minimum of three times per run, coming closer to the predator in each subsequent approach, before finally moving past the predator. Also demonstrated in the runs were lateral avoidance behaviors (once every three runs), and freezing behaviors (once ever run). Based on these data we felt that the model was able to mimic real life prey behavior in a similar situation with reasonable fidelity.
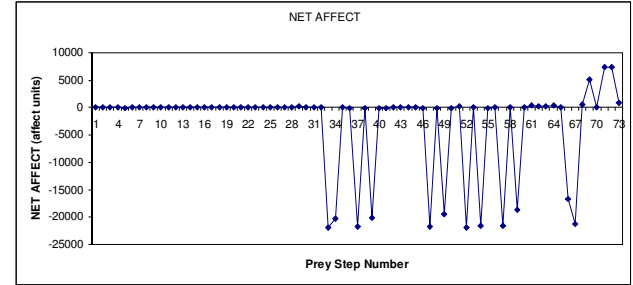

Figure 4: Variation of NET affect for sample "Turing test"

The results indicate that EP used in conjunction with the affect model appears to produce reasonable affect-based behavior. The evolved factors are specific to the environment and to the nature of the predator, as they are in nature; for example, the deer respond differently to different predators in different environments. Future work will likewise develop affect factors specific to different environments and type of objects interacted with. In short, the affect model appears generalizable to situations that are more complex in terms of environment and types of danger. The results also suggest that there is an important place for similar "Turing-like tests for affect." Next, we compare our approach with alternate approaches.

5. Comparison with other affect models

Picard [33] wrote the first book on affective computing where she used the terms "affective computing systems" and "emotion-oriented computing" to refer to systems that considered emotions. Following Picard, numerous researchers have done similar work. Space considerations permit us to address only the two major groupings of this research—appraisal theory based approaches and biologically based approaches.

5.1. Appraisal Theory Based Approaches

Appraisal theory by far is the basis of the largest amount of work in computational theories of emotion. In appraisal theory, emotions result from cognitive assessments of events in relation to one's "beliefs, desires, and intentions" [5]. The central process is "cognitive appraisal" and much work has focused on the structural relationship between appraisal variables and specific emotion variables and cognitive responses. Applications include human-computer-interaction, psychology, and AI and described in [14, 36, 40, 35, 9, 29] among others. Marinier [26] used appraisal theory as a source of intrinsic reward in "Eaters" a maze much like Pac-Man, but lacking a predator. In general, research in appraisal theory has been active for several decades; see Scherer et al. 2010 for a good summary.

The key difference between appraisal theory and the biological approach we propose is that in appraisal theory, the selection of appraisal variables is unconstrained by the need for empirical evidence in underlying biology. Instead, appraisal theory based approaches use "if-then" rules specified from outside the system. Consequently, we suggest a complementary approach; one completely constrained by the underlying biology

of affect, and with rules generated internally based on the interaction of the objects with the general goal of maximizing agent well-being. As more information of the biological underpinnings of affect becomes available, we surmise that biological based approaches, presently very much in their infancy, will eventually match the abilities of current, sophisticated, appraisal theory based approaches.

5.2. Biologically Based Approaches

Biologically based theories (including ours), try to model the biological systems that underlie affect. For this reason, we provide a more detailed review of other such research. Velasquez [42] modeled important aspects of emotional processing, for use as a biasing signal when making decisions. He identified six primary emotions based primarily on ideas from Ekman [12] and used "associative networks" comparable to Minsky's K-lines [28] in which salient external stimuli are connected to primary emotions. The main draw back of his system is that the designer explicitly specifies reward/punishment values associated with specific interactions with specific objects, and in that sense is arbitrary. This contrasts with our approach where there is no explicit specification of the reward/punishment value for each interactions and where the agent makes its own affect-based evaluations of objects.

Drawing on the work of Pfeifer [31, 32] , Canamero [6, 7] used emotions as the mechanism for action selection. She also suggested novel "artificial emotions" not found in nature, but which could be useful. While Pfeifer place basic emotions as primary motivators, Canamero considered them second-order modifiers. We do not see the need for "artificial emotions" and view the basic emotions as primary motivators. Canamero modeled six basic emotions in a neural net. We do not believe that global neural network implementations account for the modular nature of the brain's affect handling systems.

Delgado-Mata et al. [10] used an affective system as part of an ethologically-inspired action-selection mechanism for virtual animals, to show that integrating emotion resulted in emergent behavior of flocking. They did not explore the use of agents interacting with the objects to develop semantic understandings of them.

Blumberg et al. [4] used an ethologically inspired computational model for action selection and learning in an autonomous animated creature. A "Behavior System" computes the right control signals to send to the agent's motor system. Their approach is conceptually very similar to the one we propose here except their decision algorithm arbitrates among different behaviors rather than the basic emotions that drive these behaviors.

Wright [43] suggested that the assignment of value to an emotion is a "self-monitored process of credit assignment," i.e., is assigned by the agent itself. He pointed to the importance of a domain-independent representation of utility for adaptive architectures, but did not provide any implementation details. We agree strongly with both his suggestions: in our approach, agents independently develop affect-based evaluations of ob-

jects interacted with, based on the domain independent criteria of maximizing their own well-being.

6. Conclusion

We aim our future work towards generalizing our approach to handle more complex situations than the Pac-Man game. To do this, we plan to explore analytical methods such as reinforcement learning to replace the current "greedy optimization" strategy with a more efficient approach; introduce the concept of learning during an agent's lifetime based on the localized damage (health) of a single agent; introduce memory and analytical reasoning; include more emotions such as PANIC so as to model social situations, such as the teenage boy on his first date; enrich environment diversity by introducing other affect-enabled agents with different characteristics; improve the equations used in the affect model by continuing to study the underlying empirical evidence as it becomes available from mainstream neuroscience.

Rodney Brooks (2002) proposed a 'subsumption' architecture where each layer subsumes the functionality of the layer below it. In our system, there seems an obvious place for a like approach: an affect layer subsumes the layer that manages the agent's body, and with a cognitive layer subsuming the affect layer. This three-layer, "triune" architecture offers the possibility of a new kind of autonomous agent, one that because of its ability to develop affect-based understandings of objects based on its interactions with them is capable of developing semantic understandings of these objects and hence demonstrates human-like behavior in social or other complex emotional situations.

7. References

1.   Armony, J. L., Servan-Schreiber, D., Cohen, J. D., and LeDoux, J. E. 1997. Computational modeling of emotion: explorations through anatomy and physiology of fear conditioning. Trends in Cognitive Science, 1, 28-34

2.   Barrett, L. 2006. Are emotions natural kinds? Perspectives on Psychological Science 1, 28-58

3.   Barrett, L., Mesquita, B., Ochsner, K. N., and Gross, J. J. 2007. The experience of emotion. Annual Review of Psychology, 58, 373-403

4.   Blumberg, B., Todd, P., and Maes, P. 1996. No bad dogs: ethological lessons for learning. In: Proceedings of SAB'96. Cambridge, MA: The MIT Press

5.   Bratman, M. E. 1987. Intention, Plans and Practical Reason Stanford, CA: CSLI Publications

6.   Canamero, D. 1997. Modeling motivations and emotions as a basis for intelligent behavior. In W. Lewis Johnson, ed., Proceedings fo the First International Conference on Autonomous Agents, New York, NY: ACM Press, 148-155

7.   Canamero, D. 1998. Issues in the design of emotional agents. In emotional and intelligent: the tangled knot of cognition. Papers from the 1998 AAAI Fall Symposium, Technical Report FS-98-03, Meno Park, CA: AAAI Press, 49-54

8.   Combes, S. A., Rundle, D. E., Iwasaki, J. M., and Crall, J. D. 2012. Linking biomechanics and ecology through predator–

prey interactions: flight performance of dragonflies and their prey. The Journal of Experimental Biology, 215, 903-913

9.   Connat, C. and Maclaren, H. 2004. Evaluating a probabilistic model of student affect. 7th International Conference on Intelligent Tutoring Systems. Maceio, Brazil.

10.   Delgado-Mata, C, Ibanez, J. and Aylett, R.S. 2003. Let's run for it: consepcific emotional flocking triggered by virtual pheremones. Proc. Smart Graphics 2003, LNCS 2733, Springer-Verlag

11.   DeNero, J. and Klein, D. 2010. Teaching introductory artificial intelligence with Pac-Man. Symposium on Educational Advances in Artificial Intelligence

12.   Ekman, P. 1992. An argument for basic emotions. In: Stein, N. L., and Oatley, K. eds. Basic Emotions, 160-200. Hove, UK: Lawrence Erlbaum

13.   Fogel, L. 1999. Intelligence through simulated evolution: forty years of evolutionary programming. New York, NY:Wiley-Interscience

14.   Gratch, J., Marsella, S., Wang, N. and Stankoic, B. 2009. Assessing the validity of appraisal-based models of emotion. International Conference on Affective Computing and Intelligent Interaction. Amsterdam, IEEE.

15.   Greene, J. D. 2004. Why are VMPFC patients more utilitarian? A dual-process theory of moral judgment explains. Department of Psychology, Harvard University, Cambridge, Mass.

16.   Greenfield, S. 2000. The private life of the brain. New York, NY: John Wiley & Sons.

17.   Heider F. and Simmel, M. 1944. An experimental study of apparent behaviour. American Journal of Psychology, Volume 13

18.   Joseph, P. G. and Levkowitz, H. 2011. Patterns of emotion driven by affect state and environment. The Third International Conference on Pervasive Patterns and Applications, Rome, Italy (http://www.thinkmind.org/index.php?view=article&articleid=patterns_2011_3_20_70039)

19.   Joseph, P. G. and Levkowitz, H. 2012 a. An algorithmic approach to affect driven stream of consciousness. The Sixteenth International Conference on Cognitive and Neural Systems, Boston, USA

20.   Joseph, P. G. and Levkowitz, H. 2012 b. Computing affect in autonomous agents, IEEE Conference on Development and Learning—EpiRob, SanDiego, CA, USA, 2012

21.   Kalat, J. 2008. Biological psychology. Florence, KY:Wadsworth Publishing

22.   Kim, J. 1996. Philosophy of mind. Boulder, CO: Westview

23.   LeDoux, J. 2003. Synaptic self: how our brains become who we are, New York, New York: Viking

24.   Luan K. P., Wagner, T. D., Taylor, S. F., and Liberazon, I. 2004. Functional Neuroimaging Studies of Human Emotions, CNS Spectrums, Volume 9—Number 4, 258-266

25.   Mao, W. and Gratch, J. 2006. Evaluating a computational model of social causality and responsibility. 5th International Joint Conference on Autonomous Agents and Multiagent Systems. Hakodate, Japan.

26.   Marinier, R. P. 2008. A computational unification of cognitive control, emotion, and learning. PhD dissertation, Computer Science. Ann Arbor, MI, University of Michigan.

27.   Marinier, R. P., Laird, J. E., and Lewis, R. L. 2009.A computational unification of cognitive behavior and emotion. Cognitive Systems Research, 10, 48-69.

28.   Minsky, M. 1986. The society of mind. New York, NY: Simon & Schuster.

29.   Paiva, A., Dias, J. and Aylett, R. 2005. Learning by Fealing: Evoking Empathy with Synthetic Characters. Applied Artificial Intelligence special issue on "Educational Agents - Beyond Virtual Tutors", 19, 235-266.

30.   Panksepp, J. 1998, Affective neuroscience: The Foundations of Human and Animal Emotions, New York, New York: Oxford University Press

31.   Pfeifer, R. 1991. A dynamic view of emotion with an application to the classification of emotional disorders. Vrije Universiteit Brussel, AI Memo 91-8.

32.   Pfeifer, R. 1993. Studying emotions: fungus eaters. Proceedings of the Second European Conference on Artificial Life, ECAL'93, ULB, Brussels, Belgium 916-927

33.   Picard, R. 1997. Affective computing, Cambridge, USA: The MIT Press

34.   Poli, R., Langdon, W. and McPhee, N. 2008. A field guide to genetic programming. Lulu.com

35.   Purves, D., Augustine, G. A., Fitzpatrick, D., Hall, W. C., LaMantia, A., White, L. E., (2011), Neuroscience, 5th Edition, Sinaeur Associates, 858 pp.

36.   Reilly, W. S. 1996. Believable Social and Emotional Agents. Pittsburgh, PA, Carnegie Mellon University.

37.   Russell, S. and Norvig, P. 2009. Artificial intelligence, a modern approach, New Jersey: Prentice Hall

38.   Scherer, K.R., Bänziger, T., & Roesch, E. (Eds.) 2010. A blueprint for affective computing. Oxford: Oxford University Press

39.   Sterelny, K. The representational theory of mind (Oxford: Blackwell, 1990).

40.   Traum, D., Rickel, J., and Marsella, S. 2003. Negotiation over tasks in hybrid human-agent teams for simulation-based training. International Conference on Autonomous Agents and Multiagent Systems. Melbourne, Australia.

41.   Turing, A. 1950. Computing machinery and intelligence, Mind, 59, 433-460

42.   Velasquez, J. D. 1998. Modeling emotion-based decision-making. In Emotional and Intelligent: The Tangled Knot of Cognitions, Papers from the 1998 AAAI Fall Symposium. Technical Report FS-98-03. Menlo Park, CA: AAAI Press, 164-169

43.   Wright, I. 1996. Reinforcement learning and animat emotions. In: Proceedings of SAB'96, Cambridge, MA: The MIT Press

A Meta-Heuristic Approach For Energy-Efficient Multi-Agent Path Planning

Wenjie Wang, and Wooi-Boon Goh

School of Computer Engineering, Nanyang Technological University, Singapore

Abstract— *We address the problem of multi-agent path planning (MAPP) by taking into account the total energy consumption of all agents, as measured by their cumulative distance travelled. A deterministic meta-heuristic technique called guided iterative prioritized planning (GIPP) is proposed to find good solutions with respect to this cumulative cost measure in a crowded environment. GIPP progressively explores a larger search space while incorporating a local optimum detection and resolution scheme to prevent the search process from being trapped in an unfeasible locally optimal solution. Comparative evaluations show that GIPP strikes a good tradeoff between the solution quality and computational time.*

Keywords: Guided local search, path planning, multiple agents, meta-heuristic technique

1. Introduction

There exist many MAPP algorithms to navigate co-located mobile agents. However, it is still a challenging problem to ensure a dense population of such agents does not collide with each other as they concurrently move to their respective destinations. In addition, we may want these agents to perform this task in the most energy efficient manner by minimizing the cumulative distance travelled by all agents. Finding an optimal solution in MAPP can be intractable as the size of the solution space in MAPP increases rapidly with the number of agents and the size of the map. For practical computational reasons, this work adopts a local search technique to find energy efficient solutions for dense MAPP problems in a reasonable time. Unfortunately, like many traditional MAPP algorithms, the local search technique fails to find viable paths for all agents as the agent density increases since it gets trapped in locally optimal solutions that are unfeasible. We overcame this problem with an idea borrowed from guided local search and proposed a novel guided iterative prioritized planning (GIPP) algorithm that knows when to modify the cost function so as to help the search process escape from unfeasible local optima.

2. Related Work

MAPP algorithms can be categorized into coupled or decoupled approaches. Coupled approaches combine the workspace of all agents into a composite workspace, and then plan the paths for all agents simultaneously. Svestka and Overmars[4] combined the probabilistically generated map of each agent into a super graph. The viable paths for all agents are then obtained by finding a sequence of collision-free super nodes that connect the start and target super nodes in this super graph. This coupled approach is probabilistically complete because of the probabilistically generated map for each agent. Sharon et al.[6] proposed an increasing cost tree (ICT) to find an optimal solution with respect to a cumulative cost function. Though this approach is guaranteed to find the globally optimal solution, the size of ICT increases exponentially with the depth of goal node.

Decoupled approaches decompose the MAPP problem into several sub-problems and then solve these sub-problems separately. One type of the decoupled approach relies on some kind of prioritization scheme and then each sub-problem is solved in a sequential manner based on their assigned priorities. Mors et al.[3] proposed a routing planning algorithm based on free time windows (FTW) to find a viable path for each agent in a sequential manner. Since the higher priority agents do not consider the path planning process of the lower priority agents, they may inadvertently prevent lower priority agents from finding viable paths. Bennewitz et al.[1] addressed this issue by not fixing the priority order. They adopted a randomized and hill climbing search to find a near-optimal priority order. However, this random approach is not effective in finding a feasible priority order that could make viable paths for all agents. Wang and Goh[11] proposed an adaptive priority re-assignment strategy to find a feasible priority order but their technique may get stuck in a priority re-ordering loop. Silver[5] proposed a dynamic priority assignment strategy in which each agent takes turn to have the top priority for a period of time. Since it is unknown how long each agent should be assigned the top priority each time, dynamic priority assignment is still a challenging task. Standley[7] addressed this issue by allowing the coupling of the sub-problems. He proposed an Independence Detection (ID) that tries to find the independent groups of agents such that the entire problem can be solved by planning the paths for each independent group separately. In addition, he proposed an Operator Decomposition (OD) that can lead to a significant reduction of node expansions during the path planning of each independent group if a perfect heuristic is available. Though the OD+ID algorithm can guarantee the optimality of the solution, the perfect heuristic in MAPP is

impossible while the independent groups are difficult to be obtained in a crowded environment. In this case, the OD+ID algorithm may suffer from exponentially increasing computational time with the agent count. Standley and Korf[8] went on to modified OD+ID into a fast maximum group size (MGS) algorithm by dynamically dropping an optimality constraint and changing the expanding order criterion. They then proposed an optimal anytime (OA) algorithm that can improve the solution quality if more time is allowed. Though dropping the two constraints in MGS and OA may help reduce the probability of group merging, it slows down the convergence rate to an optimal solution and leads to the risk of endless idle action.

Another type of decoupled approach first solves the single agent path planning sub-problems independently and then resolves conflicts using some coordination strategies. Wang and Botea[10] adopted a puzzle sliding approach to resolve the conflicts among the paths of agents. Their approach computes alternative paths for every three locations in the independently computed path of each agent and then solves the conflicts among the paths of agents by sliding the agents with lower priorities to the alternative paths so as to give way to the high priority agents. However, this approach may fail to find alternative paths in highly crowded and narrow spaces. Luna and Berkis[2] proposed two primitive operations, namely the push and swap operations, to resolve the conflicts among the paths of agents. However, the push and swap algorithm requires at least two empty vertices in the graph, in other words, the push and swap algorithm may not be able to solve MAPP with one empty vertex in the graph like the 15 puzzle problem. Since finding an optimal solution in MAPP can be computationally intractable, the goal of this work is to find a good energy efficient MAPP solution in a reasonable time. To this end, we propose an anytime algorithm, which iteratively improves the solution.

3. Problem Definition

The energy efficient MAPP problem can be described as follows: each agent A_i for $i \in \{1, ..., n\}$, is required to move along a conflict-free route r_i from its unique starting point I_i to its destination G_i. They are required to complete this task with the lowest possible total energy consumption, which is assumed to be proportional to a cumulative cost function that measures the total travel distances of all n agents. The environment is modeled as a four connected undirected grid map and time is discretized into a series of unit time steps. The starting and target locations of each unit-sized mobile agent are directly mapped to the vertexes in the grid map. At a unit time step, each agent is allowed to take one of four move unit actions (i.e. North, South, West or East) that moves the agent from one node to the neighboring node in the four-connected grid map. Or it can stay at its current node by taking an idle unit action, which incurs no energy cost. A rotation action is assumed unnecessary when changing

directions. Assume the path node $r_i(t) = (r_i^x(t), r_i^y(t))$ denotes the position of each agent A_i at the time step t, and the agent A_i reaches its destination G_i at the time step T_i. So T_i should be large enough such that there is no collisions with the paths of other (n-1) agents if the agent A_i reaches its destination after T_i. Since all agents move concurrently, there exist two kinds of collisions. A node collision happens when two agents come to the same node at the same unit time step or one agent moves to a node occupied by static obstacles S. A head-on collision happens when two agents moves concurrently along the same edge from the opposite directions. The Boolean function $\varphi_i(t)$ denotes whether the agent A_i collides with the static obstacles S at the path node $r_i(t)$. The Boolean function $\Omega_{i,j}(t)$ denotes whether the agent A_i collides with the agent A_j at the path node $r_i(t)$. The Boolean function $\phi_{i,j}(t-1,t)$ indicates whether there exists any head-on collision between the agent A_i and A_j during time step $t-1$ and t. The Boolean function $\varphi_i(t), \Omega_{i,j}(t)$ or $\phi_{i,j}(t-1,t)$ is set to 1 if its corresponding statement is true, or else to 0. Assume the function $d_i(r)$ calculates the travel distance of the agent A_i from its starting point I_i to its destination G_i. Then the energy efficient MAPP problem is mathematically described as follows:

$$minF_d(r) = \sum_{i=1}^{n} d_i(r), \text{subjected to} \qquad (1)$$

$$\varphi_i(t) = 0 \text{ for each } \{i, t\} \qquad (2)$$

$$\Omega_{i,j}(t) \text{ and } \phi_{i,j}(t-1,t) \text{ for each } \{i,j,t\} \qquad (3)$$

Where $i \in \{1,...,n\}, j \in \{1,...,n\}, j \neq i$ and $t \in \{1,...,T\}$ ($T = max_{1 \leq i \leq n}(T_i)$). For convenience, we refer to (2) as the static constraint and (3) as the dynamic constraint and they are treated differently in this work.

Since idle action is allowed but incurs no energy cost, many alternative solutions with the equal cost exist in an energy efficient MAPP problem. In this case, a preferred solution is one with as few idle actions as possible. Since idle unit action and move unit action both take one time step, tie breaking can be done by choosing the MAPP solution with the lowest total time step. Assume the function $b_i(r)$ calculates the total time steps of the path r_i, then the second tie breaking objective is given by

$$F_t(r) = \sum_{i=1}^{n} b_i(r) \qquad (4)$$

The energy efficient solution is obtained with respect to two prioritized objectives $F_d(r)$ and $F_t(r)$. We address this vector optimization by aggregating these two objectives into a weighted function given by

$$F_a(r) = w_1 * F_d(r) + w_2 * F_t(r)$$
$$= \sum_{i=1}^{n} \{w_1 * d_i(r) + w_2 * b_i(r)\} \qquad (5)$$

Where $w_1 > w_2 > 0$ and $w_1 + w_2 = 1$. Since each path r_i is created by a sequence of unit actions, the distance and time cost functions $d_i(r)$ and $b_i(r)$ can be computed using the total cost of all the unit actions in the path r_i. However, the idle unit action is treated differently in $d_i(r)$ and $b_i(r)$. It adds no distance cost to former but contributes to the time cost of the latter. As such, we can effectively change the ratio between $w_1 * d_i(r)$ and $w_2 * b_i(r)$ by changing the relative cost between the idle and move unit actions. Assume the cost of the move unit action is 1, then the cost of the idle unit action is set to α, where $(0 < \alpha < 1)$. As a result, the energy efficient MAPP problem can be approximated by the following problem

$$minF_a(r) = \sum_{i=1}^{n} f_i(r) \qquad (6)$$

Subjected to (2) and (3), where $f_i(r) = w_1 * d_i(r) + w_2 * b_i(r)$. Since our goal is to find a good solution in a reasonable time, we next introduce a local search algorithm for the MAPP problem in (6).

4. Guided Iterative Prioritized Planning Algorithm

The local search method iteratively finds an improving solution from a local neighborhood. When evaluating a larger neighborhood, it is more likely to find a globally optimal solution at the expense of a higher computational cost. Using an efficient optimal search algorithm like the A* algorithm for single agent path planning, we can define n different neighborhood $N_i(r)$ obtained by applying changes in one component of the current solution $r = (r_1, ..., r_n)$. Beginning with an initial solution, each subsequent solution is obtained by selecting the best feasible solution from $N_i(r)$ with respect to the objective function in (6). The iterative process terminates when no further improvement can be made. Since dynamic constraints increase rapidly with agent count, the search in the MAPP solution space is likely to terminate prematurely when no feasible solution exist in $N_i(r)$. To address this, the constraints are relaxed by adding a penalty term that measures the violations of the constraints into the objective function in (6). Since eliminating the unfeasible solutions that violates static constraints does not destroy the connectedness of other solutions in the MAPP solution search space, computational cost can be reduced by strictly satisfying all static constraints. As such, the penalty function includes only a penalty term that measures the violations of the dynamic constraints and is given by

$$minF_p(r; \mu) = \sum_{i=1}^{n} f_i(r) + \mu * \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \sum_{t=1}^{T} \qquad (7)$$
$$\{\Omega_{i,j}(t) + \phi_{i,j}(t-1, t)\}$$

subjected to (2), where μ is a non-negative penalty coefficient. A fixed penalty coefficient sets μ to a high value.

This has the advantage of fast convergence but may lead to an inferior solution. The alternative incremental penalty coefficient that initially set μ to a small value and then gradually increases it, produces a superior solution but with slow convergence. For a fast feasible solution convergence, we adopted the fixed penalty coefficient. At each iteration, the next solution is obtained by selecting the lowest cost solution from $N_i(r)$ with respect to the cost function $F_p(r; \mu)$ in (7) and the static constraint (2).

Though the defined neighborhood $N_i(r)$ allows an improving solution to be found in a computationally efficient manner, the local search method can be trapped in a locally optimal solution that is unfeasible, especially with high agent count. This quickly reduces the success rate of the local search method. In order to escape from the unfeasible locally optimal solution, an idea borrowed from guided local search [9] is used to modify the cost function. In this meta-heuristic algorithm, solution features are first defined and each solution feature c_i for $i = 1, ..., m$ has a corresponding feature penalty h_i (initially set to 1). Each time a locally optimal solution x^* is reached, the penalty h_i' of the solution feature c_i' with the highest utility presented in the current locally optimal solution x^* is increased so as to make x^* more costly than surrounding solutions without the solution feature c_i'. This helps the search escape from local optima. The utility for each solution feature c_i is calculated as

$$U(x, i) = D_i(x) * g_i(x)/h_i(x) \qquad (8)$$

where $i = 1, .., m$ and $g_i(x)$ calculates the cost associated with the solution feature c_i and it can be derived from a cost function. Function $D_i(x)$ returns 1 if the feature c_i is present in the current solution x or 0 otherwise. In this way, guided local search can suppress undesirable solution features. As a result, we can define the solution features associated with the dynamic constraints in (3) so as to steer the search towards a feasible solution and avoid being stuck in an unfeasible locally optimum. Since the number of dynamic constraints could be infinite, it will be impractical to maintain infinite solution feature penalties. Instead, we define n solution features that are corresponding to the paths of each agent. Each solution feature c_i is defined as the violation of dynamic constraints imposed by the path of agent A_i. Guided local search is applied to the MAPP problem by incorporating the feature penalty to the cost function in (7) as given by

$$minF_g(r; \mu) = \sum_{i=1}^{n} f_i(r) + \mu * \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \sum_{t=1}^{T} \{0.5* \qquad (9)$$
$$h_i(r) * h_j(r) * [\Omega_{i,j}(t) + \phi_{i,j}(t-1, t)]\}$$

subjected to the static constraints in (2). Then the incurred cost $D_i(r) * g_i(r)$ associated with the solution feature c_i in

the solution r is given by

$$D_i(r) * g_i(r) = \mu * \sum_{j=1,j \neq i}^{n} \sum_{t=1}^{T} \{h_i(r) * h_j(r) \\ * [\Omega_{i,j}(t) + \phi_{i,j}(t-1,t)]\} \quad (10)$$

Since μ makes uniform scaling on utilities of each solution feature, so we define the utility function $U(r,i)$ for each solution feature c_i as

$$U(r,i) = \sum_{t=1}^{T} \sum_{j=1,j \neq i}^{n} \{h_j(r) * [\Omega_{i,j}(t) \\ + \phi_{i,j}(t-1,t)]\} \quad (11)$$

The intention is to extend local search by making other agents give way to the agent with the highest penalties. When an unfeasible locally optimal solution r^* is encountered, the utilities of all features are computed using (11). The penalty of the solution feature with the highest utility is then increased by a predetermined value ε. This iterative process stops when a feasible locally optimal solution is found. A locally optimal solution is deemed to be reached when the current solution is continuously kept for at least n iterations, where n is the number of agents. Fig 1 gives the pseudo code for the proposed GIPP algorithm.

---

1) Generate an initial solution $r^{(0)} = (r_1^{(0)}, ..., r_n^{(0)})$ and set $i \leftarrow 1, \theta \leftarrow 0, \tau \leftarrow 1$, and set a high value for μ;
2) Repeat the following procedure until there is no further improvement with the current feasible solution or an upper iteration bound δ is reached:
  a) At the iteration τ, try to find an improving solution r' with respect to the cost function $F_g(r; \mu)$ in $N_i(r^{(\tau-1)})$;
  b) $r^{(\tau)} \leftarrow r'$;
  c) If $r^{(\tau)} = r^{(\tau-1)}, \theta \leftarrow \theta+1$, or else $\theta \leftarrow 0$;
  d) If $\theta \geq n, \theta \leftarrow 0$, calculate the utility $U(r^{(\tau)}, j)$ of each feature c_j for $j=1,2,..,n$ and then increase the value of the feature penalty h_j of the feature with the maximal utility by a value of ε.
  e) $\tau = \tau + 1$;
  f) $i = 1$ if $i \geq n$, or else $i = i + 1$;

Fig. 1: Guided iterative prioritized planning algorithm.

The initial solution $r^{(0)}$ is not required to be feasible and it can be generated randomly or heuristically. In our case, we obtain $r^{(0)}$ by planning an individual path for each agent with other agents assumed to be absent because it is easy to obtain and gives the lower bound on the energy efficient solution. Since the search neighborhood $N_i(r^{(\tau-1)})$ is obtained by applying changes in one component r_i at each iteration, we essentially have a simple single agent path planning problem with the incremental positions of other agents treated as obstacles. In this work, the cooperative A* algorithm [5] was adopted to find a path for agent A_i at each iteration. Tie breaking can be done by selecting the path with the lowest traverse cost.

5. Experimental Results

The simulation program was coded in the C++ programming language and executed on an Intel(R) $core^2$ (2.67 GHz) processor with 2 GB of memory. The performance of the GIPP algorithm was evaluated with different parameter settings and against other MAPP algorithms. Performance was evaluated based on three different criteria. The first is success rate, which is defined as the percentage of times in all random scenarios all the n agents found collision-free paths to their respective destinations. The second is computational time, which is defined as the average time taken to compute the viable paths for all n agents in successful random scenarios. The last is cumulative distance, which defines the average total travel distance of all n agents when they move to their respective destinations in successful random scenarios. This criterion measures the performance of the algorithms in minimizing the energy efficient cost function stipulated in (1). All experimental results were obtained in the 30x20 grid map with 20% static obstacle density unless otherwise indicated. For each agent count, the same 100 random scenarios were used in the simulation.

5.1 Evaluation of the GIPP Algorithm

5.1.1 Different cost ratio α

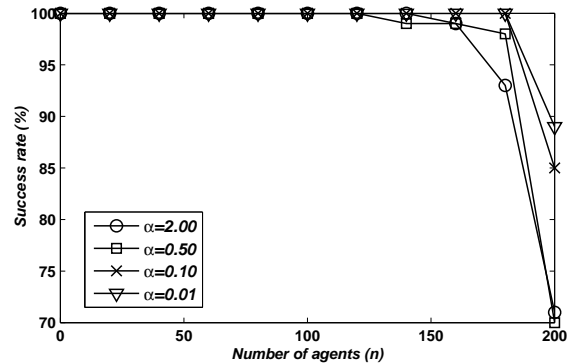Fig. 2: Comparative success rate with varying cost ratios α.

We first evaluated how the cost ratio α between the idle and move unit actions influences the search for an energy efficient MAPP solution. The performance of the GIPP algorithm (with parameter settings of $\delta = (30 * n), \mu = 100$ and $\varepsilon = 1$) was evaluated using four different idle unit action cost ratios, namely $\alpha=0.01, 0.1, 0.5$ and 2. As the relative cost of the idle unit action decreases (with smaller α), idle actions will be preferred to move actions. Since the idle unit actions produce less collision constraints compared with the move unit action (e.g. head-on collision), the success rate increases with lower cost ratio α, see Fig 2. Since idle actions do not consume energy and keep the agent from detour so as to avoid collisions, a lower cost ratio α also shortens cumulative distance, see Fig 3. The only downside
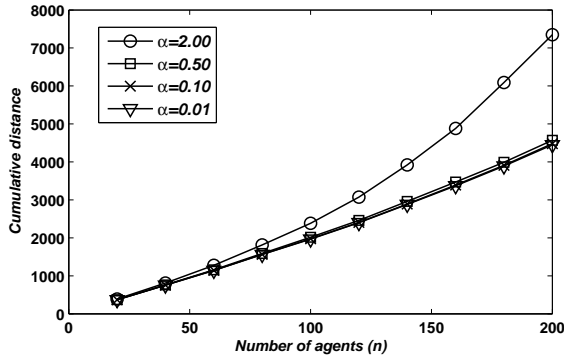
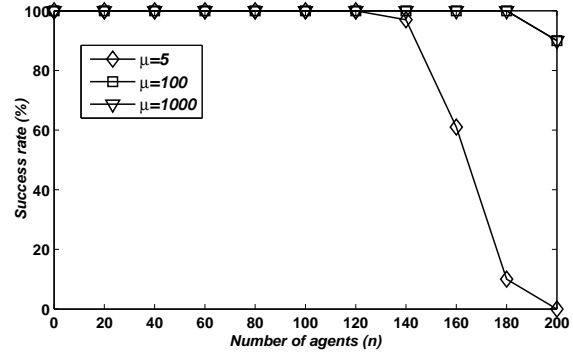Fig. 3: Comparative cumulative distance with varying cost ratios α.

Fig. 4: Comparative computational time with varying cost ratios α.

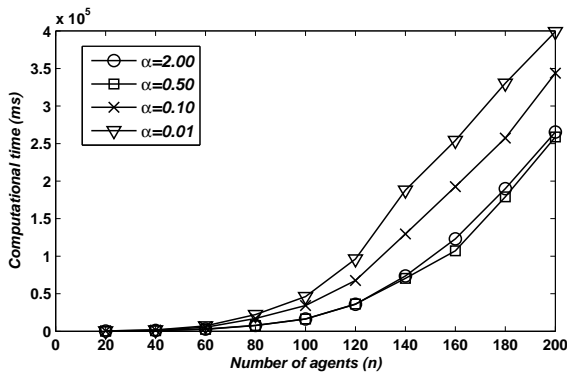Fig. 5: Comparative success rate with different fixed penalty coefficients μ.

Fig. 6: Comparative cumulative distance with different fixed penalty coefficients μ.
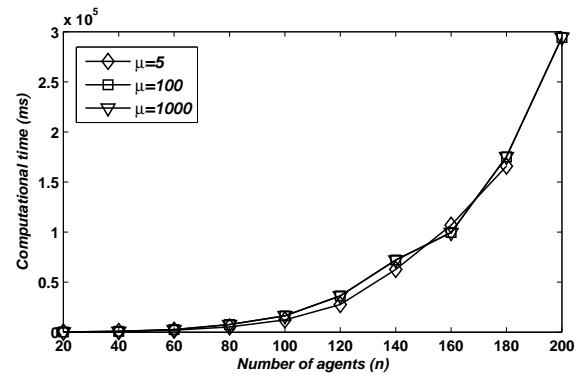
Fig. 7: Comparative computational time with different fixed penalty coefficients μ.

of lower cost ratio α is the resulting increase in feasible solutions evaluated during each solution selection process and this increases computational cost in a less crowded environment, see Fig 4.

5.1.2 Varying value for penalty coefficient μ

We also compared the performance of the GIPP algorithm (with parameter settings of $\delta = (50 * n), \alpha = 0.5$ and $\varepsilon = 1$) under different fixed penalty coefficients of μ=5, 100 and 1000. A higher penalty coefficient scales up the penalty cost and this magnifies the penalty term in the cost function, leading the GIPP algortihm to converge faster to a feasible solution. As a result, a higher penalty coefficient results in a higher success rate, see Fig 5. But the objectives $f_i(r)$ for $i = 1, ..., n$ are marginalized in the cost function, leading to inferior cumulative distance performance, see Fig 6. In addition, a high penalty coefficient increases computation cost as it may increases the cost of the lowest cost solution at each iteration of GIPP, which leads to the evaluation of more nodes that do not violate collision constraints, see Fig 7. Our observations suggest that when μ is set high enough (μ>100), the penalty term already dominates the cost function. Further increase in μ makes no difference to the
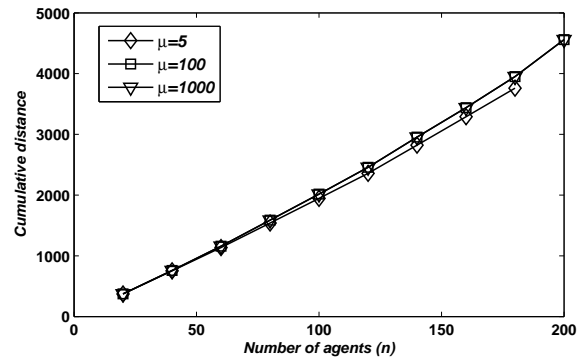
performance of the GIPP algorithm. In summary, a balanced parameter setting for the GIPP algorithm was observed to be μ=100 and α=0.5.

5.2 Comparison with other MAPP Algorithms

The proposed GIPP algorithm was compared with three other MAPP algorithms. The first is the Windowed Hierarchical Cooperative A* (WHCA*) algorithm[5], which allows all n agents to move concurrently to their respective destinations. The second is the Push and Swap (P&S) algorithm[2]. It is probably the current state of the art MAPP algorithm for finding feasible solutions in a crowded environment but it solves this problem by moving agents to their respective destination one after another. The final one is the Optimal Anytime (OA) algorithm[8], which is also an anytime algorithm like GIPP. The comparison was done for a 30x20 and a 4x4 grid map. The former with agent count varying from 20 to 200 and a static obstacles density of 20%. The latter with only agents numbering from 2 to 15. Results were obtained with GIPP parameter settings of $\mu = 100, \alpha = 0.5$ and $\varepsilon = 1$.
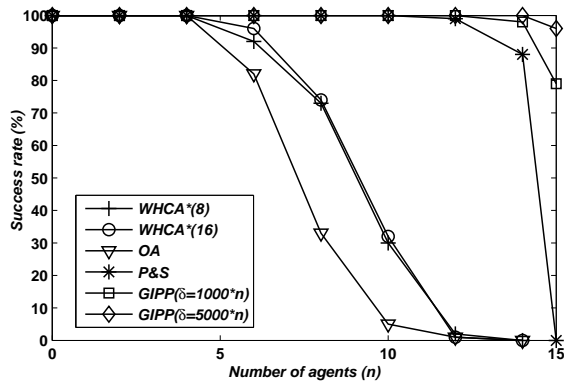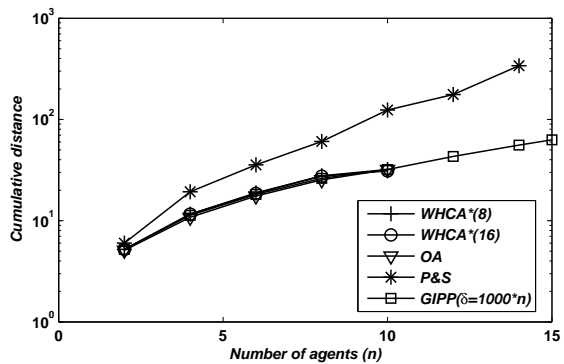

Fig. 8: Comparative success rate on a 4x4 grid map.

Fig. 9: Comparative cumulative distance on a 4x4 grid map.

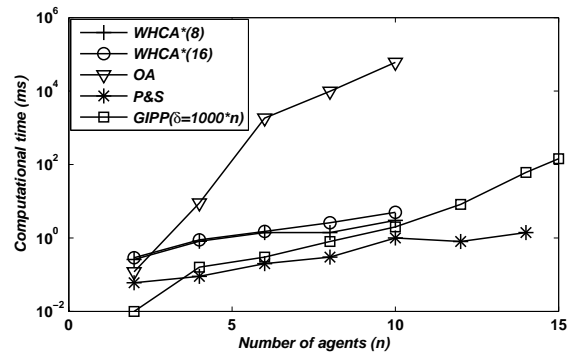The performance of GIPP was compared with the Windowed Hierarchical Cooperative A* with 8 window step

Fig. 10: Comparative computational time on a 4x4 grid map.

(WHCA*(8)) and 64 window step (WHCA*(64)), the OA (with a timeout of 1 minute) and the P&S algorithms on a 4x4 grid map. Though OA is able to find an energy optimal MAPP solution, it suffers from exponentially increasing state search space due to the group merging. This translates into a rapidly increasing computational time in OA. As the agent count increases in the 4x4 grid map, the probability of group merging increases and therefore the success rate of OA quickly decrease under the limited timeout period of 1 minute, see Fig 8. By contrast, WHCA* and GIPP can both achieve comparable cumulative distance with OA. However, WHCA*'s success rate quickly drops as the environment gets crowded. Though success rate of GIPP also drops as the agent count increases to 15, which is highly crowded, its success rate can be further improved if more iterations are allowed. Due to its efficient push and swap primitives, P&S performs well in finding a feasible MAPP solution quickly. As the agent count increases to 12, P&S can obtain comparable high success rate with GIPP at less computational time, see Fig 10. However, P&S fails as the environment gets highly crowded, especially when the agent count increases to 15 because one of the necessary conditions for its solution is at least two empty vertices in the map. In addition, P&S results in higher cumulative distances since it was not explicitly designed to find energy efficient MAPP solutions, see Fig 9.

We also compared the performance of the GIPP with the WHCA*(16), WHCA*(64), OA (with a timeout of 12 minutes) and P&S algorithms on the 30x20 grid map. A larger grid map leads to a longer path for each agent and this significantly increases the size of state search space for OA. As a result, OA quickly fails even in a sparse environment despite the longer timeout of 12 minutes. Fig 11 shows that the WHCA*'s success rate drops as the agent count increases because of its insufficient window step. Though a longer window step does improve its success rate, the improvement comes at the expense of higher computational time, see Fig 13. With agent count under 100, WHCA*(64)'s computational time is higher than GIPP. P&S is more

computationally efficient in finding a feasible solution than GIPP in a moderately crowded environment, but as expected, the energy efficiency of its solution is significant worse than GIPP. With more iteration, the success rate of GIPP can approach that of P&S, albeit at higher computational cost. If energy efficient MAPP solutions are paramount, this increase in computational cost may not be a serious problem for offline MAPP scenarios and can be addressed by using more powerful hardware.
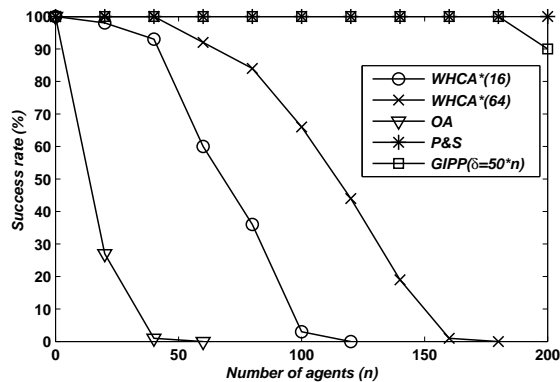

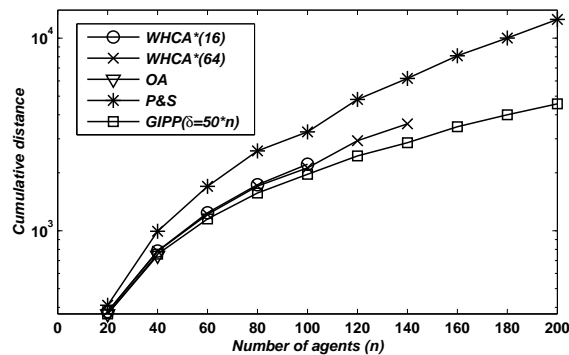Fig. 11: Comparative success rate on a 30x20 grid map.

Fig. 12: Comparative cumulative distance on a 30x20 grid map.

6. Conclusions

A meta-heuristic approach called guided iterative prioritized planning (GIPP) was proposed to address MAPP problems that require energy efficient solutions. Though the GIPP algorithm does not guarantee that a globally optimal solution can be found, it is demonstrated the most energy efficient performance among the MAPP algorithms evaluated. More importantly, GIPP also enjoys comparable success rate to the Push and Swap algorithm in a moderately crowded environment. Since the GIPP algorithm is allowed to explore the whole solution space, it does have a higher computational cost compared to WHCA*, especially in
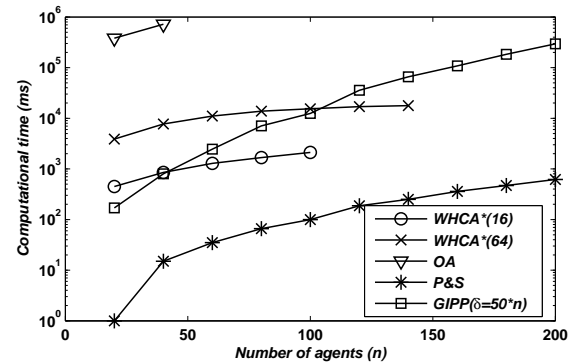

Fig. 13: Comparative computational time on a 30x20 grid map.

crowded scenarios. However, such deficiencies in an offline planning scenario can often be ameliorated by using more powerful or algorithm-customized computational hardware.

Acknowledgements

References

[1]   M. Bennewitz, W. Burgard and S. Thrun, "Optimizing Schedules for Prioritized Path Planning of Multi-Robot Systems," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 1, pp.271-276, 2001.

[2]   R. Luna and K. E. Bekris, "Push and Swap: Fast Cooperative Path-Finding with Completeness Guarantees," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pp.294-300, 2011.

[3]   A. t. Mors, C. Witteveen, J. Zutt and F. A. Kuipers, "Context-Aware Route Planning," *8th German Conference*, MATES 2010, Leipzig, Germany, pp. 138-149, 2010.

[4]   P. Svestka and M. H. Overmars, "Coordinated path planning for multiple robots," *Robotics and Autonomous Systems*, vol. 23, no. 3, pp.125-152, 1998.

[5]   D. Silver. "Cooperative pathfinding," in *AIIDE*, pp.117-122, 2005.

[6]   G. Sharon, R. Stern, M. Goldenberg and A. Felner, "The Increasing Cost Tree Search for Optimal Multi-Agent Pathfinding," in *Proceedings of the twenty-second International Joint Conference on Artificial Intelligence*, vol. 1, pp.662-667, 2011.

[7]   T. Standley, "Finding Optimal Solutions to Cooperative Pathfinding Problems," in *proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pp.173-178, 2010.

[8]   T. Standley and R. Korf, "Complete Algorithms for Cooperative Pathfinding Problems," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pp.668-673, 2011.

[9]   C. Voudouris, "Guided Local Search for Combinatorial Optimisation Problems," Ph.d. thesis, Department of Computer Science, University of Essex, 1997.

[10] K.-H. C. Wang and A. Botea, "Tractable Multi-Agent Path Planning on Grid Maps," in *Proceedings of the International Joint Conference on Artificial Intelligence*,pp.1870-1875, 2009.

[11] W. Wang and W. B. Goh, "Spatio-Temporal A* Algorithms for Offline Multiple Mobile Robot Path Planning," in *Proceedings of 10th International Conference on Autonomous Agents and Multi-Agent Systems*, pp. 1091-1092, 2011.

An Approach for Norm-Based Behavior Modification in Model-Based Reflex Agents

Gustavo A. L. Campos[1]**, Emmanuel S. S. Freire**[1]**, Mariela I. Cortés**[1] **and Wamberto W. Vasconcelos**[2]
[1]Science Computer Department, State University of Ceará, Fortaleza, Ceará, Brazil
[2]Department of Computing Science, University of Aberdeen, Aberdeen, UK

Abstract—Norms in multi-agent systems are used to regulate the behavior of agents, organizations and sub-organizations in their environments during a period of time. Much of the work on norms focuses on the specification and maintenance of normative system. Little work concentrate on the level of individual agents as the impact of norms on the different types of intelligent agent programs, as the internal modifications in the agents´ decision processes and the background information necessary for rationality in an environment governed by rules. This paper is a contribution in this direction to the case of model-based reflex agent. An approach to extend this type of agent was formalized and validated it in an implementation of a simple world in the Prolog System. The results demonstrate that the approach is adequate and must be extended to consider other kinds of intelligent agents: the goal-based agents and the utility-based agents.

Keywords—Model-Based Reflex Agent Architecture; Deontic Concepts; Norms

1 INTRODUCTION

According to [1], an agent is an entity capable of perceiving its environment through sensors and acting upon that environment through actuators. Agents are autonomous entities that are able to interact through messaging [2]. The most accepted way of classifying agents consider the different kinds of internal architectures [1, 3]. They play a central role in the development process of agents. The agent architecture specifies the components that realize the process of deliberation and choice of action to be taken by the agent, according with its perceptions and other information.

The choice of the agent architecture may be a non trivial task. It has to consider the environment task. So, the internal components in the architecture must be adequate to the external environment task properties, in order to realize the agent objectives, according to a performance measure previously established. The complexity of the choice increases if the environment task imposes norms to regulate the agents' behavior by describing the actions that can be performed or states that can be achieved (permissions), actions that must be performed or states that must be achieved (obligations), and actions that cannot be performed or states that cannot be achieved (prohibitions).

As claimed by [4], much of the work developed on norms focuses on the specification and maintenance of normative system, i.e., the research efforts has been concentrated at the macro level of the Multi-Agent Systems. Noticing that very little work has been done on the level of individual agents, they proposed some adaption in BDI agents to solve tasks in open environments with the presence of norms of four types. Strongly influenced by this work, we proposed a contribution in the same direction to the case of simple reflex agent based on condition-action rules [1, 7].

We formalized an approach to extend simple reflex agent architecture in order do considers a specific kind of norm, that is, the obligated actions that must be performed in the environment task and the prohibited actions that must not. As the work that inspired the approach [4], we did not consider the presence of sanctions and the norm related to states, because believe it would require the presence of a internal state and, in that case, we could extend the model-based reflex agent architecture.

So, this paper is a contribution to the model-based reflex agent architecture. It focuses on the second kind of norms, i.e., norms that are obligations and prohibitions to achieve states. The sanctions are not explicitly considered, but implicitly the proposed approach inherits the moral in the simple reflex agent [7], that is, in the presence an obligated and prohibited action, the agent performs a possible action that is not prohibited. This kind of behavior implies in the best agent performance in the task environment and defines the notion of rationality in specific environments. Future contributions will abolish this moral and extend the architecture to consider the presence of sanctions.

As in [7], we formalized the approach employing the Prolog System notation. It was validated in an implementation of a simplified version of a vacuum cleaner where the agent can be obligated and/or prohibited to perform some/achieve actions and states in the environment. The results demonstrate that the approach is adequate and must be extended to consider other agent architecture. The paper is structured as follows. Section 2 briefly presents the model-based agent architecture. The concepts related to norms are detailed in Section 3. Section 4 describes how to combine rightly the use of model-based agent architectures with norms. A case study is showed in Section 5 and, finally, conclusions and future works are discussed in Section 6.

2 MODEL-BASED REFLEX AGENT

The structure of this agent can be seen as the simple reflex agent with condition-action rules [1], with the insertion of a next function used to adapt an internal state to deal with partially observable environments. This new information describes aspects of the environment (called a model) that are not currently perceived by the agent´s sensors. Specifically, the next state function adapts the current internal state considering the current information perception and the information about the effects of possible actions in the environment and about how the environment evolves independently of the agent´s actions. The

Figure 1 presents a schematic diagram of the model-based reflex agent that synthesizes the Russell&Norvig´s ideas related to reactive agent program [1], as well as the abstract architecture point of view proposed by Wooldridge in [5].
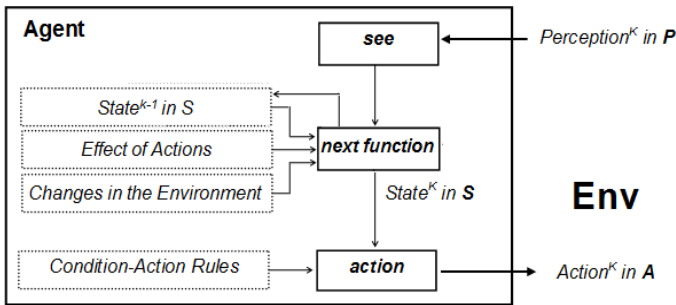

Fig. 1.   Schematic diagram of the model-based reflex agent

In addition to the functional description realized in [6] to the simple reflex agent, the description of this new structure adds an intermediate step in the five steps scheme. This new synthesis assumes that at any instant K:

(1) through sensors, the agent receives information from the environment, i.e., perceptions defined on a set of n possible perceptions from the environment (Env), P = {Perception$_1$, ..., Perception$_n$};

(2) the perception subsystem, see: P → P', processes each perception in P and maps to one possible states P', that are representations of aspects in the perceptual information that are not accessible to the agent;

(3) the next state function, next: P'xS → S, maps states in P' and the current internal state in one of possible new internal state, considering the effect of its actions and the analyze of the changes in the environment, S = {State$_1$, ..., State$_m$},;

(4) the decision making subsystem, action: S → A, processes the internal states in S and selects, according an specific rule of the set of condition-action rules, one of the actions in the set of possible actions for the agent, A = {Action$_1$, ..., Action$_n$};

(5) through actuators, the agent sends the selected action for the environment;

(6) in the interaction K+1, the agent initiates another cycle involving the perception of the world through the see function, the update of its internal state through next and the selection of a new action by action function.

In this new scheme, the condition-action rules can be seen as a set of common associations which are observed between certain conditions established from the descriptions of internal states in S and certain actions in A. The agent´s designer defines these rules having in mind the performance measure that will be applied to the agent. In this context, it is expected that, in an adequate environment, if the rules are adequate, then the agent will achieve its objectives and, consequently, it will be well evaluated.

3   NORM TYPES AND REPRESENTATION

This paper considers all types of norms discussed in [4]. The classification scheme considers whether norms are obligations or prohibitions, and whether they refer to particular actions or to states of the world. Table 1 presents the four types of norms and their intuitive meaning according to [4].

TABLE I.   THE TYPES OF NORM AND THEIR DESCRIPTIONS

	Norms Types	Description
1	obligation(**p**)	agent must try to achieve certain world state **p**
2	obligation(**a**)	agent must try to execute certain actions **a**
3	prohibition(**p**)	agent must try to refrain from achieving a state **p**
4	prohibition(**a**)	agent must try to refrain from executing an action **a**

Each type of norm has activation and expiration conditions as, for instance, a well defined validity period of time, indicating when the norm is in force and when it ceases to be in force. The schematic formalization described in [4, 6] includes notions of activation and expiration of a norm, which we represent as norm (Activation, Expiration, NormType), where Activation is the condition for the norm to become active, Expiration is the condition to deactivate (expire) the norm, and NormType is the one of the norms in Table 1. For an example, considering the vacuum cleaner agent world, the following skeleton of norm can be used to inform that the agent is obliged to keep the room clean during a period of time: norm(room(X), period(Begin, End), obligation(Clean)). So as was done in [7], it was sufficient to adapt some aspects on the description of these conditions to incorporate norm based behavior in the model-based reflex agent structure.

4   EXTENDING MODEL-BASED REFLEX AGENT WITH NORMS

In this section we propose an extension to the reference model-based reflex agent to enable the processing of norms. Most of the changes occurred in the next function, as a consequence of the insertion of one new mechanism to deal specifically with norms related to obligations/prohibitions of states that can/cannot be achieved in the world. Norms related to obligations/prohibitions of actions are normally asserted in the internal state of the agent and there is no need of a special treatment to be made by the next function in this case. This occurs because the action function in Figure 2 inherited the scheme developed for the simple reflex agent in an environment with norms related to actions [7]. So, no structural change was made in the previous work..

4.1   An Outline of the Approach

Figure 2 shows the schematic diagram of the Model-based Simplex-Reflex Agent in an environment with a specified set of norms that restrict its actions. The schematic diagram in Figure 1 describing the agent, and its associated sequence of six steps, was adapted to become valid in the case norms were activated. The adaption in the next function, step 3 in the sequence, was necessary to deal with environments with norms of Type 1 or Type 3. The adaption considered that: (1) norms of Type 1 implicitly indicate an equivalent norm of Type 2, that is, the implicit actions that achieve the obliged state in the equivalent norm Type 1; and (2) norms of Type 3 implicitly indicate an equivalent norm of Type 4, that is, the implicit prohibited actions

that achieve the prohibited state in the equivalent norm Type 3. Base on this kind of equivalence, for example, in an interaction k in which AC is true and EX is false, any norm of the form norm(AC, EX, obligation(clean)) can be "converted" to a norm of the form norm(AC, EX, obligation(suck)), because the agent can maintains the environment clean by sucking the environment in the interaction k.
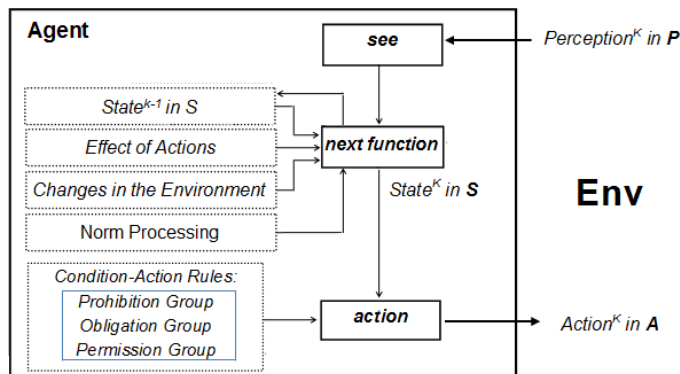

Fig. 2.   Schematic diagram of the model-based reflex agent with norms

So, based on this kind of equivalence, during an interaction k with the environment, two situations of conversion were predicted: (1) if the agent's next function detects that one norm of Type 1 is active, then: (a) it uses a successor function to infer the actions that transform the current state in the currently obliged state, and (b) it asserts in the internal state only one equivalent temporary norm Type 2, that is, that one related to the obligation of the cheapest action inferred; and (2) .if the agent's next function detects that one norm of Type 3 is active, then: (a) it uses the successor function to infer the actions and (b) it asserts all equivalent temporary norms related to the obligation of the all actions inferred. Each necessary temporary norm inherits the AC and EX conditions from the original active norm. In the interaction k+1, the agent´s next function retracts all temporary norms asserted in the interaction k, and, when necessary, it repeats the processing of situations (1) and/or (2). Norms of Type 2 and Type 4 don´t require a special treatment; the next function only asserts in the internal state this kind of norm as a non temporary norm.

Based on all norms Type 2 and Type 4 asserted in the internal state, the temporary and the non temporary norms, the agent's action function can select exactly the most successful based on the scheme explained in [7]. It involves the definition of three different groups of condition-action rules associated with the deontic concepts: obligation rules group, to specify the rules related with the actions that must be performed by the agents in the environment; prohibition rules group, to specify the rules that are related with the actions that cannot be performed by the agent; and permission rules group, to specify the rules related with the actions that can be executed. The conflicts between rules from the different groups is avoided as: (1) first, the agent looks for actions that are obliged and are not prohibited to be performed in the environment; (2) if these actions are prohibited, then the agent looks for actions that are not prohibited; and (3) in the case where there not exist any obligations and prohibitions, the agent looks for a successful action that can be performed.

A more specified description of the outcomes in the decision making subsystem of the extended can be found in [7]. So, for example, suppose the situation (1) where there is an active state norm of the form obligation(clean) and the agent knows that, independently of the current state of the room, the suck action keeps the room clean, then, employing the next function the agent. T. As the first outcome, the next function infers the suck action and, as a second outcome, it asserts in its internal state an information about a temporary norm of the form obligation(suck). Finally, as a third outcome, consequence of the assertion of the temporal norm, the action function selects the suck action if it is not prohibited. The strategy to select a rational action tries to maximize the implicit rewards that are consequences of: (a) the selection of an obligatory and not prohibited action; (b) the non-selection of a prohibited action, and (c) the selection of permitted actions, those which are adequate with the environment state conditions and with the agent´s performance measure.

4.2   The Formal Description of the Approach without Norms

This section presents a declarative description of the model-based reflex agent illustrated in Figure 1, that is, in an environment without norms. The Prolog System was employed to formally describe the agent programs in both cases. The Prolog definition of agent/2 describes the agent program discussed in Section II:

agent(P,A):- see(P,Pline), next(Pline), action(A).

where the definition of see/2 describes the perception subsystem, as indicated by Step 2 of the sequence of six steps illustrated in Section 2; the definition of nextFunction/2 describes the next function defined in step 3, it asserts the information in Pline in the agent's internal state, that is, some predicates employed to represent the environment task state; and the definition of action/1 describes the decision-making subsystem of the agent program, as indicated by Step 4 in the sequence.

The Prolog definitions next/1 depends on the specific environment task. The case study in the last section specifies this definition to the case of a vacuum cleaner agent in a very simple world. In this section we present the skeleton of the main kind of phrase in the definition:

next(Pline):- retract_all_ predicates(Pline),
        asserta_all_predicates(Pline).

where retract_all_predicates/1 is the definition of retract the set of all predicates asserted in interaction k, but that must be retracted from the internal state in the interaction k+1; and asserta_all_predicates/1 is the definition of assert the set of all predicates that must be asserted in interaction k+1.

The definition of action/1 inherits the definition employed in the case of the simple reflex agent to describe its decision making subsystem based on condition-actions rules:

action(A):- do_permission(A).

where the definition of do_permission/1 describes the Permission Rules Group for the agent in an environment without norms, which are condition-action rules highlighted in Figure 1. These rules depend of the agent's task environment and must be defined according to the available knowledge about the actions that are

most adequate to be executed when the environment is under certain conditions. The next section of this paper illustrates this group of rules for the vacuum cleaner case.

4.3   The Formal Description of the Approach with Norms

The declarative description of the model-based reflex agent in an environment with norms, illustrated in Figure 2, inherits the definition of agent/1 declared to the case without norms. But, in order to deal with all types of norms shown in Table 1, the definition of next/1 must be adapted:

next(Pline):- retract_all_ predicates(Pline),
                    asserta_all_predicates(Pline),
                    retract_all_temp_norms,
                    asserta_all_active_temp_norms.

where retract_all_temp_norms/0 is the definition of retract all temporary norms asserted in the interaction k, and the definition asserta_all_active_temp_norms/0 is the definition of assert all temporary norms related to the obligation/prohibition of executing actions, those that are equivalent to norms related to the obligation/prohibition of achieving states.

So, in order to deal with temporary and non temporary norms related to actions, definition of action/1 must be extended too. The definition below describes the new action function considering the two new groups of condition-action rules highlighted in Figure 2:

action(A):- do(A).

do(A):- do_obligation(A),!.
do(A):- do_prohibition(A),!.
do(A):- do_pemission(A),!.

where the definitions of do_obligation/1 and of do_prohibition/1 are not domain-specific, being generically defined, and are related, as we saw that the agent will execute an action which is obligated only if it is not prohibited.

The three rules below compose the Obligation Group and are in accordance with [7], but we gave an upgrade in the definitions previously described:

do_obligation(A):- obliged(Action), prohibited(Action),!,
                           do_prohibition(A).
do_obligation(A):- obliged(A).

obliged(A):- norm(AC, EC, obligation(A)), is_action(A),
                   is_True(AC), is_False(EX).

prohibited(A):- norm(AC, EC, prohibition(A)), is_action(A),
                    is_True(AC), is_False(EC).

where the predicates is_True/1, is_False/ and is_action/1 must be specified according, respectively, with the activation and expiration conditions being considered in the arguments of the predicates stating norms, and with the symbols/structures employed to represent possible actions.

The first rules of the above definition are related to the situation described in row 2 of Table 2, that is, in which the activation condition (AC) is True and the expiration condition (EC) is False in some norm of Type 2. The outcome of this situation is that the agent must perform an action (A) that is not

prohibited, but is permitted and adequate. The second rule considers the case in which none norm of Type 4 is active, that is, in which the agent must perform the action that is obligated by the active norm of Type 2.

The three rules below compose the Pohibition Group and are in accordance with [7], with an upgrade:

do_prohibition(Action):- prohibited(A),
                                 do_permission(Action),
                                 Action \= A, not(prohibited(Action)).

prohibited(A):- norm(AC, EC, prohibition(A)), is_action(A),
                    is_True(AC), is_False(EC).

The first rule of the above definition is related to the situation described in row 2 of Table 3, that is, in which the activation condition (AC) is True and the expiration condition (EC) is False. The outcome of this situation is that the agent must perform an action (Action) different from the prohibited action (A), but which is permitted and adequate, according with the conditions of the environment and the rules in the definition do_permission/1.

According to the last rule in the definition of do/1, if none of the prohibitions is active, then the agent must perform an action that is permitted and adequate. So, this rule completes the groups of rules which the approach supposes be a necessary modification to produce a rational behavior in a model-based reflex agent in an environment with the presence of all types of norms.

5 CASE STUDY

This section describes two experiments considering a very simple, but useful, problem to illustrate the ideas discussed in the last two sections: the vacuum cleaner agent with local sensors in a world containing only two rooms, in two versions: without norms and with the presence of norms. We assume the eight possible perceptions (P), which were represented by P = {[roomA,dirty,dirty], [roomA,,dirty,clean], [roomA,,clean,dirty], [roomA,,clean,clean], [roomB,,dirty,dirty], [roomB,dirty,clean], [roomB,clean,dirty], [roomB,clean,clean]}. We assume also four possible actions for the vacuum cleaner (A), which were represented by: A = {suck, right, left, no_op}. The performance measure related to the execution of action in the environment without active norms offers the reward of one point per each square clean (+1) and penalizes with the loss of one point per each movement (-1). In the presence of active norms, the measure offers three points per obligation (+3) and two points per prohibition (+2) that the agent decides to agree.

A.   Extended Vacuum Cleaner Agent

This section presents a declarative description of the vacuum cleaner agent as a model-based reflex agent program in an environment with norms, as illustrated in Figure 2. The definition of cleaner/2 describes the agent program:

cleaner(P,A):- see(P,Pline), next(Pline), action(A).

The definition of next/1 depends on vacuum cleaner world defined above. The definitions of retract_all_predicates/1 and asserta_all_predicates/1, in the skeleton proposed for next/1, were designed in following way:

retract_all_ predicates([in(Room),_]):-

retract(in(_)), retract(is(Room,_)).

asserta_all_predicates([in(Room),State]):-
        retract(clock(T)), T1 is T+1, asserta(clock(T1)),
        asserta(in(Room)), asserta(State).

where the predicates in/1, is/2 and clock/1 are dynamic predicates used to represent respectively the conditions in the antecedents of the rules do_permission/1 and the agent´s simplified clock in the experiments.

The two phrases bellow specifies the definition of is_True/1 and is_False/ to the vacuum cleaner agent:

is_True(Room):- in(Room).
is_False(time([Begin,End])):-
            clock(Hour), Hour >= Begin, Hour =< End.

Finally, the definition do_permission/1 implements the Permission Rules for the agent in an environment without norms. These rules are the condition-action rules highlighted in Figure 1. For this Case, was sufficient to generate a definition with six phrases, i.e:

do_permission(right):- in(roomA),
                is(roomA, clean), is(roomB, clean),
                norm(AC, EC, obligation(_)),
                not(is_True(AC)), is_False(EC).
do_permission(left):- in(roomB),
                is(roomA, clean), is(roomB, clean),
                norm(AC, EX, obligation(_)),
                not(is_True(AC)), is_False(EX).
do_permission(no_op):- is(roomA, clean), is(roomB, clean).
do_permission(suck):- in(Room), is(Room, dirty).
do_permission(right):- in(roomA).
do_permission(left):-   in(roomB).
do_permission(crazy).

where the two first phrases were inserted in the definition just to prevent that the agent ceases to act (no_op) in situations where the rooms are clean, but there is some active obligation; and the last phrase, was inserted to mark situations in which there are no action available to the agent, due to the presence of norm restrictions..

B. Testing the Vacuum Cleaner

Table II highlights the episodes performed by cleaner/1 in seven interactions with an environment without norms where, initially, the agent is in roomA, and roomA and roomB are dirty.

TABLE II.   PERFORMANCE IN A WORLD WITHOUT NORMS

k	Episode	Val	Hist
1	(roomA, dirty) → sukc	1	1
2	(roomA, clean) → right	-1	0
3	(roomB, dirty) → suck	1	1
4	(roomB, clean) → no_op	0	1
5	(roomB, clean) → no_op	0	1
6	(roomB, clean) → no_op	0	1
7	(roomB, clean) → no_op	0	1

The second column in the table identifies the episode in a current interaction k, and the third and fourth columns identify the

values of performance measure per episode (Val) and history (Hist). Since the perception is local and at the agent has an internal state to avoid unnecessary movements (interactions four to seven), there were no surprises in the behavior of the agent. The set of rules do_permission/1 provided a rational behavior for the agent.

Table III highlights the episodes performed by the agent in an environment in the same initial state from Table II, but with two norms, Type 2 and Type 4, that say respectively: (a) "The agent must suck the room-A from 4:00 to 6:00 a.m.", and (b) "The cannot go left in room-B from 5:00 to 6:00 a.m.".

TABLE III.   PERFORMANCE IN A WORLD WITH NORMS TYPE 2 AND TYPE4

k	Episode	Val	Hist	AC
1	(roomA, dirty) → sukc	1	1	
2	(roomA, clean) → right	-1	0	
3	(roomB, dirty) → suck	1	1	
4	(roomB, clean) → left	1	0	
5	(roomA, clean) → suck	3	3	(a)
6	(roomA, clean) → suck	3	6	(a)
7	(roomA, clean) → no_op	0	6	

The last column in the table III identifies the active norms in a current interaction k. There were no temporary active norms, as norms of Type 2 do not produce temporary norms. It is important to note that only norm (a) was active during interactions five and six. The agent won a lot of points sucking a clean room. In an environment without norms, the agent would have been penalized by this unnecessary action.

Table IV highlights the episodes performed by the agent in an environment where, initially, the agent is in roomA, and roomA and roomB are clean. In addition, in this environment there are two norms, Type 1 and Type 4, that say: (a) "The agent must keep the room-A clean from 1:00 to 5:00 a.m.", and (b) "The agent cannot go left in room-B from 1:00 to 5:00 a.m.". Table V highlights the active temporary and non temporary norms during the first five interactions.

TABLE IV.   PERFORMANCE IN A WORLD WITH NORMS TYPE 1 AND TYPE 4

k	Episode	Val	Hist
1	(roomA, clean) → no_op	3	3
2	(roomA, clean) → no_op	3	6
3	(roomA, clean) → no_op	3	9
4	(roomA, clean) → no_op	3	12
5	(roomA, clean) → no_op	3	15
6	(roomA, clean) → right	-1	14
7	(roomB, clean) → no_op	0	14

TABLE V.   TEMPORARY AND NON TEMPORARY ACTIVE NORMS

k	Temporary			AC
1	roomA	time([1, 5])	obligation(no_op)	(a)
2	roomA	time([2, 5])	obligation(no_op)	(a)
3	roomA	time([3, 5])	obligation(no_op)	(a)
4	roomA	time([4, 5])	obligation(no_op)	(a)
5	roomA	time([5, 5])	obligation(no_op)	(a)

The cleaner agent won a lot of points in each of the five interactions in which the non temporary norm (a) was active. This norm is of Type 1 and, in each interaction, the next function

asserted an equivalent temporary norm of Type 2. The good performance of the agent was due to the fulfillment of this temporary norm, that is, the non operation during five interactions. According to the last column of Table V, norm (b) was not active during any interaction. It happened because in the first five interactions the norm activation condition was false, and in the two last ones its expiration condition was true. The temporary and non temporary norms are shown in Table V.

Table VI highlights the episodes performed by the agent in an environment where the agent is in roomA, and roomA and roomB are dirty. In addition, in this environment, there are two norms, Type 1 and Type 3, that say: (a) "The agent must stay in room-A from 1:00 to 2:00 a.m.", and (b) "The agent cannot keep room-A clean from 1:00 to 2:00 a.m.". Table VII highlights the active temporary and non temporary norms during the first two interactions.

TABLE VI.   PERFORMANCE IN A WORLD WITH NORMS TYPE 1 AND TYPE 3

k	Episode	Val	Hist
1	(roomA, dirty) → no_op	3	3
2	(roomA, dirty) → no_op	3	6
3	(roomA, dirty) → suck	1	7
4	(roomA, clean) → right	-1	6
5	(roomB, dirty) → suck	1	7
6	(roomB, clean) → no_op	0	7
7	(roomB, clean) → no_op	0	7

TABLE VII.   TEMPORARY ACTIVE NORMS

k	Temporary			AC
1	roomA	time([1, 2])	obligation(no_op)	(a),(b)
	roomA	time([1, 2])	prohibition(suck)	
2	roomA	time([2, 2])	obligation(no_op)	(a),(b)
	roomA	time([2, 2])	prohibition(suck)	

During the two first interaction norms (a) and (b) were active. Norm (a) is of Type 1 and, in each interaction, the next function asserted an equivalent temporary norm of Type 2. Norm (b) is of Type 3 and the next function asserted an equivalent temporary norm of Type 4. It is important to notice that the agent won two points in each interaction. It was due to the fulfillment of the obligation in the equivalent temporary norm of (a). But, the non operation in room-A during the two interaction, fulfill too the prohibition in the equivalent temporary norm of (b), that is, the agent didn´t suck the dirty room-A. The temporary and non temporary norms are shown in Table VII.

6 CONCLUSIONS AND FUTURE WORK

In this work we discuss the influence of the norm concepts related to the model-based agent architecture in order to improve the performance of the agents executing in an environment governed by norms. In the original conception of model-based architecture, only permission rules are considered. Now, in the proposed approach two new sets of condition-action rules are incorporated in order to define prohibition and obligation rules. Additionally, the logic to implement the behavior of the model-based agents is presented. The strategy consists in select the actions aiming to minimize the penalties and to maximize the rewards. The case study simulates the behavior of a model-based agent in a vacuum cleaner word governed by obligation and prohibition rules. In this simple scenario, the proposed approach involving the implicit definition of norms as condition-action rules provides a rational behavior in consistency with the agent architecture specifications. Future work includes the analysis of the sanctions in the decision making of the architecture. In addition, the analysis of the agent behavior for the other agent architectures in the literature, considering the influence of norms in the decision making process in order to the agents can understand their responsibilities and the responsibilities of the others.

References

[1] Russell, S. and Norvig, P., 2003. Artificial Intelligence: A Modern Approach, 2nd Ed., Upper Saddle River, NJ: Prentice Hall, ISBN 0-13-790395-2.

[2] Wagner, G., 2003. The Agent-Object-Relationship Meta-Model: Towards a Unified View of State and Behavior. Information Systems, v. 28, n.5, pp. 475–504.

[3] Wooldridge, M. and Jennings, N. R., 1995. Intelligent Agents: Theory and Practice. Knowledge Engineering Review, Vol. 10, No. 2. Cambridge: Cambridge University Press, 1995.

[4] Meneguzzi, F. and Luck, M., 2009. Norm-based behaviour modification in BDI agents. In: 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009), Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary.

[5] Weiss, G., 1999. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press.

[6] Oren, N., Panagiotidi, S., Vazquez-Salceda, J., Modgil, S., Luck, M., and Miles, S., 2008. Towards a formalisation of electronic contracting environments. In Proc. 12th COIN Workshop, pages 61–68.

[7] Campos, G. A. L. , Freire, E. S. S. and Cortés, M. I., 2012. Norm-Based Behavior Modification in Reflex Agents. In: 14th International Conference on Artificial Intelligence (ICAI@WORLDCOMP), 2012, Las Vegas (USA). Proceedings of the 14th International Conference on Artificial Intelligence, 2012.

Coalition-based Approach for Reach Stackers Routing Problem in Container Terminals

H. Bouzidi, M. Kefi and K. Ghédira

SOIE Laboratory, University of Tunis, Tunisia

Abstract— *The emergence of containerization had revolutionized the freight maritime transport. The number of the daily handled containers is continuously growing, and port authorities are facing new challenges.*

Depending on the organization and the level of automation of the container terminal (CT), different vehicles are used to transport the containers, such as Automated Guided Vehicles , Reach Stackers, Straddle Carriers. In our case, we focus on using Reach Stackers.

Throughout this paper, we present, formulate and propose a model to solve the Reach Stackers Routing Problem(RSRP) as a Vehicle Routing Problem with Time Windows and Pickup and Delivery (VRPTWPD). First, we draw an analogy between these two problems. Second, we present a state of the art of methods dealing with the vehicle routing problems in CT. Finally, we propose an approximate Multi-Agent method, using coalition formation approach and implementing heuristics.

We aim to minimize the service time of containers while respecting time windows and RS capacity constraints.

To assess the performance of the proposed model , we exhibit an experimental study based on the Mitrovic-Minic benchmark. The results were very encouraging, showing a high efficiency ratio and a reasonable number of used vehicles.

Keywords

Maritime Container Handling, VRP, Negotiation, Multi-Agent Systems

1. Introduction

Container terminals(CT) are considered as complex systems and include various decision problems. These decisions are often formulated as scheduling problems. In [1], authors classify them into four classes: Arrival of the vessels, Unloading and loading of the vessels, Transport of containers and containers Stacking. Many researchers are interested to these different issues in order to improve terminals performance and optimize resources allocation. The present work focuses on the container transport problem, inside the CT and especially the Reach Stackers Routing Problem (RSRP).

When studying this problem we noticed its strong similarity to a well known problem, the Vehicle Routing Problem (VRP). This problem has been extensively studied and a multitude of resolution techniques have been proposed. In the literature, we distinguish many versions of the VRP, therefore we will try to determine which version is closer to our problem and draw the analogy between them.

The remaining of the paper is organized as follows. In Section 2 we present the RSRP and we detail its mathematical formulation. Section 3 provides a literature review of VRP and containers transport problem in CT. Section 4 describes the proposed approach which is based on coalition formation. Section 5 reports the experimental results and finally some conclusions are drawn in section 6.

2. Problem statement

2.1 Description

Depending on the organization and the level of automation of the CT, various types of vehicles are used to transport containers, such as Automated Guided Vehicles(AGV), Reach Stackers(RS) and Straddle Carriers (SC). In our case, we are interested in Reach Stackers. We consider a homogenous fleet where all the vehicles have the same characteristics (capacity, speed, maximum service time...) and are guided by a driver.
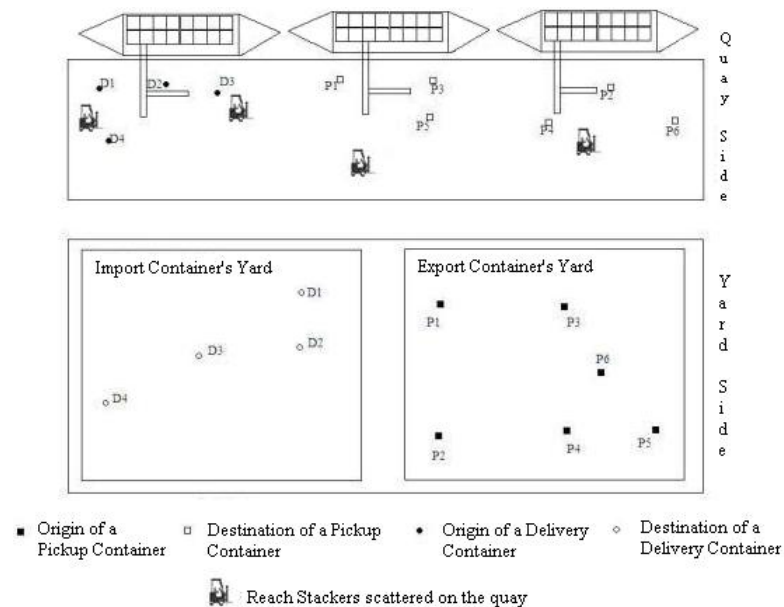

Fig. 2.1: Terminal organization

As shown in Fig. 2.1, the RS are scattered on the quay, which is considered as a kind of depot, and they have to transport the containers that are unloaded from vessels from the quay to their locations in the storage area, and collect other containers from the storage area and stack them on the quay, to be loaded later on the corresponding ship.

In addition, the operations of pick up and delivery should take place in well-defined intervals or slots(depending on the time required for a crane to retrieve the container from the vessel or the storage areas). Based on this description we note that the closest version of the VRP, to our problem, is the VRP with time windows and pickup and delivery. In Table 1, we dress the analogy between these two problems.

Table 1: Analogy between the VRP and the RSRP

VRPTWPD	RSRP
Depot	Quay
Vehicles	Reach Stackers (RS)
Customers	Locations of containers (container's coordinates)
Delivery	Stacking the container in the storage area
Pickup	Unstacking the container from the storage area
Time Windows	Container's up-time after it's handling by a quay/yard crane

2.2 Mathematical formulation

In this problem we consider several vessels loading or unloading containers in the current planning period, these vessels are going to be served by a Reach Stackers fleet.

Table 2: Notations

Symbol	Description
X_{ijv}	A binary variable indicating whether the vehicle v carries a container from the point i to j
V	A set of vehicles V = {1,2, ..., k}
P	A Set of points to visit (location of containers to pick up (n) and those to deliver(m))
Tt_{ij}	The duration of the transport of a container from a point i to j
Tm	The duration of containers handling (loading by a crane from blocks or to trucks) .
$ddvi$	The date of departure of the vehicle v from i
$[a_i, b_i]$	The time windows of the operations of pickup or delivery of a container in a point i.
(x_i, y_i)	The coordinates of the location of a container
D_i	The destination of the container collected at a point i
$DistMax$	The maximum distance a vehicle can daily travel.
$Distij$	The distance between the point i and the point j.

In what follows, we use the notations described in Table 2 and we consider the following assumptions:

1) A storage area is either dedicated to the export operations or the import operations.
2) We have a fleet of Reach Stackers (vehicles) of the same capacity (One container).

3) The RS are scattered on the quay, this is equivalent to a one single depot.
4) The Origin and destination of each container are known in advance.

The objective is to reduce the overall service time of the entire set of containers and to minimize the maximum number of RS used. Given these descriptions and hypothesis we formally define the RSRP as:

$$\min \sum_{v \in V} \left[\sum_{i \in P} \sum_{j \in P,\ j \neq i} X_{ijv} \times (Tt_{ij} + Tm) \right]$$

Subject to:

$$\sum_{i \in P} X_{ijv} = 1, \forall j \in P\ et\ \forall v \in V \qquad (2.1)$$

$$\sum_{j \in P} X_{ijv} = 1, \forall i \in P\ et\ \forall v \in V \qquad (2.2)$$

$$X_{i\,Di\,v} = 1, \forall i \in P\ et\ \forall v \in V \qquad (2.3)$$

$$a_i \leq dd_{vi} \leq b_i, \forall i \in P \qquad (2.4)$$

$$dd_{vi} + Tm_i + Tt_{ij} \leq dd_{vj}, \forall i, j \in P,\ j \neq i\ et\ v \in V \quad (2.5)$$

$$\sum_{i \in P} \sum_{j \in P, j \neq i} Dist_{ij} X_{ijv} \leq DistMax, \forall v \in V \qquad (2.6)$$

As we have already presented our objective is to reduce the service time of the containers transportation. This objective is subject to the following constraints

- Constraints (2.1) and (2.2) ensure that each point is visited only once and is only involved in one tour.
- Constraint (2.3) ensures that the RS can visit a destination point only after collecting a container. Besides, a full RS can visit another pickup point only delivering the current carried container.
- Constraint (2.4) states that the operation of pickup/delivery must be made within a defined time window.
- The constraint (2.5) is a precedence constraint between points.
- The constraint (2.6), ensures that the distance traveled by a vehicle during a route must not exceed the maximum distance that it can perform.

3. Literature review

Considering the analogy between our RSRP and VRPTWPD, we firstly present a review of the main works related to the VRP. Then, we especially focus on works dealing with container handling.

Many exact and approximate methods have been proposed to formulate and solve the VRP in the literature. Among the most known exact methods,we mention Branch and Bound algorithm, linear programming and dynamic programming detailed in [12, 20].

Because of the high complexity level of the VRP and its wide applicability to real life situations, the most successful methods in solving VRP and its extensions are heuristic methods which are described in [5] like Constructive Heuristics, GRASP [7] , Tabu Search [6] and Ant Colony System [8] .

To our knowledge, distributed methods are also applied to the general VRP we can site two main systems:

- MAS-Mars System : Proposed by Fischer et al [3], and based on Contract Net Protocol [3]. This system is a generic model representing the transport system in general.It is dedicated to chipping companies.The vehicle routing problem is considered as a particular instance of this system.
- TELETRUCK System : Based on Holonic Multi-Agent Systems and proposed by Burckert et al in [11]. It is an extension of the MAS-MARS system. In this system, physical objects (trucks, containers, trailers ...) are modeled as basic agents. These agents must work together in groups called holons (natural or artificial structure that is stable and consistent which may be made by other holons). A holonic system is characterized by a level of cooperation and/or collaboration more advanced than the standard SMA.

Another known technique in the field of the MAS was introduced to the resolution of the VRP: Coalition formation. In [9], Kefi et al propose a model (Coal-VRP) based on the formation of coalitions for solving the PTV with time windows. They distinguish two classes of agents: interface agent and control agent. The Coal-VRP has been validated against the Solomon's benchmark. This model has been improved by Boudali et al [10]. They proposed an extension to Coal-VRP, denoted DyCoal-VRP based on dynamic coalition's generation, to overcome Coal-VRP spatial and temporal complexity.

By studying the literature of the containers transport problem we noted three interesting findings:

1) The majority of papers dealing with transport problem in container terminals, propose agent-based Decision Support Systems (DSS) to optimize the whole container terminal operations [14, 15, 16], besides this few works do not provide any details of how agents are proceeding or any experimental results.

Table 3: Resolution methods applied to container transport problems

Author	VRP Version	Applied Method	Objective Function
[NIP05]	VRPPD	Genetic algorithm	Minimize the used vehicles and the traveled distance
[S06]	Asymetric-VRP	Modelization via MRPP Resolution via B&B and column generation	Minimizing the used vehicles and the traveled distance
[FZS07]	Asymetric-VRP	Variable radius and Greedy Randomized Procedure	Minimizing the used vehicles and the traveled distance
[MSWZ08]	VRP-TWPD	Modeling : MIP Resolution : MAS	Minimizing the duration of useless displacements
[BGLO09]	DVRP-TWPD	Distributed ant colonies system	Minimising the number of used straddle carriers.

2) Other works focus on simulating the CT's operations. The implemented simulators are used to evaluate the robustness of operational policies in a CT [18, 19] and to help stakeholders and to enhances the decision making process via visualization tools [17].

3) Despite the similarity between the VRP and the containers transport problem in CT, few researchers have been interested to this issue, and those that have considered the container's transport problem as a VRP use essentially meta-heuristics and heuristics approaches.These works are recapitulated in Table 3.

To summarize, we notice that the distributed resolution of VRP has not been sufficiently and comprehensively surveyed and compared, especially VRP in container terminal ports. In this paper, we concentrate on the modeling and the resolution of RSRP in container port by analogy to VRP with Time window and pickup and delivery (VRPTWPD). To achieve this goal we propose a Multi-Agent model based on coalition formation.

4. Proposed model

This section is dedicated to the description of our Multi-Agent model denoted CA-RSRP. Though, we will briefly discuss and argument the choice of an agent-based method.

The problem of containers transport is naturally distributed, it also requires the use of multiple entities (trucks, containers, drivers, customers ...). Therefore, Multi-Agent systems are very well adapted and appropriate to apprehend such problem [3] . In addition, Multi-Agent systems inherit, from artificial intelligence domain, the benefit of symbolic processing (at the knowledge level), so the real-world entities are easily modeled and manipulated by agents.

The proposed model is based on a set of Container Agents (CA) which negotiate and collaborate in order to be assigned and satisfied by a vehicle. Container agents model the containers to transport from the quay to the storage area and vice versa. Each one is characterized by its type ("delivery" if the container must be transported from the quay to the storage area, and "pickup" inversely), its origin and destination, the time window in which it has to be carried. Each CA has a state that is updated during the solving process.

The CAs negotiate and proceed in two phases, the Pairs Formation and the Coalition Formation.

In the following subsections we are going to detail the phases of the CA-RSRP.

4.1 Definitions

Throughout the remainder of this document we will use the following terms

- Container agent (CA): an agent that models a container. We distinguish two types of containers:
  - Delivery Container (L): A container that should be transported from the quay to the storage area.
  - Pick up Container (C): A container that should be transported from the storage area to the quay.
- Status of CA: Throughout the resolution process, each CA goes through the following status:
  - Free: in the beginning of the Pairs Formation phase.
  - Partially assigned (PartA): when it forms a Pair with another CA.
  - Permanently Assigned (PermA): when it is assigned to a coalition.
- A Pair: A couple of two CA that will be served successively by the same vehicle. We distinguish two types of Pairs:
  - Heterogeneous Pair: The CAs forming the Pair have different types (Delivery or Pickup).
  - Homogeneous Pair: Both CAs forming the Pair have the same type.
- Coalition: A coalition is a group of one or more pairs of container agents, which can be transported by a single vehicle.

4.2 Pair Formation

In this first phase, each agent tries to get paired with the CA with which it is served at lower cost in terms of time.

Because the RS are initially on the quay (in the Depot) , it is more suitable to transport firstly a delivery CA so that the RS is not empty when it leaves the quay. Therefore the agent that initiates the pairs formation phase is a delivery CA. Thus, each delivery CA sends a request for pairs formation to all the agents.
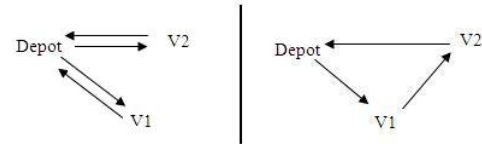

Fig. 4.1: Principle of Clarke and Wright saving algorithm

On receiving such a request, each agent calculates the time required, for a vehicle, to carry him in the same route with the proposal's transmitter. After receiving proposals, each delivery CA send an acceptance to the best proposal's sender (the one with which it is carried at the lower cost in terms of time). A refusal is sent to agents whose proposals do not meet the constraints of time windows and capacity. The selection of the best proposal is made by means of a heuristic inspired from the Clarke and Wright savings algorithm [4] (see Fig. 4.1). The agents have to confirm the proposals in the rest of the conversation. Interactions of the CAs in the pairs formation phase are described in Fig. 4.2.
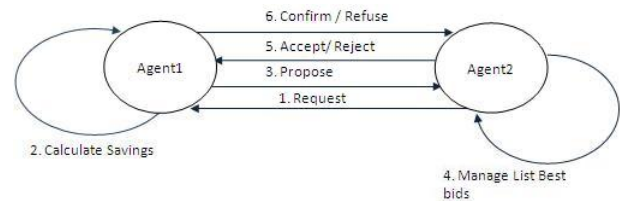

Fig. 4.2: Interaction between agents in the Pairs Formation phase

To summarize, the main idea of this phase is to take full advantage from each displacement of a vehicle. In fact, each container tries, through negotiation, to get paired with the container with who he causes the less empty time. It comes to minimize the duration of empty movements (when the vehicle is traveling without any load) to can decrease the overall service time and offer a better fleet utilization.

4.3 Coalition Formation

In the second phase, the pairs formed in the previous phase negotiate together in order to gather in larger coalitions while respecting the time windows and RS's capacity constraints.

When all CAs pass to the state "*PermA*" the coalition formation phase is initiated. Each CA is now characterized by its rank in the pair (first or second).

Agents ranking second in each pair are responsible for sending requests for coalition formation to the first ranking CA of the other pairs. These latter will calculate the cost of an eventual assignment with the request's sender, verify the constraints validity (relating to time windows and RS capacity) and send their proposals to the involved agents. Fig. 4.3 models the interaction of agents throughout the Coalitions Formation phase.

The Coalition Formation phase is inspired from the ECNP proposed in [3] and the agent communication is based on the FIPA ACL standard [13] .
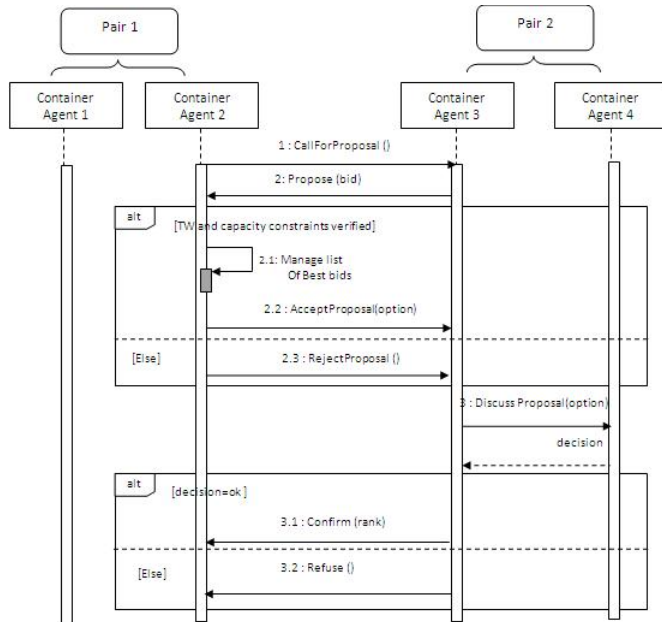

Fig. 4.3: Interaction between agents in the Coalition Formation phase

After receiving the proposals , each agent checks the capacity and time window constraints and save the coalition proposals ' senders in a sorted list according to the waiting and empty times. When all the proposals are received, the agent chooses the best bidder pair (top of the list) and behaves according to the different scenarios as follows:

1) The best bidder pair is assigned to a coalition and the receiver is not yet assigned : The receiver sends a *"Take Me with you"* message, it asks the best bidder pair if he can incorporate him in his same coalition.

2) The receiver is assigned to a coalition and the best bidder pair is not yet assigned : The receiver sends a *"Come with me"* message, it invites the best bidder to enter to his coalition.

3) Both the receiver and the best bidder pair are not assigned to a coalition : The receiver checks if there is a coalition not yet saturated and tries to fit into it. if not it creates sends a *"Create new"* message to inform the best bidder that he should create a new coalition and add him to it.

When receiving any of these messages(*"Take Me with you"*, *"Come with me"* or *"Create new"*), the agent examines its contents and decide what action to perform. Then it updates its status to *"PermA"* and sends a CONFIRM message to the issuing agent. When an agent Receives a CONFIRM message it updates its status to *"PermA"* and check if all other agents are in state *"PermA"*, in which case it sends

them alerts (INFORM message) to inform them of the termination of the coalition formation phase.

To summarize, this phase is based on the negotiation between the pairs formed in the previous phase. The CAs try to gather in coalitions by sending messages and using simple heuristics to choose the best bidders. New coalitions are formed only if a CA receives a *"Create new"* message and CAs are inserted in different existent coalitions by means of *"Take Me with you"* and *"Come with me"* messages. Capacity and time windows constraints must be validated before each one of these operations.

5. Computational results

In this section, we present the tests performed to validate the proposed model. The first subsection (5.1) describes the benchmark used and the second one (5.2) presents the results and some interpretations.

5.1 Test instances

To assess the performance of CA-RSRP so far, we choose a known benchmark developed by Mitrovic-Minic [2] . Nevertheless, we have modified it to fit our problem that considers real world features.

We add a column specifying the type (Delivery/Pick Up) of the container, and we discard the column of arrival time seen that we assume that the containers' arrival times are all known in advance. The type of each container is determined depending on its departure and destination locations with respect to the depot location.The benchmark defines for each container its starting point and destination coordinates, its time window and its type (Pickup/delivery). The tests are launched 10 times for each instance, and the results shown in the next subsection represent the average of the 10 tests values.

5.2 Results

In order to validate the results we defined two measures:

- The Activity ratio:It represents the percentage of time in which the vehicle is not idle. It's formulated as follow:

$$Activity\ ratio = 1 - \frac{Vehicle's\ waiting\ time}{Vehicle's\ service\ time}$$

- The Efficiency ratio: as one of the objectives that we have set is the minimization of the empty movements, it represents the percentage of time in which the vehicle travels while it is loaded. The Efficiency ratio is formulated as follow:

$$Efficiency\ ratio = \frac{Vehicle's\ empty\ time}{Vehicle's\ service\ time}$$

In Fig. 5.1 we compare the evolution of the number of vehicles required to serve 25 and 50 containers.We note that although the containers number is doubled, the number

of vehicles is increased (5 trucks for 25 containers and 7 trucks for 50 containers). We think that this number can be reduced if we enhance the model through merging the formed coalitions with respect to the capacity and time windows constraints. However, these changes can decrease the efficiency and the Activity ratios if coalition merging generates additional waiting and empty times.
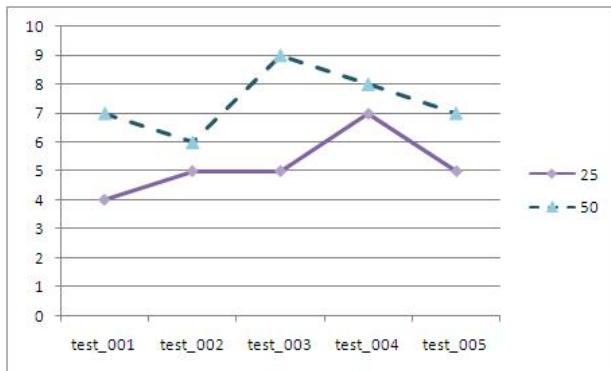

Fig. 5.1: Number of required vehicles

As shown in Fig. 5.2, the efficiency ratio is over the 0.6 for all the instances. Moreover, when the number of containers increases, this ratio is slightly improved ($\simeq 0.75$). A good Efficiency ratio evinces a good fleet exploitation.
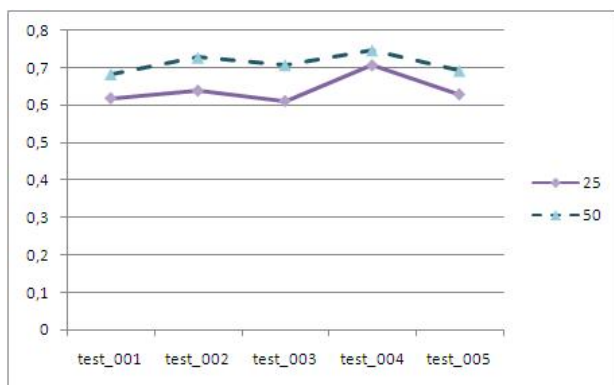

Fig. 5.2: Evolution of the Efficiency ratio

Although the values of the Activity ratio are declined, when increasing the number of containers, they are still encouraging. In fact, this ratio is always between 0.8 and 0.9 for the 25 containers and between 0.7 and 0.8 for 50 containers, as it is exhibited in fig. 5.3.
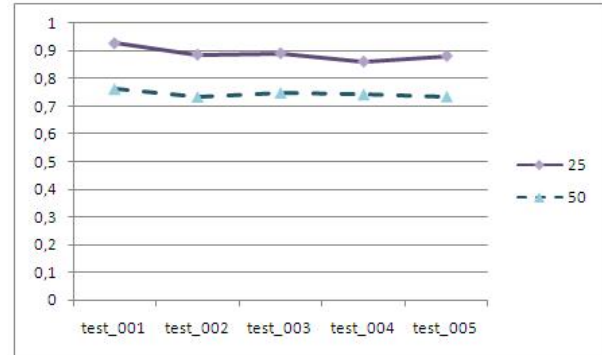

Fig. 5.3: Evolution of the activity ratio

6. Conclusion

Throughout this article we presented and formulated the RSRP as a VRP with time windows and pickup and delivery. Firstly, we have drawn an analogy between these two problems, then we presented a state of the art of agent-based methods tackling to VRP in a container terminal.

To solve this problem, we proposed an approximate Multi-agent model CA-RSRP, based on the coalitions formation and implementing simple heuristics.

To validate our resolution approach, we provide empirical study using Mitrovic-Minic benchmarks dedicated to the VRPTWPD. The results of this study are very promising. They show a high efficiency ratio and a reasonable number of required vehicles.

References

[1] Vis, I.F.A., De Koster, R. Transshipment of containers at a container terminal : an overview, European Journal of Operational Research 147(1), 1-6, 2003

[2] S. Mitrovic-Minic, R. Krishnamurti and G. Laporte, Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. Transportation Research 38 (pp 669-685), 2004

[3] K. Fischer, J. P. Muller, M. Pischel, and D. Schier. A model for cooperative transportation scheduling. In Proc. of the 1st Int. Conf. on Multiagent Systems, pages 109-116, Menlo park, California, 1995. AAAI Press/MIT Press.

[4] Clarke, G. and Wright, J.W.: Scheduling of Vehicles from a Central Depot to a Number of Delivery Points , Operations Research, Vol. 12, 1964, pp. 568-581.

[5] Gilbert Laporte. The Vehicle Routing Problem : An overview of exact and approximate algorithms. European Journal of Operational Research 59 (1992) 345- 358.

[6] Semet, F. et E. Taillard, 1993. Solving real-life vehicle routing problems efficiently using tabu search. Annals of Operations Research, 41 :469-488.

[7] C. Carreto and B. Baker (2002) A GRASP interactive approach to the vehicle routing problem with backhauls. In: C.C. Ribeiro and P. Hansen (eds.), Essays and Surveys in Metaheuristics. Kluwer Academic Publishers, pp. 185-199.

[8] R. Montemanni, L. M. Gambardella, A. E. Rizzoli and A. V. Donati, Ant Colony System for a Dynamic Vehicle Routing Problem, Journal of Combinatorial Optimization, Vol. 10, No. 4, 2005, pp. 327-343.

[9] M. Kefi, K.Ghédira. A Multi-Agent Model for a Vehicle Routing Problem with ime Windows , Urban Transport Conference, Dresden-Allemagne ,2004.

[10] I. Boudali ,W. Fki ,K. Ghedira. An interactive distributed approach for the VRP with time window. Journal of Simulation Systems, Science and Technology,2005.

[11] H.J. Burckert, K.Fischer, G.Vierke. TeleTruck: A Holonic Fleet Management System.In Proceedings of the 14th European Meeting on Cybernetics and Systems Research, Vol. 2, pp. 695-700, 1998.

[12] Cordeau, J.F., Laporte, G., Savelsbergh, M.W.P., and Vigo, D, Vehicle Routing. In: Barnhart, C., Laporte, G. (eds.), Handbooks in OR and MS, Vol. 14, Transport. North Holland, 367-428. 2007.

[13] N-T. O'Brien, P and Nicol, R, FIPA - Towards a Standard for Software Agents. In: BT Technology Journal, Vol.16:3, pages 51-59, 1998.

[14] T. Thurston, H. Hu, Distributed agent architecture for port automation. In: Proceedings of the 26th annual international computer software and applications conference (COMPSAC'02), Oxford, August 26-29. IEEE Computer Society, Los Alamitos, pp81-87, 2002.

[15] Y. Peng, J. Sun, Agent Based Container Terminal Optimization, pp.607-609, 2009 IITA International Conference on Control, Automation and Systems Engineering , 2009

[16] C. Carrascosa, M. Rebollo, V. Julian, V. Botti. A MAS approach for port container terminal management: the transtainer agent. In: Actas de SCI'01, pp 1-5. International Institute of Informatics and Systemics, Orlando, FL, 2001.

[17] W. Bockstael-Blok, I. Mayer, and E. Valentin, Supporting the Design of an Inland Container Terminal through Visualization, Simulation, and Gaming, presented at 36th Hawaii International Conference on Systems Sciences (HICSS'03), Hawaii, US, 2002.

[18] M. Rida, A. Boulmakoul, and R. Laurini, "Object Oriented Approach and Java-based Distributed Simulation for Container Terminal Operational Management," presented at International Conference on Information Systems and Engineering (ISE/SCS 2003), Wyndham Hotel Montreal, Canada, 2003.

[19] L. Henesey, P. Davidsson, and J. A. Persson, Evaluating Container Terminal Transhipment Operational Policies: An Agent-Based Simulation Approach, WSEAS Transactions on Computers, vol. 5, pp. 2090-2098, 2006.

[20] S.N.Kumar and R.Panneerselvam, A Survey on the Vehicle Routing Problem and Its Variants. Journal of Intelligent Information Management, Vol 4, 66-74, 2012.

Adjustable Group Behavior of Agents
in Action-based Games

Westphal, Keith and Mclaughlan, Brian

Kwestp02@uafortsmith.edu,

brian.mclaughlan@uafs.edu

Department of Computer and Information
Sciences

University of Arkansas - Fort Smith

Abstract *–* **Non-player characters (NPCs) within a single-player action-based game have historically been easy to kill and often do not work as a team to overwhelm the players. In addition, they generally do not adjust strategies based on the player's behavior. This research proposes that these agents should have adjustable behavior parameters that allow them to adapt to the behaviors of their opponents. In particular, this research proposes these NPC's have an adjustable aggressiveness towards the player giving them an option on whether to hide or to attack. Also, it examines the utility of agents that attempt to attack together and work in a group using simple decision making rather than being individuals. We developed a simulation that gathers information regarding the win percentages of teams of agents that have various behavior combinations versus different types of opponents. Behavior parameters examined include individuality versus teamwork and aggressive posture versus defensive posture. Results showed that certain behavior combinations have significant utility when the behavior of the opposing team is identified.**

Index Terms *–* **Artificial Intelligence, Agent, NPC, Group decision making, MMO (Massively Multiplayer Online Games), and game loop**

I. Introduction

AI-controlled non-player characters (NPCs) are the basis for opposition in single-player and cooperative team video games. These NPCs have historically been easy to destroy and often do not work as a team to overwhelm the players. In addition, they generally do not adjust strategies based on player behavior. When a skilled player plays against NPCs, the game can become rather easy.

Fortunately, computer opponents have evolved from the simple behaviors found in games such as Pac Man and Pong into the realistic war games of today such as Killzone, Modern Warfare, and the Battlefield games. With these recent increases in AI technology in video games, the AI of a game has become a significant selling point. Modern game reviewers now keep a place on their notepad just for rating the AI in these games, and when many players hear that the game has a great AI, they are more likely to want to play it.

Unfortunately, most of a game's AI resources are focused on individual NPC tactics [1]. In the latest generation of video games, the AI controlling the NPCs will hide if shot at, will also run away from grenades, and will look for advantages locations. However, they are far from acting as a team. Some of the best games out do have communication between their AI agents to assist in making the game more difficult for the player. Nevertheless, communication alone is still rather far from group decision-making.

In action-based games such as FPS games, going solo is almost guaranteed to get you killed being as the best you can hope for is a 1-on-1 match up. However when running in groups, agents have a much better chance to come across match ups that are more in their favor. This paper postulates that coordinating the group behavior and adjusting it based upon the behaviors of the opposing force can lead to increased success for the NPC team and a more rewarding play experience for the players.

Currently, the primary work being done in group decision-making in the field of Artificial Intelligence is in the game of football (soccer in USA) [2]. Being a complex sport with many rules, this research is a great starting point within the field of group decision-making. However in the case of action games where destruction is very possible, it is not the best starting platform.

There have been a number of approaches to this topic. Semsar-Kazerooni proposes a game theory approach to team cooperation, particularly team grouping [3]. Abraham has shown research into NPC team-mates

that assist the players [4].   In addition, psychological research examines the basis for human grouping and movement [5].

II. Approach

In order to test these hypotheses, two teams of five agents are placed on a map containing three bases as shown in figure [1]. The goal of the game is to capture all three bases. Alternatively, a team can win by eliminating all of the opposing team, as it is assumed the remaining team would then be able to freely capture all the bases.
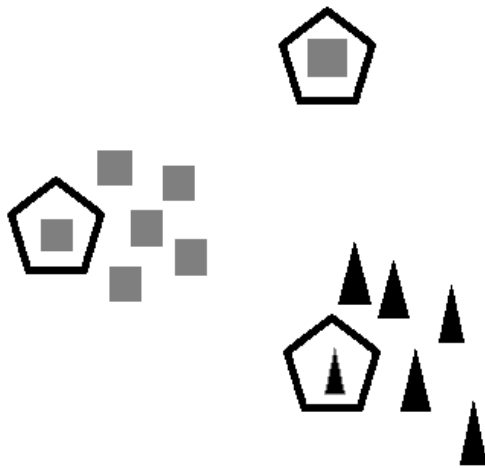

Figure 1: A typical game setup

In this game framework, various combinations of grouping and aggression are assigned to each team. Agents on a team can tend to group together or spread apart. They can tend towards making risky moves or attempt to remain in the relative safety of a base. Finally, when given a choice, they can focus more on capturing bases or eliminating enemies.

III. The Agents

Agents have multiple attributes that allow them to persist and interact within the game world. During the development of the agents we wanted to make sure that the agents were as individualistic and anonymous as possible to ensure that any results were affected by only changes within the agent itself. These attributes include:

1. Aggressiveness towards bases versus agents
2. How aggressive the agent is towards its target
3. Minimum and maximum distance to other teammates

Any of these attributes when changed will affect the outcome of each simulation.   The agents' two aggressive targets, bases and agents, make the agent choose a particular type of target with a much higher certainty.   The minimum distance and the maximum distance restrict each agent's movement to a certain area around other agents of its own team. This facilitates our version of grouping.

In addition, the agents have an armor rating, an attack rating, and a sighting distance. These values affect the agents' abilities to take damage, to deal damage, and to see targets. These values are not varied between agents in these experiments.

A.   Agent Decisions

The agents group together by continually staying at a certain distance to and from the other agents in its group. When agents are set to stay in a group the maximum distance is set to 5 while the minimum distance is set to 3 which will restrict the agents from moving to far from each other. However when the agents are set to be non-grouping their maximum distance is set to be 25 with the minimum distance staying at 3. Having their maximum distance set to 25 allows the agents to move freely throughout the map as individuals.

The aggression of the agents on each team is based off of an algorithm that determines where the agent will attempt to move. This algorithm is based on the notion of *temperature*. Agents in the system are attracted to higher temperatures. Each base has a temperature that can fluctuate depending on the amount of activity at the base. Each agent's temperature is based on its health. As each of the agents is injured and has its health reduced its temperature goes up drawing other agents towards it. The aggression algorithm is based off of the temperatures of both the agent and the bases. The equation becomes the temperature divided by the distance plus the agents corresponding aggression

$$(temperature/distance)+Aggression$$

and the object that has the highest value becomes the agent's target.

If an agent is deemed to be aggressive towards a certain object this value is increased by 1 if the target that the agent is evaluating is one of these objects. When an agent is aggressive, it is only towards agents on the other team and bases owned by the other team. If an agent is base aggressive then it is not only aggressive towards bases of the other team but also ones that are not owned by any team. In this context we deemed that an

Aggression constant of 1 as an appropriate number as it will not cause an agent to give up a very good opportunity to attack an object that it is not aggressive toward but is enough to cause it to move toward its aggressed target most often. If this algorithm were to be moved to a larger map with different temperatures this additional number would need to be changed to suit the environment. A typical aggression scenario is shown in figure [2].
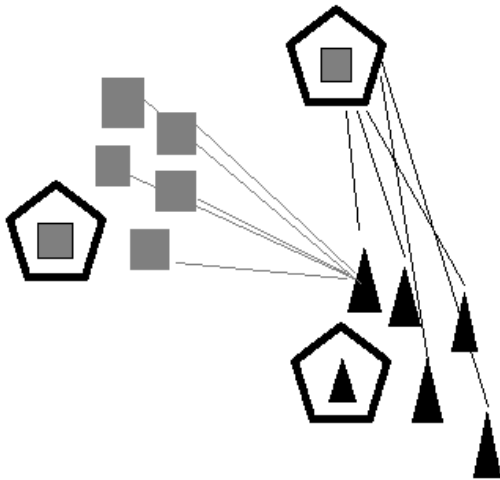
Figure 2 [In this figure the square team is agent aggressive while the triangle team is base aggressive.]

B. Agent Actions

Movement – the agents have two main functions allowing movement, a rotate towards function and a move forward function. Each agent plans its movement, and then all agents move at the same time, preventing either team from benefiting from additional knowledge.

Attack – When an agent comes within another agent's radius these agents are allowed to attack one another. This ability to attack other agents within the environment is the key ingredient to any action based game. When an agent attacks another agent, it only gets to attack one time toward one agent on each trip through the game loop. When each agent attacks, it removes the amount of attack it has from the enemy agent's armor amount.

Take a base – As an agent moves around the map it will attempt to move toward agents and bases. When an agent comes within a bases radius it will attempt to take the base. When an agent takes a base, it removes a certain amount of temperature from the adjacent base.

IV. Bases

Bases represent another key attribute to many action based games. Bases create "hot" zones that tend to

draw agents and players alike, together. In MMOs (Massively Multiplayer Online) and FPS games, bases are used to increase the amount of people in a fight creating a much higher pace of action throughout the game. Without anything to draw players and agents together the fighting becomes slow and there is a great reduction in the amount of fighting, thus reducing the appeal that the game has.

Along with the agents, the bases also have attributes that affect the outcome of each simulation. These attributes include:

1. Radius
2. Temperature
3. Owner

The radius of the base defines how far away an agent has to be to be able to capture the base. When a base's temperature reaches 0 the base is considered taken and its owner becomes the team number of the agent that captured the base. When all three bases have the same owner the corresponding team wins. In order to draw agents back to that base the base's temperature is increased by 1 each time the game loop progresses.

The initial map set up included five agents on each team placed just on the outside of the bases on the right side of the map (Figure [1]). This style of map placement is traditional in both MMO style and also online FPS games during any sort of capture the base style gameplay. The primary reason for placing the bases like this is that it gives both teams easy access to a base and then equal access to the third base. This placement promotes a fair playing field to make sure that the map placement of the bases did not affect the outcome of the simulations. To reinsure that this map placement did not affect the outcome of our simulations we rearrange the placement of the agents giving each team a chance to start from either the top or bottom right bases. This change produced near exactly the same results allowing for the conclusion that the map has no effect on the results.

V. Data

Each of the scenarios is divided into wins via killing off the other team and via capturing all of the bases. All of the tested variables did make a significant impact on the outcome of the scenarios. In general when each team was given a more aggressive tendency towards a certain object, they tended to win according to the winning scenario of that object more often. Figure [3] depicts the wins based on Team 1 as base aggressive and team 0 as base aggressive and both teams grouping.
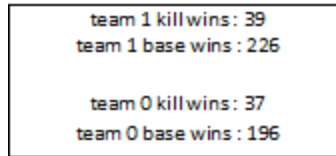
```
team 1 kill wins : 39
team 1 base wins : 226

team 0 kill wins : 37
team 0 base wins : 196
```

Figure 3 [wins for teams that are grouping with aggressiveness towards bases]

When the teams are more aggressive towards bases they tend to capture bases more often than kill off the other team, the same can be said about when agents are more aggressive towards other agents. This tendency proves to us that the attributes in our agent when changed are having an effect on the outcome of our results. According to our results both base and agent aggressiveness improves the teams' chance of survival and victory. In general when aggressive agents play against non-aggressive agents the aggressive agents win more than 300 times out of 500.
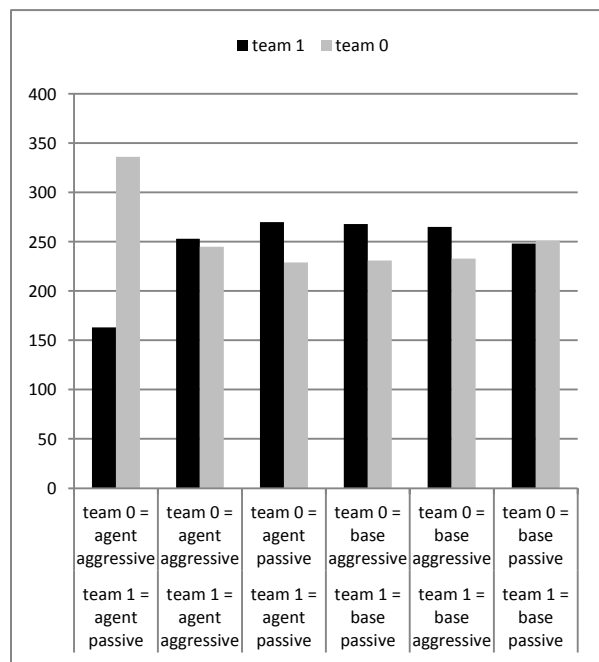

Figure 4 [both teams grouping]

In the starting hypothesis we decided that due to when agents spread out they tend to attack as solo agents and that each solo agent will become easily overwhelmed by the group of agents. When the simulation was run, the data depicted in figure [6] on almost every account showed this exact result. The agents that were grouping won more often than the agents that were non-grouping. The only time that the non-grouping team won was when the non-grouping team was agent aggressive and the

grouping team was agent passive. Thus pointing towards our hypothesis that it is good to stick together

For each different set of attributes the simulation was run 500 times in order to eliminate any outliers within the series. The attributes were tested with both teams set as grouping as in figure [4], both teams as non-grouping as in figure [5], and with one team as grouping and the other as non-grouping as in figure [6] (each of these figures depicts wins for each team pertaining to the scenario).

In figure [4] the general outline of the data is even across the board besides when placing an agent aggressive team against a non-aggressive team. We believe that the reason for this is when a team groups up their ability to make it to all of the bases is restricted because the agents pull the whole team to one base and then another and can't attack multiple bases at a time thus making it an even playing field for base wins. However when the teams group up they are able to kill other teams of agents very efficiently, thus increasing their ability to win via killing the agents on the other team. The difference in the amount of wins when run with two different kinds of agent aggressiveness can be attributed to the fact that our agents have no survival instincts, meaning that if a team that is agent passive turns towards a base it may decide to move towards the base rather than returning fire. This would allow the agents that are aggressive to kill the passive agents with no resistance.
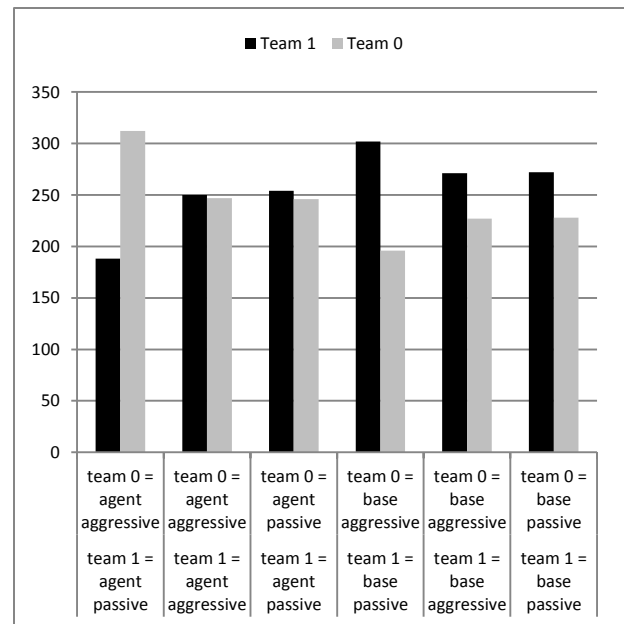

Figure 5 [both teams non-grouping]

In figure [5] the general outline of the data differs quite a lot from figure [4] there are 2 extremes in this data rather than just the one in figure [4] and, those are when you change both agent aggressiveness and base aggressiveness. We believe that the change in agent aggressiveness is due to the same reason as when teams group up, being that the passive agents tend to allow the aggressive agents kill them, however when changing the base aggressiveness in non-grouping agents we see a large increase in wins. We believe that this is due to the fact that when non-grouping agents are base aggressive they divide and conquer and are able to take multiple bases at a time rather than just one, this conclusion is based off of the fact that in our data the teams win by bases almost 2 to 1 compared to winning with kills.

In figure [6] we see the primary data, what happens when you change the way the teams group up and also their aggressiveness. We see a dramatic change in the amount of wins for the team that groups up. Our data shows that these extra wins are nearly always wins from killing the other team. We believe that this is due to the effect of swarming each agent that strays from its group is quickly overwhelmed by the team of agents that is grouped up.
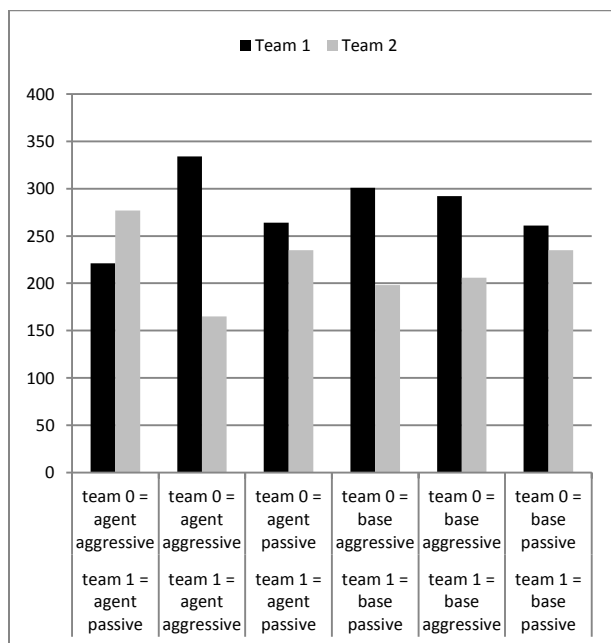
Figure 6 [Team 1 as grouping – Team 0 as non-grouping]

VI. Future Work

This project given the time limitations was not able to fulfill all of the desired outcomes. This project can take many new directions if given more work and time.

1. *Advanced group decision making*. At this point, only basic decisions are being made. It is possible that complex decisions such as having some team members assuming aggressive behavior while other team members remain behind to guard bases could yield interesting results.
2. *Complex survival instincts*. In their current form, more passive agents do not appear to have any benefits. If agents could choose between aggressively pursuing a target versus taking cover or moving out of field of fire, survivability could be increased.
3. *Enemy strategy discovery.* Further work needs to be done to allow the team to determine which combination of parameters the enemy team is adopting in order to counter that strategy.

VII. References

[1] Beij, A., and van der Sterren, W. (2005). "Killzone's AI: Dynamic Procedural Tactics" Game Developer Conference, March 10, 2005.

[2] Semsar-Kazerooni, S. (June, 2009). *A game theory approach to multi-agent team cooperation*.

[3] Abraham, Aswin T. "AI for Dynamic Team- mate Adaptation in Games." (2010): n. pag.*IEEE*. Web. 4 Jan. 2013.

[4] Ruiz, Myriam A. "Team Agent Behavior Architecture in Robot Soccer." (n.d.): n. pag. *IEEE*. Web. 4 Jan. 2013.

[5] Derks, Belle. "Working for the Self or Working for the Group: How Self- Versus Group Affirmation Affects Collective Behavior in Low-Status Groups." *Journal of Personality and Social Psychology* 96.1 (2009): 183-202. *IEEE*. Web. 4 Jan. 2013.

An Agent-Based Intentional

Multi-Robot Task Allocation Framework

Savas Ozturk[1], Ahmet Emin Kuzucuoglu[2]
[1]TUBITAK BILGEM, Gebze, Kocaeli, Turkey
[2]Department of Computer and Control Education, Marmara University, İstanbul, Turkey

Abstract - *When multiple robots are supposed to operate together, coordination and communication issues arise. "Which robot should execute which task?" is the key question of Multi-Robot Task Allocation (MRTA) problem. Among several autonomy level approaches being centralized or distributed, hybrid approaches have become popular at reaching optimal task allocation combining advantages of both centralized and distributed approaches. Hybrid approaches like auction-based or market-based are also named as intentional according to their cooperation model. Intentional type allocation model differs from swarm-type basically for number of robots employed. This study presents development of an intentional MRTA model based on multi-agent concepts using Java Agent Development Environment (JADE). A simple model is built and proposed on prior approaches in market-based task allocation in order to improve optimality. Experiments are performed using selected cases of MRTA literature and results prove that proposed model in this study produces optimal solution at each case.*

Keywords: Multi-Robot Task Allocation(MRTA), market-based, intentional, auction, multi-agent, Java Agent Development Environment (JADE)

1 Introduction

MRTA problem has become a key research topic in the field of distributed multi-robot coordination in recent years. The goal of MRTA is to choose on time a proper action for every robot to ensure that they can effectively complete all the tasks. Supposing that any individual of the system cannot complete a complicated problem, this problem can be decomposed to the series of sub-problems on some principles and these sub-problems can distributed to corresponding individual to solve the problem. Presently, there are many methods for task allocation grouped in three main approaches: fully centralized, fully distributed and hybrid approaches.

A centralized task allocation approach used for multi-robot system suffers from limitations in both system scalability and robustness. If the central robot fails, the whole system fails. On the other hand, a fully distributed system produces suboptimal solution. Hybrid approaches combine both centralized and distributed approaches combining their advantages. Each individual in the system works for its own plan and benefit and at the end global benefit is emerged. Market-based approaches are basically built on buying and selling goods to the planner or other robots.

Another classification about multi-robot task allocation is based on cooperation which groups task allocation types as swarm-type or intentional-type [1]. Swarm-type task allocation is performed by large number of robots and it depends on fully distributed approach. Intentional-type task allocation is either auction-based or market-based. This study mainly investigates intentional task allocation mechanisms and builds a taxonomy including the related literature and proposed model. Proposed model is a mix of prior techniques and includes some improvements, finally a novel MRTA task allocation mechanism is built which an objective as minimizing the overall has cost of given task allocation problems.

Due to some difficulties of procurement and execution of physical robots, agents are reckoned as real robots and experiments are carried out. Some of the most known MRTA techniques have been implemented at JADE [2]. A framework has been built similar to Contract Net Protocol (CNP) [3], MURDOCH [4] and TraderBots [5]. Despite the similarities between multi-agent and multi-robot systems, the transition from agents to robots is not straightforward. Some assumptions are made under some constraints. Optimization problem is focused instead of localization, mapping, sensory or other multi-robot concerns.

To summarize the contributions, this study has three main challenges: An agent-based system development study specified to multi-robot domain, a practical taxonomy compilation and an optimization study that minimizes overall cost by using hybrid techniques.

Section 2 of this study summarizes related literature. At section 3 it is told why JADE is chosen, how it is used and development of agent-based systems. Section 4 defines the MRTA problem and provides proposed model with comparisons to similar models including a novel taxonomy. Results on the experiments are discussed in Section 5 and study is concluded in Section 6.

2   Related Work

For the last decade, the number MRTA studies have been increasing year by year in order to handle the growing demands. Unmanned air and underwater vehicles, exploration, Robocup, security, disaster response, intelligent house, assembly lines and factories   are some of the recent examples employed MRTA principles. MRTA studies have been classified according to autonomy level.

Centralized approaches employ optimal assignment and Hungarian algorithm [6][7]. The Hungarian algorithm allows a "minimum matching" to be found. This can be used in instances where there are multiple quotes for a group of activities and each activity must be done by a different agent, to find the minimum cost to complete all of the activities. This method helps finding optimum solution but as activities and agents are increased, it turns to be solved in a long time ($O(n^3)$). The Hungarian algorithm is provably complete and optimal. A central planner is employed which has all the information about the environment (robots, tasks, map and the obstacles). Planner makes calculations and assigns tasks to each robot. Task execution agenda of each robot can be calculated and sent by central planner as well. At this type of autonomy level; robot does exactly whatever planner wants.

On the other hand, distributed approaches give all decision responsibility to the robots. To give an example, some models are based on emergent behaviour like swarm systems which robots don't work explicitly together [8]. Robots don't need to obey a planner, they use their nearby information. Local optima can be succeeded but it is hard to reach global optimum.

Some models depend on intentional behaviour which robots cooperate for a common purpose [1]. Intentional task allocation can be executed either auction-based [4] or market-based [5]. The nuance between both is represented at Fig. 1. Planner can be a central agent like a computer or it can be any of the robots.

a)   b)
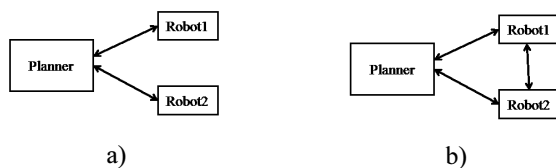
Fig. 1 Intentional task allocation; a) auction-based b) market-based

Examples to intentional task allocation have been increasingly appeared since the end of 90's. BLE (Broadcast of Local Eligibility) system uses a Publish/Subscribe method to allocate tasks that are hierarchically distributed [9]. M+ proposes a scheme for negotiated task allocation, it allows for dynamic task reallocation of subcomponents of complex tasks [10]. ALLIANCE is developed for robust, fault tolerant, fully distributed heterogeneous multi-robot systems. As there is no negotiation process, communication cost is low [1]. Gerkey and Mataric published a taxonomy study and presented a

framework based on Publish/Subscribe and first price auction mechanism, MURDOCH. According to taxonomy, robots are classified as single-task robots (ST) and multi-task robots (MT) and tasks are classified as single-robot tasks (SR) and multi-robot tasks (MR) [4][11]. TraderBots is the first introduced market-based MRTA study in the literature. Robots exchange their tasks if they negotiate. An individual seeks for its own benefit and then optimum result is reached. Market economy principles can be applied to multi robot systems according to this approach [5]. The most comprehensive study on task structures are held by Zlot which deals with complex tasks built on task trees and how to decompose them [12]. As an extension to TraderBots, Kalra proposed HOPTILES for tightly coupled robots [13]. Aforementioned approaches are based on single robot tasks; however some researchers attacked multi-robot task problems. Vig and Adams coined the term "multi robot coalition formation (MRCF)" [14]. MRCF is allocating multi-robot teams to different tasks (ST-MR problem). Tang and Parker proposed a model named ASyMTRe that enables switching between single robot tasks and multi-robot tasks dynamically [15]. Hasgul et al. applied project management principles for task scheduling to multi-robots successfully [16]. Kaleci et al. combines market-based approach and optimal assignment [17].

This study is focused especially on CNP, MURDOCH and TraderBots among all these approaches. CNP is also a de facto protocol of FIPA [18] and closely related to agent world. MURDOCH is examined as an auction based model and TraderBots as a market based model.

3   Multi-Agent Systems and JADE

Multi-robot and multi-agent research areas include similar concepts, sometimes robot experiments are performed in agent domain. Therefore at the level of explicit coordination among multiple individuals, the differences in techniques used by multi-agent and multi-robot systems are in fact very few [4]. Although robotics researchers employ sophisticated techniques while designing single robot control systems, they have tended to use techniques that are already well known in the agent community when designing explicitly coordinated multi-robot systems. Although it has some constraints and fidelity problems, agent are advantageous at developing large teams of robots, scalability, cost, development time, demonstration, flexibility. One can easily develop a robot-like agent and integrate it any agent system that is FIPA-compliant. There are several agent development environments which can be classified according to their purpose. RETSINA [19], SEAGENT [20] and JADE [2] are presented as general purpose ones whereas CONSENSUS [21] and iJADE [22] are developed for a specific purpose.

JADE is selected as an agent development environment among several agent development framework alternatives taking into below factors account:
- FIPA compliance
- Development environment (Eclipse support)
- Comprehensive documentary

- Technical support and live forum
- Amount of developers worldwide

Messaging system and behaviours of JADE are frequently used in this study. It is quite easy to send and receive messages. Messages are queued in a message list and executed by order. JADE encapsulates developer from the complexity of communication; there is no need for an expertise at TCP-IP or sockets.

JADE behaviours are easily developed as well. JADE isolates developers from parallel programming and threading issues by bringing behaviours into use. Examples to some JADE behaviour are OneShot, Cyclic, Ticker, Waker, Simple, Sequential, and Parallel. The only requirement to use behaviours efficiently is to build a feasible schedule. Activity design is an important phase of agent development.

Agents can be initiated manually by executing its code, or dynamically by calling from another agent. JADE agents are FIPA-compliant, to this end, these agents provide interoperability. It is possible to join a JADE agent container using not only computers but also mobile phones or tablets from any geographical location that has internet connection.

Four types of agents have been developed for this study:

Robot Agent:

Robot Agent behaves like a autonomous real robot. It senses, decides and actuates. Obstacle sensing and avoidance have not been implemented yet, but robots are sensible to battery level and task announcements. For example, when battery level is low, it cannot bid to any tasks, it looks for a suitable charge station and goes to charging. Robot Agent is informed about map of the simulation world including walls, other robots and obstacles.

Auctioneer Agent:

Auctioneer Agent has a planner role. It is informed about all the tasks, charge stations and robots. When a robot is added to the system, Robot sends acknowledgement message to Auctioneer Agent in order to introduce itself. Auctioneer Agent is also responsible to help robots to find suitable charge station, to perform centralized optimal assignment and to inform graphical user interface agent.

Terrain Agent:

Terrain Agent shows the terrain, tasks and robots on a graphical user interface. Robots send their momentary information (location, budget, battery level, completed tasks, charge count) with messages where Auctioneer Agent sends task and charge station information.

SimManager Agent:

SimManager Agent is responsible for selecting terrain model, defining robots and tasks, Travelling Salesman Problem (TSP) type (tour or path), random seed for task creation and execution approach (Centralized, auction-based or market-based). Some information is read from configuration files. Creation and initialization of all other agents are triggered by SimManager Agent.

4   The Problem and Proposed Work

One of the optimization objectives at multi-robot task allocation is minimization of the cost, in other words maximizing the throughput as a definition used in industrial, manufacturing processes [23]. This study focuses on cost optimization of Multi-depot Travelling Salesman Path Problem (MD-TSPP) [5]. Differences of this problem from the well-known TSP are listed below:

a) There is only one salesman at TSP. MD-TSPP includes more than one salesman departing from different depots.

b) The initial city can be indefinite at TSP where departing depots of salesmen must be known at MD-TSPP.

c) TSP is often considered as a tour that salesman must return to its initial location. MD-TSPP is considered as a path, salesman stays at the last city visited waiting for a new task. (*Tour* means that the salesman must return to the starting city, *Path* means that the salesman will not return to the starting city, but just visiting each city once and only once.)

In this study, salesmen are the robots and tasks are the cities that robots must visit. The distance between two cities is the cost of travelling between two tasks. If a robot visits a city, that task is marked as completed. If all the tasks are completed by the robots, overall goal is accomplished. When the robots are not executing tasks, they remain stationary at their current locations.

Let c_{ij} be the cost for the jth robot to visit the ith city from the $(i-1)^{th}$ city in its tour (where the 0th city is the starting location) The robot cost function for the jth robot is computed as follows:

$$rcost\,(j) = \sum_{i=1}^{n_j} c_{ij} \qquad (1)$$

Where n_j is the number of cities in the tour for robot j. The team cost function is:

$$tcost = \sum_{j=1}^{m} rcost\,(j) \qquad (2)$$

where m is the number of robots [5].

This study differs from TraderBots that TraderBots approach condones the multiple visits for a city by a robot or different robots which means the global task is complete when all cities are visited by at least one robot. The approach in this study permits only one visit for each task considering that it would help minimizing the cost.

Another difference from TraderBots lays at the robot's situation when all of its tasks are completed. Robot makes a route plan when execution starts and this plan is dependent on its tasks. If it is wanted to return to its initial location, it must add that to its own route plan as an additional task.

The problem becomes more complicated if tasks are dynamically introduced into the system and the robots must reconfigure to execute the new tasks in real time.
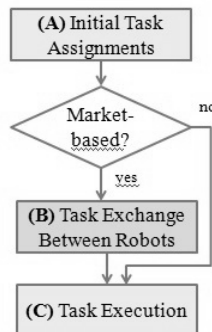

Fig. 2 Task allocation steps

Task allocation steps are examined in depth according to several nuances.

(A) INITIAL TASK ASSIGNMENTS

Once the system is started, tasks are assumed as unassigned. For a fast start-up, tasks can be allocated to the tasks in different ways. Most inconvenient but easy way is random allocation. This technique distracts reaching optimum solution. Greedy assignment is a feasible and practical solution that can fail in some circumstances. Optimal assignment is an expensive solution for initial assignments if followed with a trade process, but it eases finding better results. PRIM allocation, on the other hand, is proposed as an alternative to combinatorial assignment and it has the best score at cost/performance [24]. Random and greedy assignments may be preferred when large number of robots and tasks are used.

If random assignment is selected Tasks are assigned to robots using random number generator.

For the other assignment types, following procedure is applied:

At the moment simulation is started, a central planner (Auctioneer Agent) has all the tasks. It announces all the tasks to all the robots. Robot agents bid for each task. Bids are calculated according to metrics, in this study the only metric is distance (task to robot). Each robot is equipped with a map of the world, which enables it to calculate the cost associating with visiting each of the tasks. Thus, the robots bid for each task based on their estimated costs to visit that city.

Distance is computed according to Euclidian distance. If $u = (x_1, y_1)$ and $v = (x_2, y_2)$ are two points on the plane, their Euclidian distance is given by

$$d(u,v) = \sqrt{((x_1 - x_2)^2 + (y_1 - y_2)^2)} .\qquad (3)$$

Robot bids are recorded into a proposal matrix. When all the proposals are delivered to the Auctioneer Agent, it executes selected assignment algorithm (m: number of robots).

i. *Greedy:* Best matching **m** tasks are assigned to robots (one task for each robot). Shortly; m tasks are assigned to the their closest robots.

ii. *PRIM:* Best task is assigned to the robot with best offer. Shortly, a task that has lowest cumulative distance to the robots is assigned to the closest robot. Ony one task is assigned at each iteration.

iii. *Optimal:* Hungarian Algorithm is executed to make task assignments.

As soon as first assignment is completed, second assignment process (announcement + bidding + assignment) is executed. Assignments are done iteratively until all the tasks are owned by the robots. At each iteration, proposal matrix is cleared in order to catch changing conditions.

(B) TASK EXCHANGE BETWEEN ROBOT (TRADE PROCESS)

Initial task assignment is followed by a trading phase at market-based task allocation. But this process can be by-passed through task execution phase.

Role Assignment

Robots can bid for tasks, can announce tasks to sell or can do both of these at the same time concurrently. If all the robots try to sell all of their tasks at the same time, a complicated situation may be emerged.

After revising several studies, sequential objective-based seller robot employment model is preferred. At this model a robot is selected as a seller and others are informed to wait task offers. Robots define their most unsuitable tasks. Unsuitable task is worst task for the robot according to the cost. Robots ask themselves the question "How much my way is shortened if I get rid of this task?" for each task and the answer is sent to Auctioneer Agent. Auctioneer Agent assigns the robot which is candidate for maximum cost reduction as seller robot for the current trade.

Auction Type

Seller robots can announce for only one task, more than one task or combinatorial task bundles at each trade. Combinatorial trade requires much processor power, because all the task pair combinations should be taken into account [25].

In this study, single item-objective based method has been used which is related to proposed seller robot selection technique; unsuitable task is announced.

Bid Valuation and Auction Clearance

At a market, one has to protect its own profit. Each trade should satisfy both buyers and sellers, bids should be efficient. Bid can be calculated simply as Euclidian distance between the robot and the task. If number of tasks for a robot is lower than 20, TSP calculation can be used. Seller robot announces the task with the price as cost reduction assuming the task is

sold. Candidate robots calculate their cost increment assuming the task is bought. If amount of increment is lower than announced price, candidate robot bids for that. Fig. 3 illustrates that Euclidian distance calculation fails at some situations. WP2 (Way Point 2) is very close to Robot 2 but it is not reachable easily. To this end, TSP approach is suitable for this case.
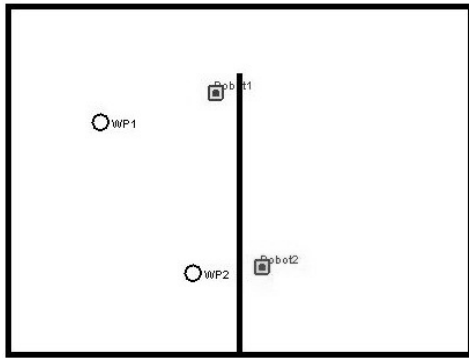
Fig. 3 A map to show an example to failure of direct Euclidian calculation

There are several bid valuation strategies in the literature as well. Joundeau et al. Used wavefront calculation [26]. Kaleci et al. has taken dynamical and sensor models of robots into account [17].

Auction usually results as selecting best bid owner as the winner. Accepting incoming bids and transition to auction clearance last until all bids are OK or until time is out.

Termination of Trading Phase

When market reaches to stability and there exists no other tasks to trade, robots should begin execution as soon as possible. Whether the system is stable or not can be decided after searching trade situations. If there is no trade executed for while, it means that there is no profit for a robot to buy or sell any tasks anymore.

(C) TASK EXECUTION

After all auctions and task exchanges between robots end, each robot has a task list to do. Tasks are executed one by one according to one of the below ways:

i. *Next task from the list:* Robot executes the task list from the beginning to the end without sorting.
ii. *Nearest task:* Robot executes the nearest task to its current location.
iii. *Schedule:* Robot creates a plan calculating either TSP path or TSP tour.

Some limitations and assumptions must be considered as well. Each items in the below list complicates the solution a bit more.

- Battery life, locations of charge stations and charging time
- Unknown or newly introduced tasks
- Unknown or newly introduced robots
- Failure of a robot or lost communication
- Obstacle avoidance

5   Experiments and Results

Proposed model is applied into a developed agent based simulation environment. This software is convenient to perform experiments with below variables :

- Terrain model (dimensions, walls, obstacles)
- Number of robots
- Number of tasks
- TSP type (tour or path)
- Bid Valuation (Euclidian or TSP)
- Intention type (With or without trade)

Examples of well-known MRTA studies in the literature are applied and results have shown that proposed model always reaches the optimal solution. Hungarian Algorithm is used for initial task assignments for all the experiments presented in this section. After initial assignments, four types of experiment configurations established:

- TSP tour with trade
- TSP tour without a trade
- TSP path with trade
- TSP path without a trade

The number at the mid-bottom of each figure represents the final total cost for the related experiment setup. Fig. 4 represents four snapshots of Zlot example therefore 470 is the total cost of the simulation of Fig4.a. Money unit in these trade examples is pixels. Dimension of each map is 400 * 300 pixels. There are three robots and 5 tasks to do for Zlot example. It can be seen that trading is more advantageous at TSP tour problems.
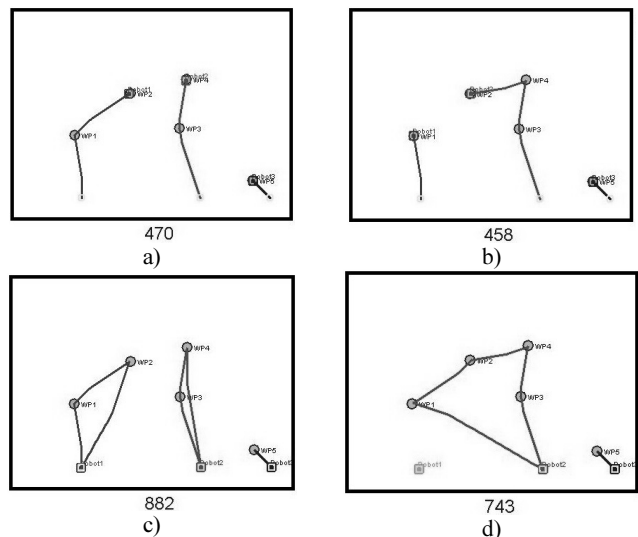
Fig. 4 Snapshots from the simulation: Zlot example [12] (a- TSP Path without a trade, b- TSP Path after trade, c- TSP tour without a trade, d- TSP tour after trade)

Fig. 5 illustrates an example from TraderBots warehouses. Robots exchange their tasks in order to achieve optimality. Although initial task assignments here are different from TraderBots, result is the same, optimal solution. This means that, proposed model fits for all similar problems.

2146
a)

1624
b)

Fig. 5 Snapshots from the simulation: TraderBots warehouse example (a- TSP Path without trade b- TSP Path with trade)

Table 2. Cost savings for examples

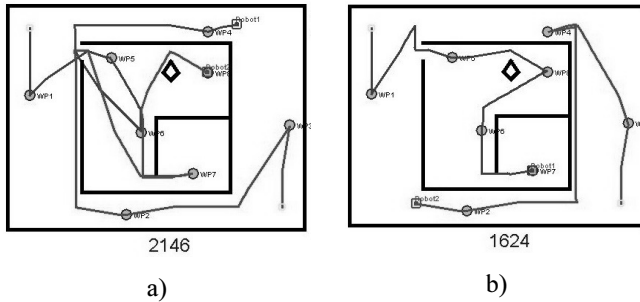| Example	Without trade	With trade	Saved (%)
Zlot – TSP Path	470	458	3
Zlot – TSP Tour	882	743	18
TraderBots	2146	1624	24
PRIM	252	146	42

Table 2 lists cost saving for each example experiment. It can be said that benefits of task exchange between robots incredibly increase when a) there are some obstacles in the area b) TSP tour approach is used. Also some experiments are performed with randomly distributed tasks and robots in an empty area (no obstacle).

As a final example, two rooms example can be shown (Fig. 6). It can be seen clearly from the figure what if market-based approach is not selected. Robot1 and Robot2 stay at their own rooms and they save 35% according to the total cost of their initial task assignments executed.

Experiment results clearly show that the combination of trade based approach with transition method between different regions using TSP gives promising results. Transition method should be improved by using legacy and novel motion planning techniques.

6 Conclusions

As a result of the focus on multi-robot systems, multi-robot coordination has received significant attention. In particular, MRTA has recently risen to prominence. The complexity of the multi-robot systems being studied has increased.

a)   1856

b)   1203
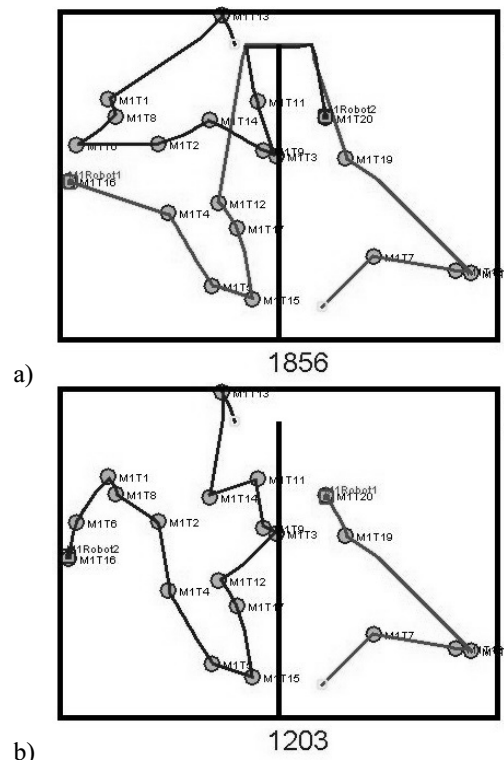
Fig. 6. Two doors example (a- without trade, b- with trade)

An agent-based MRTA model has been presented which is a hybrid approach combines trade-based task allocation, optimal assignment and TSP solution. This model produces more optimal results and suitable for larger teams and task collections as well. A simple way to select most unsuitable task for the robot has been presented.

Among several alternatives, JADE is chosen as development environment. CNP, MURDOCH and TraderBots principles are applied to the system. Tackling with delayed messages and conflicting behaviours, it is possible to implement a real-like multi-robot system with JADE. In this study, optimization strategy has been focused on instead of specific nature of multi-robot domain like sensory system, localization, mapping and etc.

Taxonomy is compiled including literature research and proposed techniques. Example cases of well-known multi-robot research are tested on the developed framework and it is seen that framework gives optimal results. Proposed model fits for all similar problems. Future work will include combining trading and execution phases to work concurrently and developing a fault-tolerant mixed environment (indoor/outdoor) multi-robot system. Improvement at efficient bidding and assignment strategies specialized to a warehouse system will be another research objective.

7   Acknowledgement

8   References

[1]   Parker, L.E., ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation. *IEEE Transactions on Robotics and Automation*, 14 (2), 220-240, 1998.

[2]   Bellifemine, F. L., Caire, G. and Greenwood, D., *Developing Multi-Agent Systems with JADE*. ISBN : 978-0-470-05747-6, Wiley, 2007.

[3]   Smith, R.G., The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers,* C-29(12):1104-1113, 1980.

[4]   Gerkey, B. P. and Mataric, M. J., A formal analysis and taxonomy of task allocation in multi-robot systems. *Journal of Robotics Research*, vol. 23(9), 939-954, 2004.

[5]   Dias, M.B., TraderBots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments. PhD dissertation, Carnegie Mellon University, 2004.

[6]   Kuhn, H. W., The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, vol. 2, 83-97, 1955.

[7]   Munkres, J., Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied* Mathemetics, vol. 5(1), 32-38, 1957.

[8]   Şahin, E., Swarm Robotics: From Sources of Inspiration to Domains of Application. Lecture Notes in Computer Science, Vol. 3342/2005, 10-20, 2005.

[9]   Werger, B., B. and Mataric, M. J., Broadcast of Local Eligibility: Behaviour-based control for strongly cooperative robot teams. *Proceedings of the Fourth International Conference on Autonomous Agents*, 21-22, 2000.

[10]   Botelho, S. C. and Alami, R., M+: a scheme for multi-robot cooperation   through negotiated task allocation and achievement. Proceedings of the 1999 IEEE International Conference on Robotics and Automation, 1234-1239, 1999.

[11]   Gerkey, B. P. and Mataric, M. J., Sold! Auction methods for multi-robot coordination, *IEEE Transactions on Robotics and Automation*, vol. 18(5), 758–768, 2002.

[12]   Zlot, R. , An Auction-Based Approach to Complex Task Allocation for Multirobot Teams. *PhD disertation*, Robotics Institute, Carnegie Mellon University, 2006.

[13]   Kalra, N., Ferguson, D. and Stentz, A. , Hoplites: A Market-Based Framework for Planned Tight Coordination in Multirobot Teams. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 1170-1177, 2005.

[14]   Vig, L. and Adams, J. A., Multi-robot coalition formation. *IEEE Transactions on Robotics*, Vol. 22 Issue: 4, 637-649, 2006.

[15]   Tang, F. and Parker, L. E., A Complete Methodology for Generating Multi-Robot Task Solutions using ASyMTRE-D and Market-Based Task Allocation. *Proceedings of International Conference on Robotics and Automation*, 3351-3358, 2007.

[16]   Hasgul, S., Saricicek, I., Ozkan, M. and Parlaktuna, O., Project-oriented task scheduling for mobile robot team. *Journal of Intelligent Manufacturing*, 20, 151-158, 2009.

[17]   Kaleci, B., Parlaktuna, O., Ozkan, M. and Kirlik, G., Market-based task allocation by using assignment problem. *IEEE International Conference* on *Systems Man and Cybernetics (SMC)*, 135-141, 2010.

[18]   Foundation for Intelligent Physical Agents(FIPA) FIPA   Agent   Management   Specification. *http://www.fipa.org/specs/fipa00023/index.html, 2004*.

[19]   Sycara, K., Paolucci, M., Van Velsen, M. and Giampapa, J.,A. , The RETSINA MAS Infrastructure. *Autonomous Agents and Multi-Agent Systems*, Vol. 7, No. 1/2, 29-48, 2003.

[20]   Dikenelli, O., Erdur , R. C. and Gumus, O. , SEAGENT: a platform for developing semantic web based multi agent systems. *International Conference on Autonomous Agents and Multi Agent Systems*, AAMAS 2005, 1271-1272, 2005.

[21]   Cooper, S. and Taleb-Bendiab, A. ,CONCENSUS: multi-party negotiation support for conflict resolution in concurrent engineering design. *Journal of Intelligent Manufacturing*, **9**, 155-159, 1998.

[22]   Li, G.C.L and Lee, R.S.T., A real-time scene segmentation system using Solely Excitatory Oscillator Networks (SEON). *Journal of Intelligent Manufacturing*, **16**, 669-678, 2005.

[23]   Mosteo, A.R., Multi-Robot Task Allocation for Service Robotics: from Unlimited to Limited Communication Range. *PhD dissertation*, University of Zaragoza, 2010.

[24]   Lagoudakis, M., Keskinocak, P., Kleywegt, A. and Koenig, S. Auctions with Performance Guarantees for Multi-Robot Task Allocation. *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 1957-1962,2004.

[25]   De Vries, S., Vohra, R.V., Combinatoral Auctions :A Survey. *INFORMS Journal of Computing*, **15(3)**, 284-309, 2003.

[26]   Jouandeau, N. and Zhi Yan, Improved trade-based multi-robot coordination. *6th Joint IEEE International Information Technology and Artificial Intelligence Conference (ITAIC)*, Vol. 1, 500 – 503, 2011.

Potential Functions for Voting Dynamics

Neelam Gohar[1] and Paul W. Goldberg[1]
[1]Department of Computer Science, University of Liverpool, Liverpool, United Kingdom

Abstract—*Recent work in the AI literature has studied the properties of voting schemes for performing preference aggregation. An important technical issue that arises is manipulation of voting schemes: a voter may be able to improve the outcome (with respect to his own preferences) by reporting his preferences incorrectly. Meir et al. [11] have studied this phenomenon as a dynamic process in which voters may repeatedly alter their reported preferences until either no further manipulations are available, or else the system goes into a cycle. Here we show how potential functions are useful for showing convergence in variants of this setting, and for obtaining upper bounds on the convergence rate. We focus on plurality voting with weighted voters, and obtain bounds on the lengths of sequences of manipulations, that depend on which types of manipulation are allowed.*

Keywords: Voting protocols, Game Theory (non-cooperative), Modeling the dynamics of Multi-Agent systems

1. Introduction

One of the newer areas explored in artificial intelligence is multi-agent systems, which analyzes interactions between multiple agents, each of which with its own personal objectives. One of the actively growing subareas explored in multi-agent systems is computational social choice theory that provides theoretical foundation for preference aggregation and collective decision-making in multi-agent domains. Computational social choice is concerned with the application of techniques developed in computer science, such as complexity analysis or algorithm design, to the study of social choice mechanisms, such as voting . It seeks to import concepts from social choice theory into AI and computing. For example, social welfare orderings developed to analyze the quality of resource allocations in human society are equally well applicable to problems in multi-agent systems or network design.

People often have to reach a joint decision even though they have conflicting preferences over the alternatives. The joint decision can be reached by an informal negotiating process or by a carefully specified protocol. Over the course of the past decade or so, computer scientists have also become deeply involved in this study. Social choice theory investigates many kinds of multiperson decision-making problems. Multiperson decision-making problems are important, frequently encountered processes and many real world problems involve multiple decision makers. Voting is one of the most popular way of reaching common decisions. The study of elections is a showcase area where interests come from computer science specialists as theory, systems, and AI and such other fields as economics, business, operations research, and political science.

Voting is a well-studied method of preference aggregation, in terms of its theoretical properties, as well as its computational aspects [12], [15]; various practical, implemented applications that use voting exist [16], [1], [14]. In multiagent systems, it is of crucial importance to be able to aggregate the preferences of multiple agents in order to achieve their joint goal even though their individual beliefs may vary over the alternatives (candidates). Voting is a general method of reconciling these differences. A *social choice function* is a function that takes lists of people's ranked preferences and outputs a single alternative (the "winner" of the election). A good social choice function represents the will of the people. However, in voting one of the major issues is manipulation. Manipulation in voting is considered to be any scenario in which a voter reveals false preferences in order to improve (with respect to that voter) the outcome of the election. A manipulative vote in considered successful if it changes the election outcome. This has various negative consequences; not only do voters spend valuable computational resources determining which lie to employ, but worse, the outcome is less likely to be one that reflects the social good. Gibbard-Satterthwaite, Gardenfors, and other such theorems open the door to strategic voting, making voting a richer phenomenon. In order to achieve some standard of non-manipulability in voting schemes, in most previous work the complexity of the manipulation is considered where one could try to avoid manipulation by using protocols where determining a beneficial manipulation is hard. For a survey, see [2].

Here we establish bounds on the length of sequences of manipulations that voters can perform. We obtain bounds in terms of the different types of moves that lead to successful manipulation. In this paper we do not concern ourselves with the impact of manipulation on social welfare; we treat manipulation as an "occupational hazard" and ask the question: in a system where manipulation may occur, when can we guarantee that the voters will end up satisfied with their (possibly manipulative) votes, in the context of the votes offered by the others? Put another way, we suggest that in various real-world situations, it may be better to reach a poor decision than no decision at all. We can regard the voting system as a game in which each voter has, as pure strategies, the set of all votes he may make. (In

plurality voting, a vote is just the choice of a single preferred candidate.) Each voter has a type, consisting of a ranking of the candidates that represents his real preferences. We ask whether pure Nash equilibria exist for any set of voter types, and more importantly whether such an equilibrium can be reached via a sequence of myopic changes of vote, by the players. This can be regarded as a very simplistic model of a negotiation process amongst the voters, and we would like to ensure that it does not end in deadlock.

We focus on plurality voting, the best-known voting system, and one that is known to be susceptible to manipulation, both in practice and theory [6], [7]. We assume that voters have full knowledge of the reported preferences of other voters. Voters manipulate based on their true preferences. Voters change their vote (make a manipulation) after observing the current state and outcome. If a voter cannot affect the outcome at some state, he simply keeps his current reported preference. The process ends when no voter has objections and the outcome is set by the last state.

a) **Manipulation dynamics:** The model is similar to Meir et al. [11]: at each step, a single voter may change his reported vote so as to (according to his own preferences) improve the winner of the election. In game theory, this is sometimes called the elementary stepwise system (ESS) [3]: at each step a single player makes a myopic best or better response. Given any fixed voting rule (such as plurality, considered here) there is a well-defined notion of a valid move: the voter should change the outcome of the election to one that he prefers. Observe that a voter may switch his support to another candidate in response to the moves of other voters, so that a sequence of moves occurs. This sequence may stop at a steady state where no voter has an available manipulation, which in game-theoretic terms is a pure Nash equilibrium (in which a voter's pure strategies consist of the set of possible votes that he may announce, and submit to the voting system).

A related dynamical model was considered by Airiau and Endriss [8], in which at each step, a voter is selected at random and may propose a single alternative to the one currently winning the election; a pairwise vote takes place between the current winner and the new alternative. As is [11] cycles may arise; the ability of the chosen voter to select a single alternative for a pairwise election indeed makes it possible to exhibit cycles which cannot be escaped.

b) **Potential functions:** Potential functions were originally used in the context of game theory by [5]; the paper of [4] showed an equivalence between potential games and congestion games. Potential functions are valuable for proving the existence of pure Nash equilibria and the convergence of best response dynamics.

Given a process involving a finite set $\mathcal{S}$ of states, suppose we have certain allowed *transitions* between states (here, a state is a profile of reported preferences and a transition is a manipulation by a single voter, changing his reported preference). A potential function

$$\Phi : \mathcal{S} \to \mathbb{R}$$

should have the property that any allowable transition from state S to new state S' should always increase the value of Φ. (One could alternatively require the value of Φ to always decrease.) If it's possible for Φ to only take a finite number of distinct values, this will show that the process of making transitions must terminate.

Here states can be defined as the profiles of *reported preferences* of voters, where a reported preference (also known as a declared preference, or an announced preference) is the vote that a voter submits to the social choice function in use. A transition is a manipulation move (change of reported preference) by a single voter. Suppose transitions always cause Φ to increase. Then the process must terminate, and a simple bound on the number of steps is the number of alternative values Φ can take.

c) **Our results:** We focus on plurality voting with weighted voters, in which each voter reports a single preferred candidate. A voter's weight is fixed throughout. The *score* of a candidate is the total weight of voters who support that candidate, hence the winning candidate is the one with highest score, and we assume a standard lexicographic tie-breaking rule in which a candidates have a given total order on them that determines the winner if two or more of them have maximal score. [11] also considers this tie-breaking rule, and compares it with a randomized one.

We investigate rate of convergence, i.e. the number of steps of manipulation that may be needed to reach a pure Nash equilibrium. We focus on types of manipulations where there are no cycles in the state/transition graph, where convergence is guaranteed, and we analyze bounds on the number of steps required. The rate of convergence will be expressed as a function of the number of voters n, the number of candidates m, the ratio $w_{\max}$ of maximum to minimum weights, and the number of distinct weights K. Guaranteed convergence may also depend on types of manipulation available; a classification is given in Section 2.2.

Section 2.4 identifies combinations of types of moves that are able to lead to cycles of manipulation moves. Section 3 considers combinations of move types where convergence is guaranteed, and exhibits various potential functions to obtain upper bounds on the number of manipulation steps possible. Alternative types of moves seem to require alternative potential functions, and we give upper bounds as expressions in terms of the parameters n, m and K.

2. The Model

2.1 Preliminaries, notation

In an election, n voters express their preferences over a set of m alternatives. In general, each voter reveals linear preferences: a ranking of the alternatives. Let us denote the set of alternatives $\mathcal{A}$, where $|\mathcal{A}| = m$, and a set of n voters $\mathcal{V} = \{1, 2, \ldots, n\}$. Let $\mathcal{L} = \mathcal{L}(\mathcal{A})$ be the set of linear preferences over $\mathcal{A}$; each voter $i \in \mathcal{V}$ has true preferences $\succ_i \in \mathcal{L}$. These true preferences are constant. Voters also have "reported preferences" (not fixed) associated with a state of the system. For each voter $i \in \mathcal{V}$ state S has reported preferences for i denoted as $\succ_i^S$. Another notation $a \succ^S b$ represents that candidate a is preferred over b, with respect to reported preferences in state S.

We denote the preference profile, consisting of the voters' reported preferences, by $\succ_{\mathcal{V}}^S = \langle \succ_1^S, \succ_2^S, \ldots, \succ_n^S \rangle$. A voting rule is a function $f : \mathcal{L}^{\mathcal{V}} \to \mathcal{A}$, that maps preference profiles to winning alternatives.

In most of our results we use a weighted voting system. A weighted voting system is one in which the preferences of some voters carry more weight than the preferences of other voters, and the candidate with the highest score as defined in Section 2.2 becomes the winner. Our results have dependence on the voters' weights. We have results for 3 different weight settings.

1) General weight setting:
   A weight function is a mapping $w : \mathcal{V} \to \mathbb{R}^+$. For this type of setting we have bounds in terms of m and n.
2) Bounded real weight setting:
   All n voters have real-valued weights in the range $[1, w_{\max}]$. For this setting we seek bounds in terms of $w_{\max}$ as well as m and n.
3) Bounded integer weight setting:
   Voters have integer weights in the range $\{1, \ldots, w_{\max}\}$. In this setting, the weight function is a mapping $w : \mathcal{V} \to \mathbb{N}$. We seek bounds in terms of $w_{\max}$, m and n.

An additional parameter K, the number of *distinct* weights, can also be added to all 3 settings of weights.

2.2 Types of moves

There are various different types of moves that a voter can perform to make a manipulation. The following classification of moves applies to the plurality rule, where each voter reports a single candidate.

1) Loser to new winner: A move from candidate C to C', where neither was winner beforehand, and C' is winner after move.
2) Loser to existing winner: A move from candidate C to the existing winner C' to improve the score of C'.
3) Winner to loser: A move from a winning candidate C to C' to make C'' a new winner where C'' is different from C and C'.

4) Winner to winner: A move from a winning candidate C to a new winning candidate C' because the manipulator prefers C' over C.
   a) Winner to larger winner: A move from a winning candidate C to another candidate C' such that C' is winner after the move with total score more than the previous score of C.
   b) Winner to smaller winner: A move from a winning candidate C to another winning candidate C' in which the new score of C' is less than the previous score of C.
   c) Winner to winner of same size: A move from a winning candidate C to another winning candidate C' in which the new score of C' is equal to the previous score of C.

[11] considers sequences of moves of type 1, 3 and 4. Moves of type 2 do not change the winning candidate. So, type 2 moves arguably need not be considered in a game-theoretic setting, although upper bounds on convergence rate would ideally allow type 2 moves to take place. Type 3 is arguably unnatural since (for plurality), a type 1 move $C \longrightarrow C''$ would have the same effect, and be more natural.

Observation 1: In a sequence of states where tie-breaking is not needed, moves of type 1, 2 and 4a all increase the score of the winner. Hence, the score of the winning candidate may be used as a potential function to show termination for these types of manipulation move. In a sequence of states where ties may occur, we still have termination for moves of types 1, 2 and 4a, since there is a bound of m on the length of any subsequence in which the score of the winner does not increase.

Most of our upper bounds are for sequences of moves of types 1, 2 and 4a. We shall see however that in some situations one can design "smarter" potential functions that are more useful for showing a faster convergence rate.

We assume that some tie-breaking rule applies if 2 candidates receive the same level of support.

The following example illustrates how voters may change their votes in response to each other.

Example 1: The Chairman's paradox: Suppose there are voters $\mathcal{V} = \{1, 2, 3\}$, alternatives $\mathcal{A} = \{A, B, C\}$. Suppose that voter 1 has preferences $A \succ_1 B \succ_1 C$, voter 2 has preferences $B \succ_2 C \succ_2 A$ and voter 3 has preferences $C \succ_3 A \succ_3 B$ (a Condorcet cycle). Suppose further that in the event that the voters vote for distinct candidates, then the choice of voter 1 (the "chairman") is the winner. This rule of breaking ties in favor of voter 1 can be implemented with voter weights: let voter 1 have weight $\frac{3}{2}$ while voters 2 and 3 have weight 1.

If initially the voters support their favorite candidates, then voter 2 has an incentive to deviate, and he migrates to voting for C. Afterwards, no further migrations are possible. The

chairman's least favorite candidate is chosen.[1]

Suppose instead that initially voter 1 votes for B, and voters 2 and 3 vote for C. Then voter 2 can migrate to B (type 4a move), after that, voter 1 migrates to A (type 4b move), at which point the voters are supporting their preferred candidates. So, voter 2 returns to C (suggesting that voter 1's myopic move to A was a tactical blunder).

2.3 Implementation of lexicographic tie-breaking via additional voters

We show that given any instance of a set of weighted voters where ties may occur, and a lexicographic tie-breaking rule, there exists a (somewhat larger) instance in which no ties occur, but the behavior of the original instance is simulated. This means that while Example 2 uses a lexicographic tie-breaking rule, that is mainly for convenience and there is no need to assume that tie-breaking needs to take place.

Proposition 1: Let $\mathcal{V}$ be a set of weighted voters and let $\mathcal{A} = \{A_1, \ldots, A_n\}$ be a set of alternatives. Assume a lexicographic tie-breaking rule under which A_i is preferred to A_j for $i < j$. Let S be a sequence of strategic moves by the voters $\mathcal{V}$.

Then there exists a set of weighted voters $\mathcal{V}^+ = \mathcal{V} \cup \mathcal{V}'$ for which the same sequence S of strategic moves may occur without tie-breaking; furthermore the additional voters $\mathcal{V}'$ have no incentive to move.

Proof: Let ϵ be the smallest positive winning margin that can occur for a winning candidate, amongst all possible votes of $\mathcal{V}$. We construct $\mathcal{V}'$ as follows.

$\mathcal{V}'$ contains voters v_i, $1 \le i \le n - 1$ with weights $\epsilon.2^{-i}$. v_i has true preferences

$$\succ_i = A_i \succ A_{i+1} \succ A_{i+2} \succ \ldots \succ A_n \succ A_1 \succ \ldots \succ A_{i-1}.$$

v_i initially votes for his favorite candidate A_i. We claim that none of these voters v_i (that comprise $\mathcal{V}'$) ever have an incentive to switch their support. This can be seen by an inductive argument: v_1 can resolve ties that may occur amongst voters in $\mathcal{V}$ but has insufficient weight to have further effect on the outcome. The remaining voters in $\mathcal{V}'$ have less weight than v_1, so v_1 has the incentive to resolve those ties by staying with his preferred candidate A_1. Suppose alternatively that A_2 is the winner amongst voters $\mathcal{V}$ with the tie-breaking rule. Then v_2's vote for A_2 will implement that tie-break, meanwhile v_1's vote for A_1 is insufficient to let A_1 win, since the tie was between A_2 and A_j for $j > 2$; A_1's support (amongst $\mathcal{V}$) falls short of A_2's by at least ϵ. v_1 has no incentive to move, since the best he can do is allow A_2 to win.

A similar argument applies to voters v_i for $i > 2$; given that $v_1 \ldots v_{i-1}$ support $A_1 \ldots A_{i-1}$ respectively, v_i

[1]The paradox [13] is the stronger result that under the *impartial culture* assumption, where preference lists are chosen at random, the chairman actually does worse *on average* than the other voters!

can resolve ties between A_i and A_j for $j > i$, so continues to support A_i. Meanwhile, $v_1, \ldots, v_{i-1}$ can do no better than allow A_i to win. ∎

2.4 Combinations of move types that can lead to cycles

Proposition 4 of [11] gives a simple cycle of manipulation moves involving just 2 manipulating voters, using moves of type 1 and 3. The following new example shows that cycles are also possible using only moves of types 1 and 4. Let $\{A, B, C, D, E\}$ be the candidates. Assume ties are broken lexicographically, in favor of A, then B, then C, then D. Proposition 1 shows how to modify the example so that no tie-breaking need occur. This example contrasts with Theorem 3 of [11] that shows convergence in the case of deterministic tie-breaking and *unweighted* voters.

In addition to the voters $\{1, 2, 3\}$ that make manipulation moves, we assume extra "passive" voters that never move, as follows: A is supported by a voter with weight 1.6, B is supported by a voter with weight 1.9, C is supported by a voter with weight 2, D is supported by a voter with weight 1.9, E is supported by a voter with weight 1.8.

Example 2:

Voter	true preferences	weight
1	$BECAD$	0.1
2	$DEBAC$	0.2
3	$AECDB$	0.4

The tables below show the sum of weights of voters supporting each candidate, for a cyclic sequence of states $S_1, S_2, \ldots, S_9 = S_1$. The manipulation move is given above each table: the notation $i : X \longrightarrow Y$ means that voter i switched his support from X to Y to obtain the state indicated in the table.

State S_1

A	B	C	D	E	**Winner**
2	2	2	1.9	2	A

State S_2, $1 : B \longrightarrow E$ (Type 1 move)

A	B	C	D	E	**Winner**
2	1.9	2	1.9	2.1	E

State S_3, $2 : E \longrightarrow D$ (Type 4c move)

A	B	C	D	E	**Winner**
2	1.9	2	2.1	1.9	D

State S_4, $1 : E \longrightarrow C$ (Type 1 move)

A	B	C	D	E	**Winner**
2	1.9	2.1	2.1	1.8	C

State S_5, $2 : D \longrightarrow B$ (Type 1 move)

A	B	C	D	E	**Winner**
2	2.1	2.1	1.9	1.8	B

State $S_6, 3 : A \longrightarrow E$ (Type 1 move)

A	B	C	D	E	**Winner**
1.6	2.1	2.1	1.9	2.2	E

State $S_7, 1 : C \longrightarrow B$ (Type 1 move)

A	B	C	D	E	**Winner**
1.6	2.2	2	1.9	2.2	B

State $S_8, 2 : B \longrightarrow E$ (Type 4a move)

A	B	C	D	E	**Winner**
1.6	2	2	1.9	2.4	E

State $S_9, 3 : E \longrightarrow A$ (Type 4b move)

A	B	C	D	E	**Winner**
2	2	2	1.9	2	A

In common with the examples of [11], it is possible to escape this cycle and reach a pure Nash equilibrium, by selecting alternative valid manipulation moves. It is an open question as whether there exist alternative moves where cycles can be escaped. These examples are important in ruling out the possibility of applying our approach in Section 3 below, to less restricted manipulation moves.

3. Bounds on the number of manipulations, under constraints on the move types

Our first result is a simple extension of Observation 1.

Theorem 1: In the general weight setting, the process of making type 1, type 2 and type 4a moves terminates within $m. \min(2^n, n^K)$ steps.
The bound is thus better for small K.   *Proof:* We use the potential function $\Phi(S) = N_{\text{win}}(S)$, where $N_{\text{win}}(S)$ is the score of the winning candidate at state S. Consider a transition from state S to new state S'. If no tie-breaking was needed, for these moves we have $\Phi(S') > \Phi(S)$, so the potential increases. If tie-breaking was applied (possible for a type 1 move) we may have $\Phi(S') = \Phi(S)$; in this case the winner in S' has (by the tie-breaking rule) priority over the winner in S, so at most $m - 1$ such moves can occur consecutively. There are at most 2^n distinct possible values for $N_{\text{win}}(S)$ since the level of support of any candidate C is determined by, for each voter i, the binary choice of whether i supports C. This gives the upper bound of $m.2^n$ on the length of a sequence of moves.

If K is the number of distinct weights in the system, the level of support for a candidate C is determined by K numbers in $\{1, \ldots, n\}$. For each weight, if we are given the number of voters having that weight who support C, then we have the score of C. Hence there are $\leq n^K$ values for

this quantity, and we get the upper bound of $m.n^K$ on the length of a sequence of moves.   ∎

If instead of a limit K on the number of distinct weights, we have a limit $w_{\max}$ on the ratio between largest and smallest weights, then we see next that can also obtain polynomial bounds. The result is straightforward if weights are integers (Theorem 2) and a more interesting proof also holds for real-valued weights (Theorem 3).

Theorem 2: In the bounded integer weight setting, the process of making type 1, type 2 and type 4a moves terminates within $\frac{1}{2}(n \cdot w_{\max})^2$ steps.

Proof: We are looking for an upper bound on the number of moves that applies to the case where weights of voters are integers and belong to the set $\{1, \ldots, w_{\max}\}$. Let $N_j(S)$ be the total weight of voters supporting candidate j at state S; w_i represents the weight of voter i. We define the potential function of state S as,

$$\Phi(S) = \sum_{j=1}^{m} (N_j(S))^2 \qquad (1)$$

Suppose a transition from state S to S' is caused by a voter i with weight w_i moving his vote from candidate x to candidate y. The increase in potential for Φ between two successive states S and S' is:

$$\Phi(S') - \Phi(S) = 2 \cdot w_i \cdot (N_y(S') - N_x(S)) \qquad (2)$$

where $N_x(S)$ is the sum of weights of voters that voted for candidate x at state S and $N_y(S')$ is the sum of weights of voters that voted for candidate y at state S'. Since weights are integers, with each move the potential function Φ increases by at least 2. Also, $(n \cdot w_{\max})^2$ is the maximum potential attainable, occurring when n voters all of weight $w_{\max}$ vote for the same candidate. Hence, under the discrete integer weight setting at most $\frac{1}{2}(n \cdot w_{\max})^2$ number of type 1, type 2 and type 4a moves are required to terminate the process.   ∎

Next we show that for the same types of move, we obtain a polynomial bound on the number of steps if instead of limiting K, we limit the ratio $w_{\max}$ between the largest and smallest weight.

Theorem 3: In the bounded real weight setting, the process of making type 1, type 2 and type 4a moves terminates within $mn^3(w_{\max})^2$ steps.

Proof: We apply two non-negative potential functions Φ and Ψ whose values lie in $[0, n^2(w_{\max})^2]$ and $[0, mn]$ respectively. Φ is real-valued; Ψ is integer valued. We will see that any transition increases Φ or reduces Ψ (or both), and for transitions where Φ increases by less than 1, Ψ is reduced. This gives the claimed bound on the number of moves.

We use the potential function Φ from Equation (1) and we know the potential difference between two consecutive states S and S' is the same as (2), i.e. $\Phi(S') - \Phi(S) =$

$2 \cdot w_i \cdot (N_y(S') - N_x(S))$. So, Φ increases with each migration. We find the length of the sequence of moves when $\Phi(S') - \Phi(S)$ is greater than or equal to the weight of the migrating voter and the maximum possible number of steps are $n^2(w_{\max})^2$ as the maximum possible potential being attained when n voters all of weight $w_{\max}$ vote for the same candidate. To find an upper bound on the number of moves in which $\Phi(S') - \Phi(S)$ is less than the weight of the migrating voter we introduce another potential function Ψ.

$$\Psi(S) = \sum_{i \in \mathcal{V}} |\{x \in \mathcal{A} : N_{\mathrm{win}}(S) - N_x(S) < w_i\}| \quad (3)$$

where w_i is the weight of voter $i \in \mathcal{V}$. $N_{\mathrm{win}}(S)$ and $N_x(S)$ are the total support of the winning candidate and any other candidate x at state S, respectively.

Note that for any state S, $0 \le \Psi(S) \le nm$; the highest possible value nm arises in a state where the gap between the winning and all other candidates is less than the weights of voters.

Lemma 1 below shows that in a transition where Φ decreases by less than 1, Ψ decreases by at least 1. That imposes a limit on the number of nm on the number of consecutive transitions where Φ reduces by less than 1. The result follows since there can be at most $n^2(w_{\max})^2$ transitions where Φ increases by at least 1.

Note that the value of Ψ can actually increase, in steps when Φ increases by more than 1, so this approach does not immediately yield a stronger bound than the one claimed. ∎

Lemma 1: If a move by a voter with weight in $[1, w_{\max}]$ reduces Φ by an amount less than the weight of the migrating voter, then it reduces Ψ by at least 1.

Proof: Suppose voter i with weight w_i moves from x to y; let S and S' be the original and the new state, respectively.

The potential difference between two successive states S and S' is:

$$\Phi(S') - \Phi(S) = 2 \cdot w_i \cdot (N_y(S') - N_x(S))$$

equivalently,

$$\Phi(S') - \Phi(S) = 2 \cdot w_i \cdot (w_i - (N_x(S) - N_y(S))$$

In order for $\Phi(S') - \Phi(S) < w_i$, the value of $w_i - (N_x(S) - N_y(S))$ must be less than $\frac{1}{2}$, i.e.,

$$w_i - \frac{1}{2} < (N_x(S) - N_y(S)).$$

For both types of move to take place, we already know that $(N_x(S) - N_y(S)) < w_i$. For $\Phi(S') - \Phi(S) < w_i$, gap between $N_x(S)$ and $N_y(S)$ should be greater than $w_i - \frac{1}{2}$ and less than w_i.

At state S, when voter i migrates from a winning candidate or a loser to a new candidate y, we know that $N_x(S) - N_y(S) < w_i$. After migration, for both types of

moves $N_{\mathrm{win}}(S') - N_x(S') > w_i$. This shows the potential Ψ decreases with migration of voter i. Because at state S, $N_x(S) - N_y(S) < w_i$ was true. After migration of voter i, at state S' there is at least one voter i for which $N_{\mathrm{win}}(S') - N_x(S') < w_i$ is false. So, clearly potential Ψ reduces by at least 1. ∎

We extend Theorem 3 to a setting where a small number of voters may have weights larger than $w_{\max}$; we show that if the number k of such voters is logarithmic in n we still have a polynomial bound. More precisely we have

Theorem 4: In the bounded weight setting, the process of making type 1, type 2 and type 4a moves terminates within $2^k m^2 n^3 (w_{\max})^2$ steps when k voters have weights $> w_{\max}$.

Proof: Divide the n voters into two categories: k "heavy voters" having weights $> w_{\max}$ and the remaining $n - k$ "ordinary voters". Let $\{1, \ldots, k\}$ be the heavy voters, with weights $w_1, \ldots, w_k$, assumed to be arranged in descending order so that for $i < k$ we have $w_i > w_{i+1}$.

Notice that for any pair x, y of candidates, if voter 1 (the heaviest voter) migrates from candidate x to y using a move of type 1,2 or 4a, then if S is the new state, we have $N_x(S) < N_{\mathrm{win}}(S) - w_1$, i.e. the support of x is less than the support of the winner by more than w_1. Hence, thereafter x cannot possibly win, since the support of y is less than the support of the winning candidate by a quantity greater than w_1 (the largest weight of any voter). (As an aside, x could possibly become the winner if a move of type 4b was allowed.)

Hence, voter 1 may only migrate at most $m - 1$ times, and furthermore, may not return to a candidate that he previously supported.

Consider voter 2. If voter 2 migrates from x to y, then in order for him to return to x, it is necessary for voter 1 to migrate to x beforehand. By an argument similar to the above, no voter with weight $\le w_2$ is able to make x the winner. Since we have seen that voter 1 may only move to x once, it follows that (for any x) voter 2 may only move to x (at most) twice.

Applying the above idea repeatedly, for any candidate x, for $i \le k$ voter i may only migrate to x at most 2^i times, which results in a bound of $2^i . m$ on the number of moves i may make.

Theorem 3 applies in the presence of heavy voters that do not migrate, so that in between any pair of moves by heavy voters, the ordinary voters may only make at most $mn^3(w_{\max})^2$ moves in total. This results in the overall bound of $2^k m^2 n^3 (w_{\max})^2$ on the number of moves of all voters. ∎

We know from Example 2 that transitions of types 1, 4a and 4b may cycle (the type 4c move can be made into one of type 4a by adding the low-weight voters as discussed). We show next that combinations of types 4a and 4b do not have cycles, and neither do combinations of types 1 and 4b.

The following lemma is used in the proof of Theorem 5:

Lemma 2: In the unrestricted real weight setting, the process of making type 4a and 4b moves terminates within $n \cdot (m-1)$ steps.

Proof: Notice that in a type 4 moves (either 4a or 4b) when a voter moves from candidate x to y, that means that he truly prefers y to x.

Technically we are using the potential function:

$$\chi(S) = \sum_{i \in \mathcal{V}} | \{x \in \mathcal{A} : x \succ_i \text{vote}_S(i)\} | \qquad (4)$$

where $\text{vote}_S(i)$ is the candidate supported by i in state S.

We complete the proof with the observation that each migration reduces χ by at least 1 in a single move of type 4, and χ is at most $n \cdot (m-1)$ (that upper bound occurs in a state where all voters vote for their least-preferred candidate). ∎

Theorem 5: A mixture of type 1 and type 4b moves converges within $2^n mn$ steps.

Proof: For a mixture of type 1 and 4b moves, we have the following observation.

Observation 2: At every time step the total weight of voters supporting the second-highest supported candidate either remains the same or increases when moves of voters are of type 1 and 4b.

We use a similar potential Φ as we have used in Theorem 1. But this time instead of score of the winning candidate, our potential function is based on the score of the 2nd-highest candidate as per Observation 2, the support of the 2nd-highest candidate increases or remains the same. Let $N_{2nd}(S)$ be the score of the 2nd-highest candidate at state S. So the potential function is

$$\Phi(S) = N_{2nd}(S).$$

From Observation 2, the score of the 2nd-highest candidate never decreases. So the largest number of times the size of the second-highest candidate can increase is 2^n as there are 2^n possible set of voters. The support of the 2nd-highest candidate can remain the same when a move is of type 4b. From Lemma 2 we know at most mn consecutive moves of type 4b are required to terminate the process of manipulation. So the support of the second-highest candidate remains the same at most mn times and hence the possible number of type 1 and 4b moves are $\leq 2^n mn$. ∎

3.1 Bounds in terms of K

We consider in more detail the results obtainable in the case where the number of *distinct* weights K is small. We begin with a simple result for the case of unweighted voters (i.e. where $K = 1$). For this situation, [11] (Theorem 3) gives an upper bound of $m^2 k^2$ where k is the number of voters that may manipulate, and best-reply moves are assumed. The following result is for moves just of types 1 or 2, but on the positive side, better-reply moves are also allowed.

Theorem 6: With a lexicographic tie breaking rule, if all the voters have weight 1, any sequence of type 1 and type 2 moves has length at most n^2 or mn.

Proof: With type 1 move the votes of the new winner either remains the same or increases and with type 2 move the total votes of the winner will always increase. So with both these types of moves the votes of the winner never decreases. The tie breaking rule has a significant impact on outcome. If there is more than one winner then the candidate who is first in the sequence is the winner. Applying potential function from Equation 1, potential Φ increases with each such migration and is bounded by n^2. ∎

Suppose there are K distinct weights where weights are positive real numbers and let there be n voters where $n \leq K$. For this setting we seek bounds in terms of K as well as m and n. For K different real weights we obtain improved bounds for Theorem 3 and Theorem 5 in terms of K, n and m.

Theorem 5 The bound of Theorem 5 can be improved to n^K using the general observation that the support of any candidate may only take at most n^K distinct possible values, and Theorem 5 uses the potential function consisting of the total weight of voters supporting the second-most supported candidate.

Theorem 3 The bound of Theorem 3 can similarly be improved to $n^K \cdot mK$. The proof uses the two potential functions denoted Φ and Ψ. We noted that Φ may only take n^K distinct values since it represents the support of the winning candidate.

For Ψ, we rewrite Equation 3 as follows. Let $\mathcal{W}$ be the set of distinct weights; $|\mathcal{W}| = K$.

$$\Psi(S) = \sum_{w \in \mathcal{W}} | \{x \in \mathcal{A} : N_{\text{win}}(S) - N_x(S) < w\} | \quad (5)$$

Ψ cannot take a value larger than mK. Then we claim that in a similar way to the proof of Theorem 3, if Φ is not reduced in a manipulation move, then Ψ is reduced by at least 1.

4. Conclusions and further work

These results, in conjunction with the ones of [11] provide quite a complete knowledge of what combinations of types of manipulation move can result in cycles. In the cases where cycles cannot occur, we also obtain polynomial bounds on the lengths of sequences of manipulations. We note that for alternative voting rules, we have some preliminary results that suggest that it is generally easier to find cycles (those examples require just 2 voters that change their reported preferences).

In cases where polynomial bounds have been obtained that depend on parameters K and w_{max}, there remains the possibility that polynomial bounds exist that do not depend on those parameters, but just on m and n. Example 2 and examples in [11] indicate that one might alternatively want to consider relaxing the assumption of worst-case selection

of manipulation move, and show that where cycles exist, it is still possible to reach an equilibrium after a reasonably small number of steps. One reasonable question to investigate is the possible convergence of randomly-selected manipulations. A rich range of results have emerged in other game-theoretic contexts. We have fast convergence to approximate equilibria in congestion games that require exponentially-long paths to reach exact equilibria [9]. In the context of matching markets it is found that simple local search heuristics may be stuck in a cycle for exponentially many steps, even when there are short paths to Nash equilibria [10].

References

[1] Sumit Ghosh and Manisha Mundhe and Karina Hernandez and Sandip Sen, *Voting for movies: The anatomy of a recommender system*. Proceedings of the 3rd Annual Conference on Autonomous Agents (AGENTS),434–435, 1999

[2] Piotr Faliszewski and Edith Hemaspaandra and Lane. A. Hemaspaandra, *Using Complexity to Protect Elections*. Communications of the ACM, 53, 11, 74–82, November, 2010

[3] A. Orda and R. Rom and N. Shimkin, *Competitive routing in multiuser communication networks*. IEEE/ACM Trans. on Networking, 1, 5, 510–521, October, 1993

[4] D. Monderer and L.S. Shapley, *Potential Games*. Games and Economic Behavior, 14, 124–143, 1996

[5] Robert. W. Rosenthal, *A Class of Games Possessing Pure-Strategy Nash Equilibria*. International Journal of Game Theory, 2, 1, 65–67, 1973

[6] Donald. G. Saari, *Susceptibility to manipulation*. Public Choice, 64, 1, 21–41, 1990

[7] Robert Forsythe, Thomas Rietz, Roger Myerson and Robert Weber, *An experimental work of voting rules and polls in three-candidate elections*, International Journal of Game Theory, 25, 3, 335–383, 1996

[8] Stephane Airiau and Ulle Endriss, *Iterated Majority Voting*. Proceedings of ADT, 2009, 38-49, Springer-Verlag LNAI 5783

[9] Steve Chien and Alistair Sinclair, *Convergence to Approximate Nash Equilibria in Congestion Games*. Games and Economic Behavior, 71, 2, 2011, 315-327, Elsevier

[10] Heiner Ackermann and Paul W. Goldberg and Vahab S. Mirrokni and Heiko Rï£¡glin and Berthold Vï£¡cking", *Uncoordinated Two-Sided Matching Markets*. SIAM Journal on Computing, 40, 1, 2011

[11] R. Meir and M. Polukarov and J. S. Rosenschein and N. R. Jennings, *Convergence to Equilibria in Plurality Voting*. Proceedings of 24th Conference on Artificial Intelligence (AAAI-10), 2010

[12] Vincent Conitzer and Tuomas Sandholm and Jï£¡rï£¡me Lang, *When are elections with few candidates hard to manipulate?*. Journal of the ACM, 54, 3, 1–33, June, 2007

[13] R. Farquharson, *Theory of Voting*. Yale University Press, New Haven, 1969

[14] Thomas Haynes and Sandip Sen and Neeraj Arora and Rajani Nadella, *An automated meeting scheduling system that utilizes user preferences*. Proceedings of the 1st Annual Conference on Autonomous Agents (AGENTS), 308–315, 1997

[15] Ariel. D. Procaccia and Jeffrey. S. Rosenschein, *Junta distributions and the average-case complexity of manipulating elections*. Journal of Artificial Intelligence Research, 28, 1, 157–181, January, 2007

[16] Eithan Ephrati and Jeffrey. S. Rosenschein, *A heuristic technique for multiagent planning*. Annals of Mathematics and Artificial Intelligence, 20, 1–4, 13–67, 1997

Applying Knowledge Model To Agent Based Systems

Khaoula ADDAKIRI
Hassan 1 University, FSTS
Department of Mathematics and Computer Science
Khaoula.addakiri@gmail.com

Mohamed BAHAJ
Hassan 1 University, FSTS
Department of Mathematics and Computer Science
mohamedbahaj@gmail.com

Abstract— **Nowadays, multi-agent system is become promising means for the development of distributed systems, however its disadvantage is that it lacks the interconnection with semantic web such as Ontology Web Language (OWL). In this article, we aim to present a semantic knowledge model of an agent suitable for discrete environments as well as implementation and a use of such model using different softwares (JENA, JADE, JESS and Protégé) in order to allows interconnection of Agent and Semantic Web technologies which can be used in an agent based application where such interconnection is needed.**

Keywords— *Multi Agent System ; Web Ontology Language (OWL); Java Expert System Shell (JESS); SPARQL1.1.*

I. INTRODUCTION

Multi-Agent System (MAS) is a powerful paradigm in nowadays and is become promising means for the development of distributed systems [1, 2, 3]. In Multi-Agent System the interoperability allows agents to communicate and cooperate in order to attain their own objectives and sometimes to solve a common problem. As part of the open multi-agent system, heterogeneous agents, is designed by different organizations and in different languages and can dynamically join or leave the system. This implies a dynamic environment changing functionality, which makes the problem of interoperability of heterogeneous agents very complex. Among the proposed solutions, FIPA [4] proposes specifications that ensure interoperability at different levels. In FIPA-ACL communication protocol already offers the use of ontologies for describing the content of messages. Unfortunately, ontologies are most often used simply to write the format of messages exchanged forgetting the semantic part.

The Semantic Web [5, 6] is an extension of the current web. According to the World Wide Web Consortium (W3C),"The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries"[7]. The main goal of the Semantic Web is to enable users to find their applications more efficiently by allowing the machine to understand and respond to human request according to their meaning. For that to happen, Web resources must be described using a set of W3C standards and technologies to enable its processing. Among these standards are Resource Description Framework (RDF), RDF Schema and the Web Ontology Language (OWL) [8], all of which aim to provide a formal description of concepts, terms and relationships within a given knowledge domain.

Java Agent Development Framework (JADE) is a software framework for the development of multi agent applications in compliance with The Foundation for Intelligent Physical Agents (FIPA) specifications [9]. Many approaches to build knowledge model for JADE agent can be found. The most promising approach is using ontology based knowledge representation which is one of the main standards for the Semantic Web proposed by World Wide Web Consortium (W3C), and it is based on Description Logic (DL). Representing knowledge based on ontology provides many advantages over other representations. The traditional approach used conventional rule engine. Java Expert System Shell [10] (JESS) is a familiar rule engine written entirely in Sun's java language, provides a powerful tool for developing systems with intelligent reasoning abilities. The Jess gives the possibility for building Knowledge in the form of declarative rules and facts, and reason about it.

The aim of this article is to build an agent knowledge model that takes advantages of description logics expressivity used in semantic web technologies OWL. A behavioral architecture is implemented to build an intelligent agent in JADE platform with knowledge models based on OWL ontology which is implemented in Protégé. Moreover, Jess language was used to enrich the ontology with logic and functionality. Then, we used The Jena Semantic Web [11] Toolkit which is a Java Application Programming Interface (API) and software toolkit for manipulating RDF, RDFS, OWL and SPARQL which can be used to query data that is structured in hierarchies.

This paper is structured as follows. In section 2, we give an overview of the multi-agent systems and the different agent architecture. Section 3 introduces the benefits of introducing ontology and semantic web into these systems. Section 4 introduces the Ontology web Languages which is based on Description Logic and explains how to build knowledge model in JADE platform. Section 5 illustrates how steps to build successfully the Semantic Web application; finally, we draw conclusions in section 6.

II. RELATED WORK

At the beginning of the decade, the publishing of the agent technology roadmap [12] pointed out the lack of interconnection between multi-agent systems and semantic web technologies. Since then, several applications and frameworks have been developed to bridge this gap:

B.Schiemann et al. [13] proposed Owl-dl as a FIPA ACL content language in order to facilitate the building of simple interaction protocols that are based on information dialogues and to separate of speech act semantics from content semantic. One limitation of this approach is that they only use propositions or referential expressions in the content field of speech acts (Inform and Query-ref).
 M. Laclavik et al. [14] presented a semantic knowledge model of an agent suitable for discrete environments as well as implementation and a use of such model using the Jena semantic web library, the JADE agent system and model it in the Protégé ontology editor.
M.Babik et al. [15] described how semantic web technologies can be applied in multi-agent system and an agent knowledge model was created to allow the   possibility to model the agent environment, agent context and its results. The agent knowledge model was developed and extends the JADE agent system, which is the most popular MAS toolkit.
A.M.Zarafin et al. [16] proposed a semantic description of multi-agent systems, showing the advantages regarding integration with Web semantic technologies.
M.Obitko et al. [17] presented how semantics and ontologies can be employed in industrial systems, considering particularly distributed, agent-based solutions and demonstrated the integration of new manufacturing ontology with an agent-based simulation and control system MAST.
N.B.Aldabagh et al. [18] developed a comparative study between the implementation of an intelligent agent in JADE platform with two different knowledge models. The first one is based on OWL ontology, the second is by integrating the agent with the Jess.

III. MULTI AGENT SYSTEM

 An agent is defined as "person who's acting on behalf of other people" [19]. In the context of computer science, mobile agent is considered as an entity that moves from one machine to another in the network to perform certain tasks on behalf of the user [20]. Mobile agents have the following properties which distinguish them from other programs: Adaptability, Autonomy, Communication, Mobility and Persistence [21].

Agent architectures are the fundamental mechanisms underlying the autonomous components that support effective behavior in real-world, dynamic and open environments. From the start, initial efforts are focused on the development of intelligent agent architectures [22], FIPA develop open specifications, to support interoperability among agents and agent based applications. FIPA give nothing about how to build internal knowledge in an agent, leaving that to the developers. Thus, we can see different approaches to build intelligent agent in different FIPA complaint agent systems [23].

Many agent architectures are developed to support intelligent agent:

- Reactive architectures implement decision-making as a direct mapping of situation to action and are based on a stimulus–response mechanism.

- Belief Desire Intention (BDI): Are the most popular agent architectures and can reason about their actions.

- Behavioral architecture: An agent has many behaviors and it can be executed in sequence or in parallel depending on the task to effect. This architecture is more suitable for used in real applications and our implementations will based on it.

A. JADE

The Java Agent Development Framework (JADE) is a flexible agent platform that provides a middleware layer for the development of distributed multi-agent systems in compliance with FIPA specifications [9]. It provides the following mandatory components for agent's management:

- AMS (Agent Management System), which besides to provide white page services as specified by FIPA and to play the role of authority in the platform.

- DF (Directory Facilitator), which provides yellow pages services to other agents.

- ACC (Agent Communication Channel), which is responsible for sending and receiving messages on an agent platform.

Among the advantages of JADE is the ability to integrate with others java implementation tools, like Jena .This tool can be used to build knowledge model within an agent and reason over it.

B. Building Knowledge Model in JADE

Knowledge of the agents is stored in the Web Ontology Language (OWL).It rests primarily on two existing and linked APIs from JENA and Protégé. The JENA framework is an open-source Semantic Web framework for Java that provides

APIs to build semantic Web applications managing the RDF and OWL languages. It also provides methods to write and read in RDF/XML, along with a SPARQL (Query Language for RDF) query engine. The Protégé-OWL plugin is described as an extension of Protégé that supports OWL. Protégé is a free, open source ontology editor and knowledge-base framework. The Protégé-OWL API provides methods to load and save OWL and RDF ontologies, and to edit these ontologies in a powerful way.

IV.   ONTOLOGY AND SEMANTIC WEB

Ontology is a term which has been used to name the general discipline of metaphysics, in the traditional first philosophy of Aristotle, It is, many times, faced as a complement to the idea of epistemology (science of knowledge) [24]. Ontology based knowledge representation allow the developers to share knowledge between different entities, also to reuse it over well defined Web ontologies. Thus such knowledge model will improve interoperability between different agents in different platforms.

The utilization of ontologies is mainly related to the Semantic Web (SW). SW (Figure1) aims to provide a common framework in order to allow data to be effectively shared and reused. It is envisioned as an extension of the World Wide Web that brings the semantic description of content so that it can be found, processed and integrated by software agents more effectively [25]. The core semantic web technologies are Resource Description Framework (RDF) and Web Ontology Language (OWL).
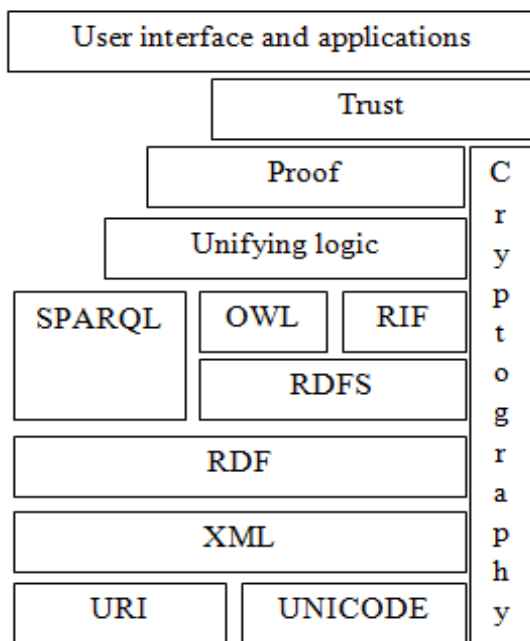

Figure1: The semantic web diagram by W3C

V.   WEB ONTOLOGY LANGUAGE

Web Ontology Language (OWL) it's a language recommended by the W3C for expressing ontologies in the Semantic Web [3]. OWL comes in three increasingly expressive sublanguages. OWL-DL is a one of sublanguages which provides the maximum expressiveness with complete and decidable reasoned. Such languages are based on Description Logic [24].

A. Description Logics

Description Logics (DL) are a family of knowledge representation formalisms used to represent ontology based knowledge. The basic syntactic building blocks are concepts which correspond to classes, roles which represent relationships between two concepts or concept and a data type and individuals which represent classes instances. In DL the knowledge base consists of a:

- TBox(terminological box): which contains definitions and assertions about classes and properties.

- ABox(assertion box): which contains the facts about the instances (individuals) in terms of basic classes, properties and classes.

VI.   EXAMPLES

- First step:

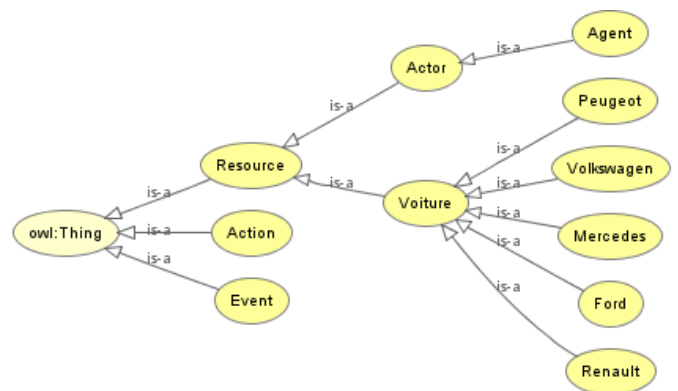We create a simple ontology using protégé (Figure2).

Figure2 : The ontology

- Second step

The next step is to enrich the ontology. This is managed using the JessTab plugin of Protégé which allows writing code in Jess in the protégé environment.

For example if we want to add an individual F7 to the class Ford (Figure3) with three datatype: title, price (prix) and state (etat) .The code writing in Jess is as follow:

```
(make-instance F7 of Ford (title "Ford cayen") (prix 200000)
(etat "excellent"))
<External-Address:SimpleInstance>
Jess> (mapclass Ford)
Ford
Jess> (facts)
f-0 (object (is-a Ford) (is-a-title "Ford cayen")
(OBJECT <External-Address:SimpleInstance>)
(prix 200000) (title "Ford cayen") (etat "excellent"))
For a total of 1 facts.
```
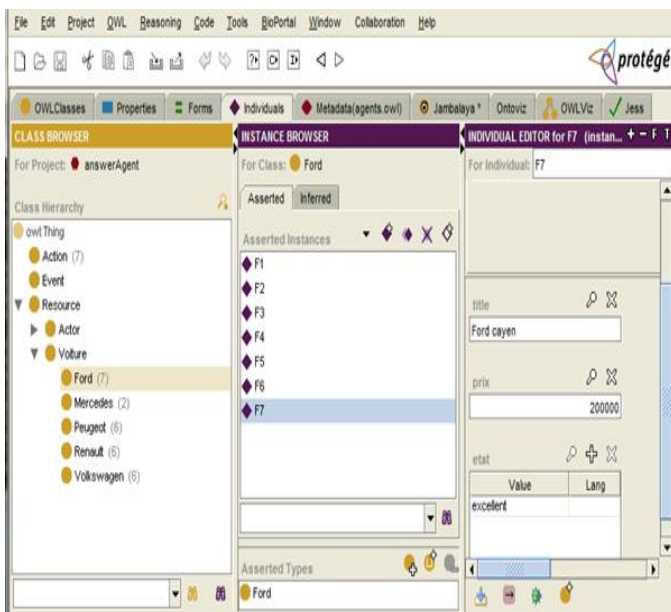

Figure3:The exuction of the code

- Third step:

We develop two agents:
- AskAgent
- Answeragent

The answer and ask agents use the same ontology, but with different individuals. In Figure 4 you can see the model of the AnswerAgent. For example, the model contains five brands: Ford, Renault, Volswagen, Peugeot and Mercedes .The brands ford contain 7 individuals: F1, F2, F3, F4, F5, F6, and F7.
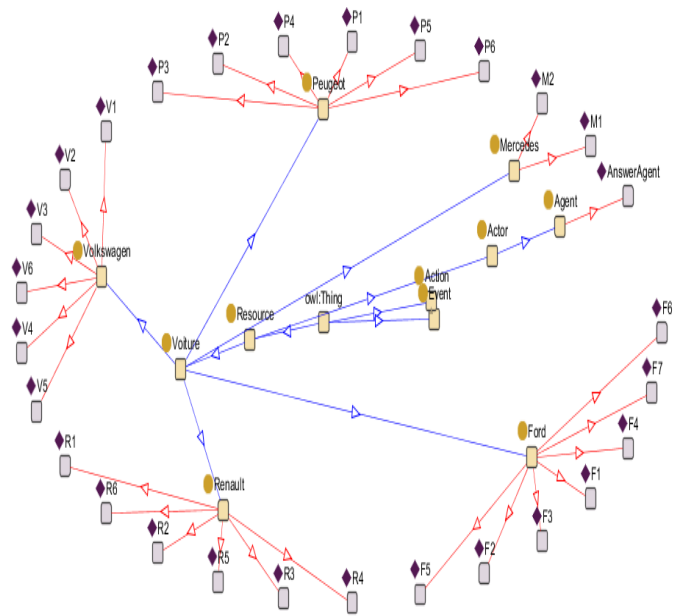

Figure4: The model of AnswerAgent

- Forth step

A user selects a type of resource for which the AskAgent search. The type of the resource is passed to an agent by the XML-RPC method call. One of the AskAgent behaviors is activated and the AskAgent produces the SPARQL query and passes the ACL QUERY message to the AnswerAgent.

```
(QUERY-REF
 :sender ( agent-identifier :name AskAgent@khaoula-
PC:1099/JADE )
 :receiver (set ( agent-identifier :name AnswerAgent@khaoula-
PC:1099/JADE ) )
 :content "PREFIX ont: <http://fsts.ac.ma/agents.owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?x ?y WHERE { ?x  ont:prix ?y.FILTER (?y>94000)}"
 :language  SPARQL  :ontology  http://fsts.ac.ma/agents.owl )
```

The AskAgent asks the AnswerAgent to return to it all cars it has in the memory with the condition the price is superior of 94000. The AnswerAgent receives an ACL QUERY message and performs an SPARQL query on its memory. The result is passed as various ACL INFORM messages consisting of the RDF description of requested resource.

```
(INFORM
 :sender ( agent-identifier :name AnswerAgent@khaoula-
PC:1099/JADE )
 :receiver (set ( agent-identifier :name AskAgent@khaoula-
PC:1099/JADE ) )
 :content "<rdf:RDF
   xmlns:rdf=\"http://www.w3.org/1999/02/22-rdf-syntax-ns#\"
   xmlns:owl=\"http://www.w3.org/2002/07/owl#\"
```

```
 xmlns:j.0=\"http://fsts.ac.ma/agents.owl#\"
 xmlns:xsd=\"http://www.w3.org/2001/XMLSchema#\"
 xmlns:rdfs=\"http://www.w3.org/2000/01/rdf-schema#\">
 <owl:Class rdf:about=\"http://fsts.ac.ma/agents.owl#Ford\">
  <rdfs:subClassOf
rdf:resource=\"http://fsts.ac.ma/agents.owl#Voiture\"/>
 </owl:Class>
 <j.0:Ford rdf:about=\"http://fsts.ac.ma/agents.owl#F5\">
  <j.0:title
rdf:datatype=\"http://www.w3.org/2001/XMLSchema#string\"
  >Ford Mondeo</j.0:title>
  <j.0:etat
rdf:datatype=\"http://www.w3.org/2001/XMLSchema#string\"
  >presque une épave</j.0:etat>
  <j.0:prix
rdf:datatype=\"http://www.w3.org/2001/XMLSchema#int\"
  >120000</j.0:prix>
 </j.0:Ford>
</rdf:RDF>"
```

When the AskAgent receives an ACL INFORM message, it stores its context into its memory model.

- Five step

When a user clicks on the "getXML" button, we get the result in the XML format.

```
<Agent ID="AskAgent">
 <resource>
  <Volkswagen ID="V4">
   <titre>Volswagen Caddy Tramper</titre>
   <etat>révisé régulièrement</etat>
   <prix>160000</prix>
  </Volkswagen>
 </resource>
 <resource>
  <Volkswagen ID="V2">
   <etat>presque une épave</etat>
   <prix>150000</prix>
   <titre>Volswagen Caddy Maxi</title>
  </Volkswagen>
 </resource>
 <resource>
  <Renault ID="R6">
   <title>Renault Scenic</title>
   <prix>95000</prix>
   <etat>presque une épave</etat>
  </Renault>
 </resource>
 <titre>AskAgent</titre>
</Agent>
```

VII.   CONCLUSION

Multi-Agent System (MAS) is a powerful paradigm in nowadays and is become promising means for the development of distributed systems. However its disadvantage is that it lacks the interconnection with semantic web standards such as Ontology Web Language (OWL).

This paper presents how semantic web technologies as Ontology web Language (OWL) can be applied to build an intelligent JADE agent. So, an agent knowledge model was created using protégé and using JESS language to enrich the ontology. Then, using Jena which provides methods to write and read in RDF/XML. Last but not least the use of Sparql1.1 like Filter which restricts solutions to those for which the filter expression evaluates to true.

REFERENCES

[1]   D.Chess,C.Harrison, A.Kershenbaum.Mobile Agentes: Are They a Good Idea?. IBM ResearchReport, 1998 .

[2]   J.Kelemen.The Agent Paradigm. – Foreword. In Computing and Informatics,Vol. 22, 2003, No. 6, p. 513–519.

[3]   D.Lavbic, R.Rupnik .Multi-Agent System for Decision Support in Enterprises.JIOS, Vol. 33, No. 2 (2009).

[4]   FIPA: Foundation for Intelligent Physical Agents website. http://www.fipa.org/.

[5]   M.Obitko,P.Vrba,V.Marik. Applications of Semantics in Agent-Based Manufacturing Systems. Informatica 34(2010) 315–330 315.

[6]   J.J.Samper,F.J.Adell,L.Berg and J. J. Martinez.Improving semantic web service discovery.JOURNAL OF NETWORKS, VOL. 3, NO. 1, JANUARY 2008.

[7]   T. Berners-Lee, J. Hendler, and O. Lassila, ,(2001). The Semantic Web.Scientific American, May 2001, p.28-37.

[8]   D.Q. Zhang, T. Gu, H.K. Pung.Ontology Based Context Modeling and Reasoning using OWL. In Proc. Second IEEE Annual Conference, March 2004, p. 18-22.

[9]   A. Lupasc.A multi agent platform for developments of Accounting Intelligent Applications.annuals of "Dunares de jos", University of Galati, Fascicle1- Economics and Applied informatics, 2008.

[10] JessWeb Site [Online]. Available: http://www.jessrules.com

[11] Semantic Web Framework for Java. http://jena.sourceforge.net/

[12] M. Luck, P. McBurney, and C. Preist, "Agent technology: Enabling next generation computing", AgentLink II, 2003.

[13] B.Schiemann, U.Schreiber, "OWL DL as a FIPA ACL content language", In Nicola Guarino Roberta Ferrario and Laurent Prevot, editors, Proceed-ings of the Workshop on Formal Ontology for Communicating Agents (FOCA),18th European Summer School of Language, Logic and Information, pages 73–80, University of Malaga, July 2006.

[14] M.Laclavik, Z.Balogh, M Babik,L.Hluchy, "AgentOwl: semantic knowledge model and agent architecture", Computing and Informatics, Vol. 25, 2006, 421–439.

[15] L.Hluchy, M. Laclavik, Z.Balogh, Emil Gatial, "Semantic Knowledge Model and Architecture for Agents in Discrete Environments", ECAI 2006: 17th European Conference on Artificial Intelligence.

[16] A.M.Zarafin, A. Zimmermann,O.Boissier, "Integrating Semantic Web technologies and Multi-Agent Systems: a Semantic Description of Multi-Agent Organizations", 15-16 October 2012, Dubrovnik, Croatia.

[17] M.Obitko, P.Vrba and V. Marik," Applications of Semantics in Agent-Based Manufacturing Systems", Informatica 34(2010) 315–330 315

[18] N.B.Aldabagh, B.S. Mustafa, "A Comparative Study between Using OWL Technology and Jess Rule Based For Applying Knowledge to Agent Based System", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 10, No. 7, July 2012.

[19] P.M .Reddy," Mobile Agents Intelligent Assistants on the Internet", July 2002.

[20] A. Kaur , S.Kaur, "Role of Mobile Agents In Mobile Computing", Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007) RIMT-IET

[21] M. L. Griss," Software Agents as Next Generation Software Components", Component-Based Software Engineering: Putting the Pieces Together, Edited by George T. Heineman, Ph.D. & William Councill, M.S., J.D., May 2001,Addison-Wesley.

[22] F.Bellifemine, G.Caire, D.Greenwood," Developing Multi-Agent Systems with JADE".2007, ISBN-13: 978-0-470-05747-6.

[23] N.B.Aldabagh, B.S.Mustafa," A Comparative Study between Using OWL Technology and Jess Rule Based For Applying Knowledge to

Agent Based System". (IJCSIS) International Journal of Computer Science and Information Security, Vol. 10, No. 7, July 2012.

[24] R.R.Azevedo, C.Rodrigues, J. Siqueira , "An Autonomic Ontology-Based Multiagent System for Intrusion Detection in Computing Environments". International Journal for Infonomics (IJI), Volume 3, Issue 1, March 2010.

[25] I. Herman.W3C Semantic Web Homepage. http://www.w3.org/2001/sw.

Intelligent Agent-based Framework for Project Integration Management

Ogechikanma Linda Ihekweaba[1], Chukwugoziem Ihekweaba[1], Prof. H.C Inyiama[2]

[1]Computer Engineering Dept. Michael Okpara University of Agriculture, Umudike, Abia State Nigeria.

[2]Department of Computer Electronics, Nnamadi Azikiwe University, Awka, Nigeria.

Abstract - The dynamism required in the continuous aggregation of the results of all subsidiary planning activities in a project into a consistent document(the project plan), the effective and progressive elaboration of such a document with updates ,the continuous communication requirement during project execution activities coupled with the computational effort required for various ongoing performance evaluation analysis and the constant coordination of changes across the entire elements of a project is definitely overwhelming to the project manager using conventional Project Management Information System. This paper proposes the exploitation of the **dynamism** of *intelligent agents* to support *project integration management*, a core knowledge area of project management. Using the **Belief-Desire-Intention (BDI)** agent architecture this paper defines an *intelligent agent based framework* to support project integration management.

Key Words: *Intelligent Agents, Project Integration Management.*

1.   Introduction

Artificial Intelligence (AI) is currently applied in many fields. One of such fields is project management. Project management is the application of knowledge, skills, tools and techniques to project activities to meet project requirements [PMBOK]. Project management is accomplished through the application and integration of its processes: *initiating, planning, executing, monitoring and controlling*[PMBOK].A sizeable number of research works[1,2,3,4,5,6,7] have approached the subject matter of applying AI/Expert system(ES) to project management. These efforts address various areas including knowledge-base project schedule planning, using ES to analyze and control project schedule, intelligent project estimate validation, intelligent quality prediction model of risk allocation in construction contracts etc.

This paper focuses on the application of intelligent agent concepts in project integration management. However the proposed framework is not meant to be viewed as an effort to replace the project manager with artificial entities, rather is conceived to be an intelligent tool to assist the project manager .It is intended to enable the project manager to focus less on too many details and focus on the pig picture while the complex details of ensuring effective coordination of information across the project scope is left to intelligent agents.

The rest of the paper is organized as follows. Next section gives background information on intelligent agents. Section 3 provides the definition of project integration management and stresses the need to apply AI techniques to it. Section 4 provides background information on BDI architecture. Section 5 presents the design of the proposed agent framework, stating the underling scheme for the construction of its BDI structure. Finally section 6 concludes the paper.

2. Intelligent Agents

The concept of intelligent agents belongs to the field of artificial intelligence (AI), a branch of computer science [14, 15]. The idea of intelligent agents was introduced in the mid-1950s [15]. Agent technology has constituted a remarkable field of research interest for about a couple of decades or more. Agent systems has been used in various applications such as workflow, scheduling, optimization, distributed systems, groupware, air traffic control, automated manufacturing, space shuttle, robotics[9,12,13].However the exact nature of an agent has been the subject of extensive debate within the research community, in the sense that there is little agreement about the definition of what an agent could be and what architecture is most suitable to use[9,10,11].Though there is not a universally accepted definition of intelligent agent, there is a broad consensus that agents should have the property of autonomy. They may also have other attributes, such as **social** ability, **reactivity**, **pro-activeness**, **rationality** and **mobility**, etc [15, 16].

In the way of presenting a definition in this paper, agents can be defined as autonomous entities that act within an environment, that is, agents are free to **choose** their own actions [17].An agent is often referred to generally as an entity(piece of software) that accomplishes some tasks on behalf of its user. Agents may **react** to stimulus from their environment and are also capable of **changing** their environment. **Pro-activeness** is the ability of the agent to pursue goals and persist in trying to achieve its set goals. **Autonomy** is the ability of the agent to act without necessarily being told to take particular actions. The agent **social** attribute means its ability to team up with other agents towards the actualization of some goals. **Situated** in an environment the agent influences and is also influenced by its environment. The agent perceives its world through **sensors** and affects its world via **actuators**.

A number of different approach have been reported for modeling agent oriented systems[18,19,20,21,22,23]. One of such agent modeling paradigm reported in the literature as a mature and commonly adopted architecture is the BDI (Belief, Desire, Intention) model. The BDI agent architecture is adopted in the definition of the agent based project integration management framework carried out in this paper.

3. Project Integration Management and the need for AI techniques

Project integration management is one of the project management knowledge areas [PMBOK]. The project management knowledge areas describe project management knowledge and practice in terms of its component processes. Project **integration management** includes the processes required to ensure that the various elements of the project are properly coordinated [PMBOK]. Project integration management includes the following major processes [PMBOK]: **Project Plan Development**- taking the results of other planning processes and putting them into a consistent, coherent document. **Project Plan Execution**- caring out the project plan by performing the activities included therein. **Overall Change Control**-coordinating changes across the entire project. These processes interact with each other and with other project management knowledge areas.

In the course of **project integration management**, the project plan is progressively elaborated by updates and controlled and approved through **integrated change control**, which together recursively overlaps with project plan execution processes requiring extensive(depending on project complexities) and dynamic coordination of

information to direct the various technical and organizational interfaces in the project. Change request could unexpectedly emanate from stakeholders, previous project assumptions may have been proved wrong (requiring alterations somewhere), a project task may have been delayed requiring update to project schedule etc. Hence it could be safely said that the relationships existing among the iterative and overlapping processes that constitute project integration management are not linear. It is well known that handling situations involving non linear interactions are best addressed using intelligent techniques. This paper approaches this problem domain by reasoning within the context of intelligent agents.

A real time project integration management framework based on AI concepts is most appropriate for instance to enable the project team identify variances (in cost, budget, schedule etc.) or potential risk early, before they become serious threats to success. Early and real-time (or near real-time) notification of serious variations in project elements will definitely help in avoiding serious errors that might lead to costly rework of deliverables.

4. The BDI Agent Architecture

The underlying principles of the Belief-Desire-Intention (BDI) architecture were established in the mid-1980s based on the philosophical work by Bratman[24,25]. As reported, the foundation of this agent architecture is based on the premise that a rational entity must have three cognitive structures: **Beliefs, Desire, and Intentions**. These structures are often referred to as the agentøs mental attitudes.

Beliefs represent the agents knowledge of the world or the information it has about the state of the environment. The agents beliefs may be implemented as a variable, a database, a tuple , a set of logical expressions, or some other data structure[18].Beliefs can also include inference rules that allow forward chaining to lead to new beliefs[26]. Desire (understood to be more or less synonymous with goals) represents the motivational state of the agent, that is, what the agent is trying to achieve. However it is argued in the literature that Desire if represented all, have only a transient representation as a type of event. The desire has been referred to as the agents information about objectives to be accomplished, or priorities associated with various current objectives, either of which are thought to be generated instantaneously or functionally, and thus not requiring any state representation [18]. Goals play a central role in some of the properties of rational systems as described by BDI theories. Goals are essentially a partial state of the world which the agent has decided to achieve [27]. In computational terms, a goal may simply be the value of a

variable, a record structure, or a symbolic expression in some logic [24].

A typical BDI agent [28, 29, 30] has a so-called procedural knowledge constituted by a set of **Plans** which defined sequences of actions and test steps to be performed to achieve a certain goal or react to a specific situation [31].plan which can be viewed as special form belief represent the information about the means of achieving certain future world states and options available to the agent [30, 18]. Looked at another way, plans are sequences of actions that an agent can do to achieve one or more of its intentions [26]. Intentions represent the deliberative state of the agent, that is, which plans the agent has chosen for eventual execution. Events are triggers for reactive activity by the agent, and may update beliefs, trigger plans or modify goals [26]. Events can be generated externally by other sources or internally by the agent or other agents. Internal events are those that are created by the agent itself. This allows other plans that the agent may have to be invoked. External events are those which are not created by the agent and may be created by either the environment or by other agents. When an event is perceived, the agent may choose to execute certain plans.

5.   The design of the framework

The framework consists of a number of cooperating intelligent agents designed to ensure dynamism in the capture of the vast details (e.g relating to change request, performance reports, information on authorized work etc.) that is continuously generated by the processes required in project integration management. T his enables the dynamic communication of project information and work instruction to project team members e.g. via automated e-mail, sms, alerts etc. This intelligent infrastructure possesses the dynamism to intelligently react to changes (e.g. in project scope, budget, schedule etc.) in the environment thereby ensuring proper synchronization of information between the different project integration management processes in real-time. This provides reliable and consistent update of required communication to project stockholders. The major components of the framework are independent agents, each providing unique services for successful project integration management. Based on the requirement for project integration management and the communication and synchronization requirement within the framework, the following agents are identified for the system: **Plan Development Agent, Project Execution Agent, Change Control Agent and Interface Agent.**

5.1.   Defining the BDI structure for the framework

For the description of the complete BDI (agents mental attitude) to assign to the agents identified, the software development process to capture system requirement in the sequence of **Intentions, Desire, and Belief** [24, 32, 33] is utilized. This model specifies that **goals** are extracted from system requirements, Plans (**Intentions**) from activity diagrams, and **beliefs** (data moving in the environment) from data flow diagrams. In the case of the proposed framework, from overall requirement for project integration management, the goals(outputs from each constituent process in project integration management) are discovered, then the intentions(project management activities that should be executed when certain goals are reached) are identified, and the beliefs(information from subsidiary project planning activities, informational output from other project integration management process) that will be necessary for each goal to be completed are extracted. From this the completed BDI structure to assign to each agent in the framework is defined.

5.2.   The Plan Development Agent

Fig.1 (inspired by [34]) shows the BDI structure of the Plan development Agent. The agent is shown alongside other agents in the framework to clarify data exchanges (that also alter the agentsø mental attitude) between the agents. It also shows overlap of processing activities among   the agents as required for successful project integration management.

Beliefs: the beliefs of the plan development agent are (1) the beliefs as indicated in fig.1 - this beliefs stem from other subsidiary planning activities in the project (2) project management plan update ó this stem from the execution of intentions (plans) in the Project Execution Agent and the execution of intentions in the Change Control Agent. **Intentions**: the possible intentions of the Plan Development Agent are as indicates in fig.1. The set of plans to execute embodies procedural knowledge involved in the control of project scope, control of schedule, performing quality control, controlling cost etc. The agent can exploit inference rules within a knowledge-base while deliberating on these activities The execution of the plans in the intention list is to ensure verification, control and update actions that ensure the coherent coordination of all subsidiary planning processes and resulting documentations. The intentions are sets of algorithms that are iteratively processed as project planning activities are carried on. These activities are executed when goals are reached. **Goals:** The goal the

Plan Development Agent is trying to achieve is the **project management plan.**

The project plan being a document that is progressively elaborated by updates and controlled and approved through integrated change control process[PMBOK], means that the Plan Development Agent has to continuously observe(or probe its environment) for the next percept. The agent will continue to process its belief, execute actions iteratively in order to achieve its intentions based on a set of algorithmic steps in its intention structure. For instance, during the planning process of project integration management, to achieve the intention **control scope,** the agent would have to execute

Furthermore a scope change would mean update to project plan (hosted in Project Management Information system-PMIS). This means the agent would have to update its beliefs, computationally which entail executing steps to update the project management plan. The agent sends instructions to update an external PMIS. The agent is motivated by the attainment of its goals. The attained goal further constitutes events that trigger deliberation (intention execution) in the other agents in the framework. The goals/intentions further create/update the agent beliefs (referring mainly to the updates to the project plan).

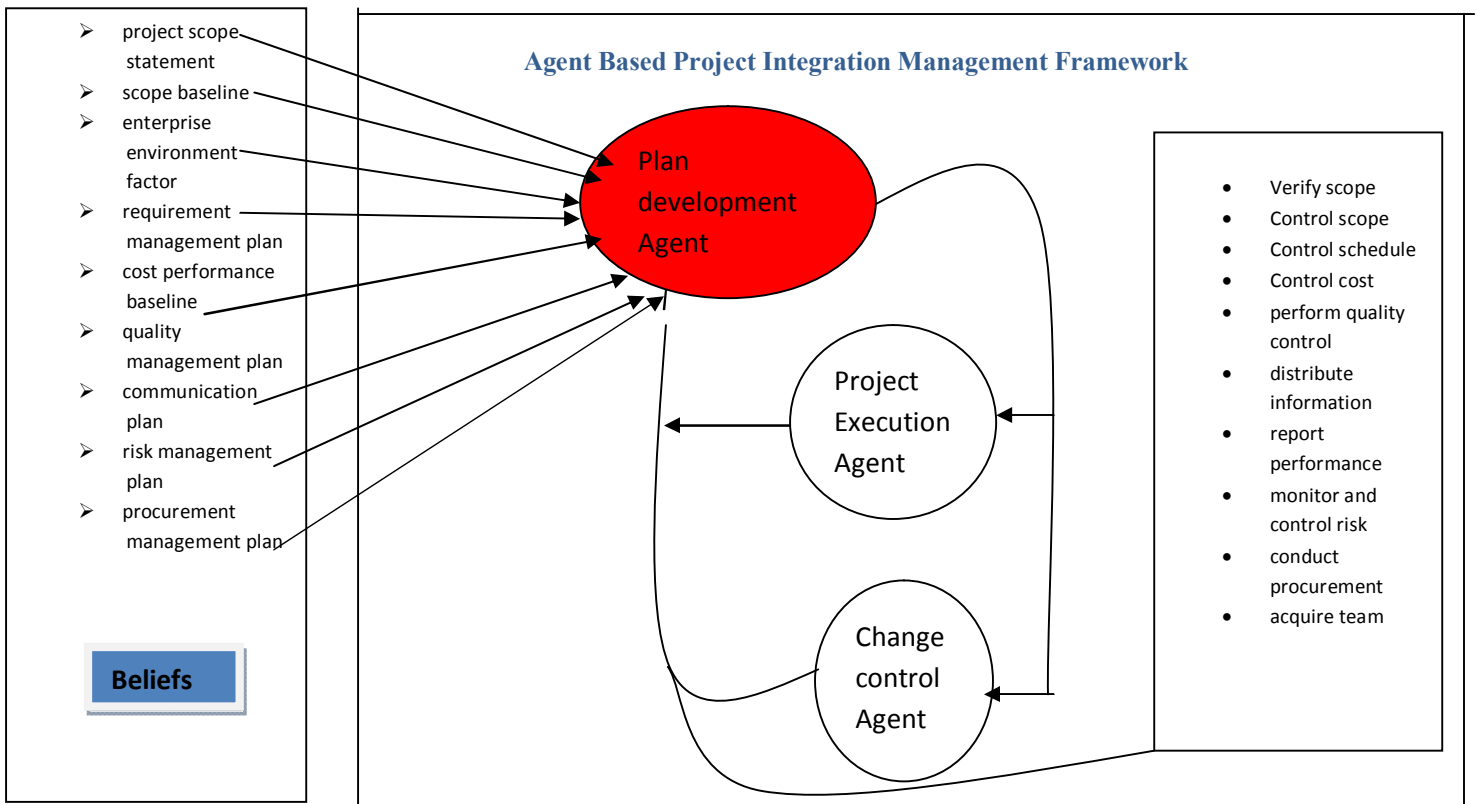**Fig.1** BDI structure of the Plan Development Agent

steps(algorithms) to ascertain whether inputs from subsidiary planning activities are within the project scope.

5.3. **The Project Execution Agent**

Fig.2 shows the BDI structure of the Project Execution Agent. Like fig.1 it shows overlap of processing activities

among the agents as required for successful project integration management.

Beliefs: referring to fig.2, the data(arrows) entering the agent constitutes its beliefs. These include: Project management plan, Approve change request, signals from the enterprise/performing organization coming via the interface agent.

Intentions: the possible intentions (change control, schedule control, perform quality management etc.) are indicated in the figure.

Goals: work performance information, deliverables, project document updates, change request, work performance information (indicated by signals moving out of the agent).

The Project Execution agent continuously probe or query the project management information system (i.e. the environment in order to obtain belief from its store or from other agents.) as project execution advances.If the agent arrives at project integration management goals, that is finds **deliverables, project document updates** made, etc , it commits to the execution of some intentions. For instance if the agent arrives at the goal *deliverable*, the agent executes the steps in the intention *perform quality control*; possibly if the agent arrives at the goal work *performance information* for example, the steps in the intention *control scope, control schedule, control cost, perform quality assurance* are executed. Based on such executions, a change request might be sent to the **change control agent**. Execution of steps in the intention *report performance* might entail the agent sending messages to the **interface agent** to send e-mail to project stakeholders, comparing work output information with baselines to measure the variance, updating the key performance indicator document etc. Based on the appropriate goal being reached, the execution of steps in the intention *control cost* might entail running algorithms to verify whether expended cost are within the variance as specified in the **cost management plan**(a subsidiary project management plan[PMBOK]).
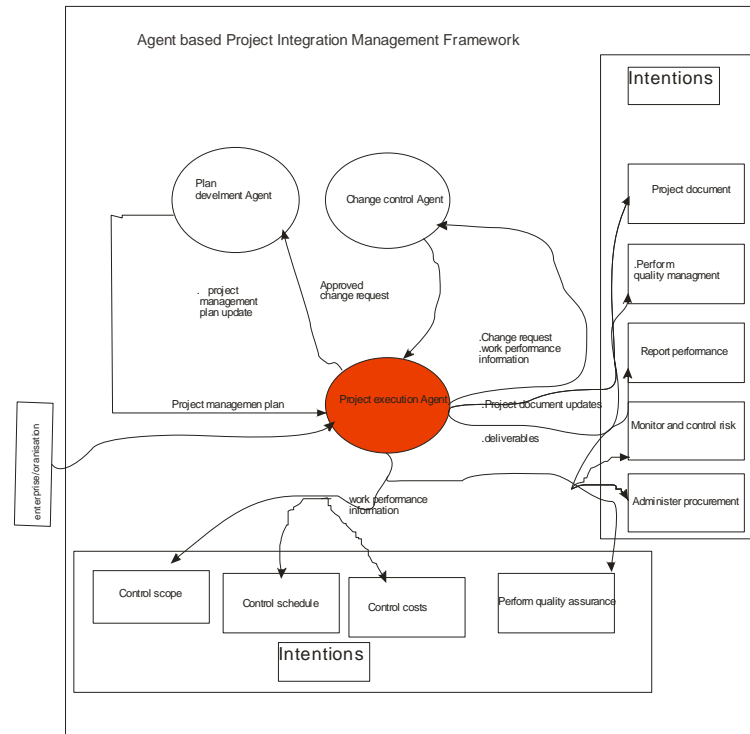

Fig.2.BDI structure of the Project Execution Agent

5.4 The Change Control Agent

Goals: project management plan update, project document update, change request **Beliefs:** project management plans, change request, enterprise/organization assets/factors, information from the execution of intention within other agents in the framework. **Intentions**: perform quality control, documentation, escalate change request to change control system.

The change control agent monitors its environment continuously for events. If its beliefs change, for instance if there is a change request, it commits to the execution of its set of intentions. Change request can come from other agents (from execution of intentions by the **plan development agent** or **project execution agent**, from the environment (from the project team or stakeholders). The change control agent does not approve changes. It needs human input. The **change control agent** routes any change request to a Change Control System (CCS). The CCS is in this case an external system. This is the performing organizations change control system. In some organizations the CCS is called the change control board or the technical review board. This is a group of people responsible for approving or rejecting changes to a

project. The routing of the change request to the CCS for approval might take the form of file update, database entry, or e-mail etc. If the agent perceives an approved change event from the CCS the agent executes intention steps to update the project plan, update the project baselines. This might have the effect of triggering the execution of steps in the intention *perform quality control.*

5.5 The Interface Agent

This agent manages user interaction with the framework i.e. handles message commutation between the framework and external entities. For instance it enables the project manager to assign support task to the framework.

Conclusion

This paper has focused on the task of explicitly representing the complex iterative and overlapping information interchanges and processing among the Project Integration Management processes in the form of BDI structure, providing the basis for the development of agent based systems to support project integration management. The work is still subject to further development especially in the area of collab oration and negotiation protocols for agent-agent interaction (and possibly human agent teaming technique), leveraging with project management principles and practice. Further step in this research would be to further develop the framework towards full implementation.

References

[1] Anand S. Rao, Michael P.Georgeff. *BDI Agents. From theory to Practice.* Technical Notes 56 April, 1996.

[2] Anand S. Rao and Michael P. Georgeff. *An Abstract for Rational Agents.* In W.Swartout c.Rich and B. Nebel, editors, Proceedings of the third International Conference on *Principles of knowledge Representation and Reasoning(KR'92),* San Mateo, C,1992.Morgan Kautmann Publishers.

[3] Anan S. Rao.*AgentSpeak(L):BDI Agents Speak out in a logical computational language.* In MAAMAW 96:7th European Workshop on *Modeling Autonomous Agents in a Multi-Agent World*, LNAI 1038. Springer-Verlag, January 1996.

[4] A.S.Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture . In J. Allen, R.Fikes, and E.Sandewall, editors, *Proceedings of the Escond International Conference on Principles of Knowledge Representation and Reasoning. Morgan Kaufmann Publishers, San Mateo, CA, 1991*

[5] Avots. Ivars, *Application of expert systems concepts to schedule control* Project Management Institute.

[6] Bratman, Michael E., *Intention, Plans,and Practical Reason,* Harvard University Press: Cambridge, MA, 1987

[7] Chang-Hyung Jo, Guobin Chen, James Choi,*A framework for BDI agent-based software engineering.* Studia informatica Universalis

[8] Einhorn, Jeffery m. and Jo,Chang-Hyun, *A Use-Case Based BDI Agent Software Development Process,* The 2nd International Workshop on *Agent-Oriented Methodologies,* 7-20, 00pla 2003, Anaheim, CA, USA, 2003

[9] Guoying Liu. *The application of intelligent agents in libraries: a survey.* Leddy Library Publications, University of Windsor.

[10] Hossein Iranmanesh, Mahboubeh Madadi . *Framework of intelligent Systems to support Project scope.* University of Tehran, Teehran

[11] Hosley, W N,(1987).*The application of artificial intelligence software to project management.*1987. Project Management Institute.

[12]H.S.Nwao. *Software agents. An Overview.* Knowledge Engineering Review, II(2):205-244,Oct.1996

[13] Jennings, N. and Wooldridge, M.(1998). *Applications of Intelligent agents.* In Jennings, N.R. and Wooldridge, M.J., editors, *Agent Technology: Foundations, applications, and Markets, Chapter 1, pages 3-28.*Springer.

[14] John Thangarajah, Lin Padgham, James Holand. *Representation and Reasoning for Goals in BDI Agents.* RMIT University Melbourne Austrialia.

[15] Jo.Chang-Hyun and Arnold, Allen J.,. *The Agent-based programming Languages*: APL,ACM AC202, 27-31, Maddrid, Span.

[16] J.Doyle.*Raionality and its role in reasoning.* Computational Intelligence, 8(2)376-409, 1992.

[17] K.C.Lam,D.Wang,Patricia T.K. Lee, Y.T. Tsang.(2007). *Modeling risk allocation decision in construction contracts.* International Journal of Project Management 25,2007, 85-493.

[18] K.C. Iyer, N.B. Chaphalkar, G.A. Joshi.(2008),*Understanding time delay disputes in Construction contracts.* International Journal of Project Management 26,2008,174-184

[19] K.Ghedira.MASC: *une approche Mult-Agents de Problĕmes de statifaction de contraintes* .PhD thesis, 1993.

[20] Maria-Athina Mountzia.*an Intelligent-Agent based Framework for Distributed Systems Management.* Department of computer science Technische Universitat Munich,Germany

[21] M.Wooldridge, J.P. Muller, and M.Tambe. *Agent theory, architecture, and languages:* A bibliography. In.M.Wooldridge, J.P. Miller, and M. Tame, editors, *Intelligent Agents II,* Volume 1037 of *Lecture Notes In Artificial Intelligence,* Pages 408-431 Springer Verlag, 1996.

[22] Mohammad A. Rajabi. *Chapter 4 Project Integration Management(PMBOK guide)* University of Thram:http://www.marajabi.com

[23] M.E.Bratman, D.Israel, and M.E. Pollack. *Plans and resource bounded practical reasoning.* Computational Intelligence, 8(2):377-409, 1992.

[24] M. Wooldridge. *An introduction To Multi-Agent Systems.* John Wiley and Sons Ltd., first edition 2002.

[25] M.P.Georgeff and F.F. Ingrand. *Decision-making in an embedded reasoning system.* In Proceedings of the International Joint conference on Artificial Intelligence, Detroit, Mi. USA, 1989

[26] Nemati H, Todd, D.W. Brown, Paul D,(2002).*A hybride intelligent system to facilitate information system project management activities,*2002, Project Management Institute.

[27] Nordin BY, Babcock, DL, Colin DB,(1990).*Development of a knowledge-based schedule planning system.*,1990, Project Management Institute.

[28] Paolo Busetta, James Bailey Kotagiri Rammamohanarao *A Reliable Computational Modeling For BDI Agents* . Univerity of Melbourne, Australia.

[29] Poole, D., Mackwrith, A. and Goebel, R.(1998), *Computational Intelligence; A Logical Approach,* Oxford University Press, New York, NY.

[30] S.J.Rosenschein and L.P. Kaelbling.*The synthesis of digital machines with provable epistemic properties.* In J.Y. Halpern, editor, *Proceedings of the First Conference on theoretical Aspects of Reasoning about Knowledge,* San Mateo, CA, 1986. Morgan Kaufmann Publications.

[31] The Noranis Mohd Aris. *Object-Oriented programming semantics representation utilizing agents.* Journal of theoretical and Applied Information Technology(2005). ISSN:1992-8643.

[32] Wooldridge, M., Jennings, N.R.(1995).*Intelligent agents: theory and practice,* Knowledge Engineering Review, Vol. 10 No. 2, pp. 115-152

[33] Xiaoqing(Frank) L, Gauttam Kane, Monu Bamboo(2006). *An intelligent early warning system for software quality improvement and project management.* 2006, Journal of Systems and Software, Volume 79, Issue II 1552-

[34]Y.Shohan. *Agent-Oriented Programming. Artificial Intelligence, 60*(1),:51-92, 1993.

SESSION

INTERNATIONAL WORKSHOP ON INTELLIGENT LINGUISTIC TECHNOLOGIES - ILINTEC'13

Chair(s)

Dr. Elena Kozerenko

Method for Extracting Translation Correspondences from a Parallel Corpus[i]

Yu. Morozova

Institute for Informatics Problems of the Russian Academy of Sciences, Moscow, Russia

Abstract – *The research paper deals with actual problems of semantic studies using methods of corpus linguistics. It overviews a new research area - distributional semantics. The method for extracting translation correspondences from a parallel corpus using context vector spaces is described. The model was tested on a parallel corpus of patent texts in Russian and French. An approximate evaluation of precision is 78%. False positive results can be explained by two main reasons. In the first place, productive syntactic transformations result in translation correspondences containing two words which are semantically equivalent but belong to different parts of speech. In the second place, words are often part of multi-word expressions and in such cases a correct single-word translation correspondence cannot be found. We propose to enhance existing models by moving from single lexemes to multi-word expressions.*

Keywords: distributional semantics, vector spaces, multi-word expressions, collocations, parallel corpora.

1 Distributional Semantics Models Overview

Distributional semantics is a field of linguistic research that aims at calculation of semantic proximity between different linguistic units using their distributional properties in large linguistic corpora. Distributional models are used in numerous research projects dealing with semantics of natural language and have a diverse range of potential and working applications. Main application areas of distributional semantics models are: lexical ambiguity resolution, information retrieval, document clustering, automatic extraction of lexicographic information (dictionaries of semantic relations, multilingual dictionaries, semantic maps of different subject areas), modeling of synonymy, document topic detection, sentiment analysis, bioinformatics.

Theoretical foundations of distributional semantics go back to the distributional methodology proposed by Z. Harris [1, 2]. Similar ideas were expressed by the founders of structural linguistics F. de Saussure and L. Vitgenstein. The theoretical basis of distributional models is the distributional hypothesis stating that linguistic units with similar distributions have similar meanings [3, 4].

Linear algebra is used as the computational instrument and as the means of model representation. First the information on linguistic units distribution is represented in the form of multidimensional vectors. These vectors constitute a matrix, in which vectors correspond to linguistic units (words or word combinations) and dimensions correspond to contexts of different sizes (documents, paragraphs, sentences, word combinations, words). When a matrix is populated from texts, semantic proximity between linguistic units can be calculated as the distance between vectors.

To compute the distance between vectors one can use various formulas: Minkowski distance, Manhattan distance, Euclidean distance, Chebyshev distance, scalar product, cosine measure. The most widely used formula is the cosine measure:

$$\frac{x \bullet y}{|x| \bullet |y|} = \frac{\sum_{i=1}^{n} x_i \bullet y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \bullet \sqrt{\sum_{i=1}^{n} y_i^2}} \qquad (1)$$

where x_i and y_i represent frequency counts for different dimensions of the vectors.

There are many different types of distributional semantics models which differ according to the following parameters:

- type of the context (its size, left or right, ranking);
- measure used to calculate the frequency of a word in a given context (absolute frequency, entropy, mutual information etc.);
- method used to compute the distance between vectors (cosine measure, scalar product, Minkowski distance etc.);
- method used to reduce matrix dimensionality (Random Projection, Singular Value Decomposition etc.).

The most popular distributional semantics models are Latent Semantic Analysis which was designed to solve the synonymy problem in information retrieval [5], and the model of Hyperspace Analogue to Language created as the model of human semantic memory[6].

The idea of semantic vector spaces was first realized in the information retrieval system SMART[7]. Documents from a text collection are represented as vectors in a vector space. A user inquiry is viewed as a pseudodocument and

is also represented as a vector in the same vector space. The system finds n vectors of documents which are closest to the vector of the inquiry. The results are sorted by distance between vectors which reflects semantic proximity and shown to the user.

Later on the idea of semantic vector spaces was applied successfully to other semantic tasks. For example, in the research [8] a vector space was used to evaluate semantic proximity of words. The system reached the accuracy level of 92.5% on TOEFL tests to choose a synonym out of a set of words, average human result for this test being 64.5%.

At the present time there are many research projects aimed at unifying the model of vector spaces and working out a common approach to different tasks of detection of semantic relations from text corpora [9].

2   Alignment Models Overview

The task of finding single-word translation correspondences from parallel texts is part of the alignment problem and is discussed in numerous papers. In [10] an alignment is defined as an arbitrary relation between source and target language words (including correspondences of one word to an "empty" word and of one word to several adjacent words). However, the development of alignment models that are able to deal with this general model is difficult. Most often there are additional constraints to the general definition of alignment. Usually each source word is assigned to exactly one target word (including empty words). Some papers propose to add linguistic knowledge to alignment models that is used to filter out incorrect alignments.

The alignment models were first implemented by Brown and colleagues [11]. They use aligned parallel corpora and establish a translational relation between terms that occur with similar distributions in corresponding text segments.

There are two types of alignment models: heuristic models and statistical alignment models. Heuristic models are more widely used by researchers as they are easier to understand, implement and tune.

Heuristic models use a function of the similarity between words of two languages. Most often variations of the Dice coefficient are used as the similarity function:

$$dice(i, j) = \frac{2 \bullet C(e_i, f_j)}{C(e_i) \bullet C(f_j)}$$

(2)

where *f, e* are the source word and the target word, *C(e,f)* is the co-occurrence frequency of *e* and *f* in the parallel corpus, *C(f)* is the frequency of *f* in the source sentences, *C(e)* is the frequency of *e* in the target sentences.

For each sentence pair a matrix with association scores between every source word and every target word is built. Then the word with the largest association score is chosen as the translation correspondence for a given word.

Within the statistical approach to alignment, the translation probability $\Pr(f_1^J \mid e_1^I)$ which describes the relationship between a source language string f_1^J and a target language string e_1^I is modeled. Statistical models depend on a set of unknown parameters Θ that is learned from the corpus. For each sentence pair (f_s, e_s), the alignment variable is denoted by $a = a_1^J$. The unknown parameters Θ are determined by maximizing the likelihood on the parallel training corpus:

$$\Theta = \arg \max \prod_{s=1}^{S} \sum_{a} p_{\Theta}(f_s, a | e_s)$$

(3)

The expectation maximization algorithm is typically used to perform this maximization.

The paper [10] provides a comparison of error rate percentages for different alignment models. Heuristic models give the best result of 21.5% error rate and statistical models give the best result of 16.4% error rate for the training corpus of 0.5 K.

The papers [12, 13] introduce the cognitive approach to alignment using semantics of language units. A new grammar formalism called Cognitive Transfer Grammar (CTG) is described. The basis of CTG is composed of the proto-typical language structures, their most probable positions in a sentence, statistical data about the distributive characteristics of structures, the schemes of the complete parse of sentences.

In CTG the functional values of language structures are determined by the categorial values of head vertices. Probability characteristics are introduced into the rules of derivation in the form of the weights assigned to the parse trees.

The cognitive approach to alignment is based on the principle « from the meaning to the form ». It establishes correspondences between structures belonging to different language levels, for example : word → word, phrase structure → phrase structure, word → phrase structure, morpheme → word, morpheme → phrase structure.

The elementary structure of CTG is a *transfeme*. A transfeme is a unit of cognitive transfer, i.e. a semantic element embodied in a translatable semantically relevant language segment taken in the unity of its categorial and functional characteristics, that establishes semantic correspondence between language structures, which belong to different language levels. The types of transfemes are determined by the rank of transfemes:

- rank 1: lexemes as structural signs, i.e., a word, considered as a categorial - functional unit without taking into account the specific lexical value of this word;

- rank 2: a word combination, i.e., the syntactic structure, which consists of two and more syntactically connected words, but never a complete sentence (clause);
- rank 3: a clausal unit, i.e., dependent (subordinate) clause;
- rank 4: a sentence (either a simple sentence or the main clause of a complex sentence);
- rank 5: a scattered structure, i.e., a word group, which is characterized by a syntactic and semantic unity, but is discontinuous, i.e., between the members of the group there appear other language objects, which are not the members of this group;
- rank 0: the morphological units, which are not independent words, but which form a part of a lexeme of a source language, and in the language of transfer can be expressed by a clause and the units of other ranks.

Transfemes are represented as the rewriting rules in which a nonterminal symbol is in the left side and right sides contain the aligned pairs of chains of terminal and nonterminal symbols belonging to the source and target languages:

$$T \rightarrow\; <\rho\,,\,\alpha,\,\square> \qquad (4)$$

where *T* is a nonterminal symbol (transfeme), ρ and α are the chains of the terminal and nonterminal symbols which belong to Russian and English, $\square$ is the symbol of correspondence between the nonterminal symbols occuring in ρ and with the nonterminal symbols occurring in α. During the alignment of parallel texts on the basis of CTG the process of derivation begins from the pair of the connected initial symbols Sρ and Sα, further at each step the connected nonterminal symbols in pairs are copied with the use of two components of uniform rule.

The linguistic knowledge base described in [13] comprises the following components :
- the initial basic collection of grammar rules represented in the formalized form (CTG);
- the mechanisms of expansion and refinement of the system of rules, implemented by means of the methods of machine learning on parallel texts.

The CTG allows to automatically extract syntactical translation rules from parallel texts. Texts need to be aligned on the level of sentences and on the level of words before the rule extraction module can start working.

Different types of phrase structures are described as functional meanings together with their categorial embodiments. The transferability of phrase structures is possible when language units belonging to the same functional transfer fields (FTF) are chosen in the source and the target languages, notwithstanding the difference of their syntactic categories. The most important FTF are the following:

- Primary Predication FTF (complexes of finite verbal forms and tensed verbal phrase structures);
- Secondary Predication FTF (non-finite verbal forms and constructions, subordinate clauses comprising the finite verbal forms);
- Nomination and Relativity FTF (language structures performing the nominative functions, including the sentential units);
- Modality and Mood FTF (modal verbs and word combinations expressing modality, subjunctivity and conditionality);
- Connectivity FTF (lexical – syntactic means employed for concatenation of similar syntactic groups and subordination of syntactic structures);
- Attributiveness FTF (adjectives and adjectival phrases, nominal modifiers of different kinds);
- Metrics and Parameters FTF (language means for presenting entities in terms of parameters and values, measures, numerical information).

3 Extracting Translation Correspondences from Parallel Texts

Our research is aimed at implementing the model of semantic vector spaces to extract single-word translation correspondences from parallel texts.

The paper [14] describes a method for applying distributional semantics models to extract translation correspondences of single terms from aligned parallel texts. In general, systems extracting translation correspondences use the co-occurrence frequency of terms in the source and the target language in aligned segments as the basis for their work. The authors of [14] propose to use sentences rather than words as a minimal unit for analysis as "the primary meaning bearing unit is the utterance, the coherent expression of something meaningful by a speaker or a writer". Lexical units occurring in the same sentence are linked by syntagmatic relations and the sentence in the source language as a whole is related to the sentence in the target language by the relation of translation correspondence. Thus each word in the source sentence is related to each word in the target sentence.

In the model proposed in [14] the identification numbers of aligned regions are used as dimensions of vectors. Context vectors describing words of source and target languages are put in the same vector space. To compute the correlation between words the cosine measure is used. Words of different languages whose vectors are closest to each other are considered to be translations of each other. This approach is especially efficient when one needs to find not only the best translation but several ways to translate a term.

Within the framework of our research a test corpus of parallel texts in French and Russian aligned at the level of sentences was created. It comprises texts of scientific

patents from different areas. The volume of the corpus is 100000 word tokens. The texts were uploaded into the online corpus management system Sketch Engine [15] which provided morphological annotation of texts (lemmas, parts of speech and grammatical characteristics).

We developed a vector space model for extracting single-word translation correspondences and tried its work on the test corpus. The model is characterized by the following parameters:

- type of linguistic units: single terms;
- type of context: aligned regions;
- frequency measure: absolute frequency;
- method used to compute the distance between vectors: cosine measure.

The computer program realizing this model was implemented by Charnine M. M. [16].

Before populating the vector space we preprocessed the texts in the following way:

- words were replaces by lemmas
- the most frequent words were removed (mainly functional words such as prepositions and conjunctions)
- punctuation marks were removed.

In the result of application of the vector space model we received a list of single-word translation correspondences, for example:

moyen ("means") → средство ("means")

caractériser ("to characterize") → отличать ("to characterize")

moins ("less") → мера ("measure")

notamment ("in particular") → частность ("detail").

In many cases a translation correspondence contains two words which are semantically equivalent but belong to different parts of speech, for example:

connaître ("to know") → известный ("well-known").
Words of different parts of speech can translate each other in certain contexts but such word pairs in general should not be included in a translation dictionary. Thus the results produced by the vector model need to be filtered so that only words with the same category are left.

At the same time translation correspondences containing words of different syntactic categories provide an interesting by-product as examples of this kind correspond to productive syntactic transformations occurring during translation.

The most frequent transformations of syntactic categories for patent texts are the following:

- verbal infinitive (French) → noun (Russian)
  For example,
  traiter ("to process") → обработка ("processing").
- noun (French) → adjective (Russian)
  For example,
  crochet ("hook") → крюкообразный ("hook-shaped").

- verbal infinitive (French) → adjective (Russian)
  For example,
  connaître ("to know") → известный ("well-known").

Information about shifts of syntactic categories is a useful resource for development of a syntactic transfer module of a Machine Translation System.

On the other hand, words are often part of multi-word expressions for which the number of words is different in two languages, for example:
au moins ("at least") → по меньшей мере ("at least").
In such cases it is not possible to find a correct translation correspondence using the existing vector space model.

We envisage enhancing the current vector space model by moving from single words to multi-word expressions or collocations. By collocations we mean statistically stable word combinations. Computational linguists use different statistical measures to extract collocations from texts. As it is stated in the paper [17], the Mutual Information Measure (MI) gives the best results on average. It can be computed using the following formula:

$$MI = \log_2 \frac{f(n,c) \times N}{f(n) \times f(c)} \qquad (5)$$

where n is the first word of a collocation; c is the second word of the collocation, f(n,c) is the absolute frequency of two words occurring together, f(n), f(c) are the absolute frequencies for each single word, N is the size of the corpus in tokens.

Using MI on the texts from the test corpus we compiled a dictionary of multi-word expressions in Russian and French for the subject area of scientific patents. The following step will be to adjust the vector space model so that it can find translation correspondences of collocations from parallel texts.

4   Conclusions

The paper overviews main research areas and models of a new linguistic discipline – distributional semantics. Multi-dimensional matrices of linear algebra are used as the mathematical model, which represent a suitable formalism for computer realization. Using distributional models it is possible to automatically compile different linguistic resources on the basis of large corpora: semantic dictionaries, translation dictionaries, semantic maps of subject areas.

The present research aims at implementing distributional approach for compiling a dictionary of translation correspondences on the basis of a parallel corpus. The task of finding single-word translation correspondences from parallel texts is part of the alignment problem. The paper overviews different approaches to alignment: heuristic models, statistical alignment models, cognitive approach. Heuristic models are used most widely

as they are easy to implement. The best practical results are obtained using statistical alignment models. The cognitive approach to alignment aims at establishing correspondences between structures belonging to different language levels, for example : word → word, phrase structure → phrase structure, word → phrase structure, morpheme → word, morpheme → phrase structure. The Cognitive Transfer Grammar described in [12, 13] makes it possible to automatically extract syntactical translation rules from parallel texts.

In the result of the present research a vector space model for extracting single-word translation correspondences from a parallel corpus was implemented and tested on a parallel corpus of patent texts in Russian and French. The results of testing the model can be divided in the following groups:

- correct correspondences (semantically equivalent words of the same part of speech);
- semantically equivalent words of different parts of speech;
- fragments of multi-word expressions;
- erroneous correspondences.

Correct correspondences constitute 78% of results. Translation correspondences containing words of different syntactic categories correspond to productive syntactic transformations occurring during translation. This information can be used for development of a syntactic transfer module of a Machine Translation System. Fragments of multi-word expressions cannot be processed correctly using the current vector space model. Multi-word expressions (also called collocations) occur frequently in texts so it is necessary to include collocations into the model.

The first step consists in extracting collocations from texts in Russian and in French independently. This can be realized using statistical measures of association. The Mutual Information Measure (MI) gives the best results on average [17]. Using MI on the texts from the test corpus a dictionary of multi-word expressions in Russian and French was compiled. The next step will consist in including multi-word expressions in the vector space and finding translation correspondences between them by computing the distance between vectors.

5   References

[1]   Harris Z. S. "Papers in Structural and Transformational Linguistics". D. Reidel., 1970.

[2]   Harris Z. S. "Mathematical Structures of Language". Interscience Publishers John Wiley & Sons, 1968.

[3]   Sahlgren M. "The Distributional Hypothesis. From context to meaning"; Distributional models of the lexicon in linguistics and cognitive science (Special issue of the Italian Journal of Linguistics), Vol. 20, Issue 1, 33-53, 2008.

[4]   Turney P. D., Pantel P. "From frequency to meaning: Vector space models of semantics"; Journal of Artificial Intelligence Research, Vol. 37, 141-188, 2010.

[5]   Landauer Th. K., McNamara D. S., Dennis S., Kintsch W. "Handbook of Latent Semantic Analysis". Lawrence Erlbaum Associates, 2007.

[6]   Lund K., Burgess C. "Producing high-dimensional semantic spaces from lexical co-occurrence"; Behavior Research Methods, Instruments & Computers, Vol. 28, Issue 2, 203-208, 1996.

[7]   Salton G. M. "The SMART Retrieval System: Experiments in Automatic Document Processing". Prentice-Hall, 1971.

[8]   Rapp R. "Word sense discovery based on sense descriptor dissimilarity"; Proceedings of the 9th MT Summit, 315–322, 2003.

[9]   Turney P. "A uniform approach to analogies, synonyms, antonyms and associations"; Proceedings of COLING, 905–912, 2008.

[10]   Franz Josef Och, Hermann Ney. "A Systematic Comparison of Various Statistical Alignment Models"; Computational Linguistics, Vol. 29, Issue 1, 19-51, 2003.

[11]   Brown P. S., Cocke V., Della Pietra F., Della Pietra F., Jelinek R., Mercer, Roossin P. "A statistical approach to language translation"; Proceedings of the 12th Annual Conference on Computational Linguistics, 1988.

[12]   Kozerenko E.B. "Parallel Texts Alignment Stratagies: The Semantic Aspects"; Informatics and its Applications, Vol. 7, Issue 1, 82-89, 2013.

[13]   Kozerenko E.B. "Syntactic Transformations Modelling for Hybrid Machine Translation"; Proceedings of WORLDCOMP'11, 875-881, 2011.

[14]   Sahlgren M., Karlgren J. "Automatic Bilingual Lexicon Acquisition Using Random Indexing of Parallel Corpora"; Journal of Natural Language Engineering (Special Issue on Parallel Texts), Vol. 11, Issue 3, 2005.

[15]   Web site for Sketch Engine: http://www.sketchengine.co.uk/

[16]   Kuznetsov, I.P. Elena B. Kozerenko, Mikhail M. Charnine. "Technological peculiarity of knowledge extraction for logical-analytical systems"; Proceedings of ICAI'12, WORLDCOMP'12, July 18-21, 2012, Las Vegas, Nevada, USA. CRSEA Press, USA, 2012.

[17]   Zaharov V.P., Hohlova M.V. "The Analysis of Efficiency of Statistical Methods of Collocations Detection for Texts in Russian"; Computational Linguistics and Intelligent Technologies (International Conference "Dialog"), 2010.

Knowledge-based and vertical-driven information retrieval

Gavin Matthews
Vertical Search Works
gmatthews@verticalsearchworks.com

Max Petrenko
Vertical Search Works
mpetrenko@verticalsearchworks.com

Abstract – *The paper introduces the architecture and functionality of the knowledge-based information retrieval technology developed at Vertical Search Works. A large-scale language-independent ontology is used during indexing, query analysis, and document retrieval as part of a web-scale vertical search engine. Three specific areas are examined: the knowledge resource, its visualization and editing toolbox, and ways of fine-tuning the ontology to enable intrinsic consistency checks will be discussed; using the knowledge resource for indexing and query analysis purposes, concentrating on how the ontology-driven semantic latching and disambiguation module tackles challenges arising from ambiguous input; tools for evaluating the performance of knowledge-based information retrieval available at Vertical Search Works.*

Keywords: information retrieval, ontology engineering, disambiguation, vertical search engine

1   Introduction

Vertical Search Works (VSW) maintains and develops the Excalibur semantic web search engine (inherited from Convera) and the Editorial Related Advertising system (inherited from FirstLight ERA). Together these use semantic and linguistic processing of web pages to provide vertical search portals and context-sensitive advert placement. Behind the scenes, Excalibur relies on a large scale general-purpose ontology (loosely informed by WordNet), extended for the various verticals that are supported. The ontology includes both subsumption and lateral relations, supports inheritance and reasoning, and is used to latch and disambiguate concepts within unstructured natural language text.

Matching adverts to articles is handled essentially as a search problem, where the query is the article on which the adverts appear, and the search corpus is the set of available adverts including landing page, ad title and description, and any other available meta-information. The task is slightly different because the key information from the article is not the content *per se*, but what it implies about the likely demographic of the reader. This means that the interpretation of articles and adverts is asymmetrical.

2 Structure and management of knowledge resources

The section will outline the structure of knowledge resources maintained by Vertical Search Works and discuss engineering, visualization and maintenance principles as well as maintenance and quality assurance tools developed at Vertical Search Works.

2.1 VSW knowledge resources

The word "synset" is used here to refer to concepts represented in the ontology that may be referenced in text (the name is inspired by WordNet's synonym sets). Structurally, the ontology features a language-independent layer of concepts, a language-specific layer of expressions, topic-specific facets, and domain-specific verticals. Having verticals in the ontology allows concepts to be clustered without restructuring the core ontological hierarchy. Vertical clustering thus adds to the flexibility and adaptability of the knowledge resources to different applications.

Like many general purpose ontologies including SUMO, DOLCE, Cyc, OntoSem (see [5], [1], [3], [4]), the VSW ontology has an upper ontology of entities, processes, attributes and relations, with further mid-level subclasses, augmented with lateral links (e.g. has theme, acts in, is the author of, is an accessory for). To maintain an acceptable level of abstraction during document retrieval, top-level ontological branches are inaccessible to the disambiguation and indexing modules. There are over half a million known concepts, including not only collections, but many individual known entities.

The ontology incorporates a lexicon of over two million expressions. Each expression is annotated with

a language, sensitivity to stemming and folding (case, diacritic, and punctuation), and other such restrictions. Most expressions are in English or are universal (language independent labels like proper names).

The expressions in the VSW lexicon are primarily direct natural language, but some are composed of other synsets to reflect semantic compositionality in language. An expression can incorporate by reference either all synonyms for a synset, or all synonyms for the synset and its subsumed descendants. For example, one of the expressions on the concept for "wood floor" is "$$cpr.005QO $cpr.000ZW". Here "$$cpr.005QO" denotes the concept of wood and all its children (e.g. oak, pine), and "$cpr.000ZW" denotes synonyms of "floor" (e.g. "plancher" in French, "piso" in Spanish). Compositional expression can mix synsets and words; another expression on "wood floor" is "$cpr.000ZW made of $$cpr.005QO".

The underlying representation of Excalibur's ontology is a set of frame-and-slot flat files. These are managed in a source-control system in parallel to source code, and are branched and merged in the usual ways. Ontologists do interact with these flat files, but also have a variety of other tools for viewing and editing the ontology. The ontology is also translated into other languages, such as RDF.

Verticals are topic areas (e.g. medicine, food, finance, photography, boxing, etc.) that may be used for the following purposes:

1) Concept grouping: each concept may be in zero or more verticals. This is deduced from ontology links and subsumption reasoning;

2) Document classification: documents may be in zero or more verticals, depending on the verticals of the concepts they contain; document membership is ranked.

3) Search interface specialization: the query system can be configured to prefer in-vertical interpretation of ambiguous expressions, and to prefer in-vertical result documents.

The mechanism of document search and retrieval described in (3) also incorporates facet- and drilldown-based search options.

Facets are document classifications, mostly applied by a pattern matching language based on concepts and word lists.

Drilldowns are groups of concepts, members of which might usefully be offered as narrower searches in some search interface.

Word and phrase lists are compiled that are useful for purposes like document classification, and detection of spam and adult content

2.2 VSW knowledge resource check and quality assurance toolbox

Editing the ontology

Ontologists modify knowledge resources in a number of ways: flat file editing, generating knowledge resource change files, and by using an online knowledge resource and management tool called SAGE.

Flat file editing

The most flexible and powerful way is simply to open the flat file representation in a text editor or to apply *ad hoc* editing scripts to those files. This is always available as a fallback mechanism, but is not the easiest way to perform day-to-day tasks. In particular, it is especially vulnerable to clashing with changes made by others. Source control merging is available and is, of course, invaluable, but is not semantic aware and cannot handle duplicate symbols and the like. The flat file format is frame-and-slot and does not directly represent additions and deletions.

Generating Knowledge Resource change files

For batch edits, and automatic symbol generation, there is a simple text format designed to represent additions and deletions of frames and slots. Such change files are often created by hand, but there are also several tools and scripts that can generate them. Because these change files represent an edit to the ontology and are less sensitive to edit conflicts, they are easier to polish over time and circulate for review.

SAGE

A web-based editing interface is also available. It can offer the frame-and-slot representation of some part of the ontology, such as some existing concept with all slots and inverse slots. The user can then add, remove, and modify slots, and even create new nodes (again with automatic symbol generation). Any difference between the set of assertions initially presented, and the set of assertions saved is taken to be the edit that the user intends. Such additions and deletions accumulate in a session that can be reviewed, abandoned, committed to source control, or passed on to another ontologist for review.

Underlying SAGE is a database reflecting the latest state of the ontology as submitted to source control, and a second database representing all the editing sessions and their respective modifications.

The history of any assertion or synset can be tracked over time, and proposed future changes can be detected. Simple reasoning can be performed against the ontology, either from the perspective of some point in the source control history, or through the lens of a specific session. An API is available for creating new sessions and populating them with changes from an external source.

This means that SAGE provides a framework for integrating human editing and review with any source of knowledge, and will be able to support such services as concept creation wizards, *ad hoc* intrinsic tests, and other ways to suggest changes automatically.

Browsing and visualizing the ontology

SAGE provides a way to browse the ontology, frame-by-frame, but a more powerful facility is embedded in every Query Front End (QFE). This additionally provides implementations of the special-purpose reasoning incorporated in the indexing and query pipelines, such as that for triggering verticals and facets, and ancestor latching. A key feature provided allows the user to enter a short document in a text field and immediately view how the text is processed by the tokenizer and stemmer, and all potential latches permitted by dictionary matching. Output may be formatted in HTML for human viewing, or in XML for use by external visualization tools.

The QFE provides multiple buttons to visualize an ontology node in various ways. These visualizations are provided using Graphviz [10], and are driven by data provided by the QFE's debugging XML API. All ontology diagrams shown in this paper were taken directly from this tool. These always reflect the state of the knowledge resources in use by some search or indexing cluster. The QFE provides a SAGE "Edit" button on every synset that allows immediate lightweight correction of any issue by any engineer.

Figure 1 below illustrates the representation of the polysemous word "Java" in the knowledge resources with its three senses mapped to respective ontological concepts. The graph displays lexical and ontological subsumption. Relatively few horitzontal links are shown here because the three focal concepts are not strongly related. The chains of subsumption BT ("broad term") and BI_noI ("uninheritable broad term") links lead to top-level collections "physical object" and "mental object", which are known to be disjoint (via the DJ link). BT_noI links point to top-level concepts that are excluded from inheritance, and

Figure 1: mid- and top-level ontological branching

the HOMY ("homonymy") links capture the three-sense polysemy of the word "Java".

Consistency checks in knowledge resources

As part of assuring quality and coverage in the knowledge resources, intrinsic consistency checks are routinely performed, either as basic validation or as reports (which are permitted to have extant violations). Below is a concise list of basic principles that inform internal consistency check in the ontology. Consistent implementation of the principles should prevent unwarranted cyclicity, reflexivity and redundancy in vertical and horizontal link chains and also ensure minimal depth of concept description, sufficient partition of concept classes, and enable domain and range link restrictions.

Knowledge resource quality principles:
- No broken links (links to undefined nodes) or duplicate nodes are allowed.
- Every concept must have at least one explicitly defined horizontal link
- Relations are irreflexive and anti-symmetric unless defined otherwise. Irreflexive relations defined as transitive are acyclic.
- Asserted links are not redundant with the inferential closure, such as subsumption links between a node and its grandparent.
- Collections defined as disjoint have no instances or sub-collections in common
- Every node has a description and a parent description available in every supported language so that it can be described to the user
- No pair of sibling concepts is ontologically (that is, ignoring the lexicon) indistinguishable.
- Links comply with any domain and range type constraints defined for the relation.
- Every indexable concept occurs at least once in the corpus.
- Every indexable concept falls into at least one vertical.
- Special identifiers (e.g. homepage URLs for known entities) are well-formed.

3 Using knowledge resources in information retrieval

The section will discuss the ontology-based architecture of document indexing, including disambiguation, and query analysis developed at Vertical Search Works.

3.1 Using ontology during indexing

During indexing, documents pass through a number of phases of processing: HTML parsing and boilerplate detection, language detection, tokenization and stemming, expression matching, disambiguation, dynamic entity extraction, and document classification.

Below is a brief outline of the indexing pipeline:

HTML parsing and boilerplate detection:
The HTML DOM (Document Object Model) is extracted, and the nodes are classified into core content and boilerplate. The text of the document is extracted from the core content.

Language detection:
Using character trigrams, the language of the document is established.

Tokenization and stemming:
The document text is converted into a stream of tokens. Each token may have variants depending on stemming, case-folding, diacritic-folding, and punctuation-folding. Token keywords and bigrams are assigned ranks.

Expression matching:
The token stream is compared to synset expressions, subject to any expression-specific matching restrictions. Candidate latches are identified.

Disambiguation
The disambiguation module serves to "latch" surface expressions to underlying ontological concepts and calculate reliable confidence and rank scores for each concept found in the document. Lexically ambiguous input may result in multiple concept candidates for a single input string. The latching, confidence and rank estimation rely on a window-based spreading activation network (a well-established procedure – for more details see [2] and [6]). The network is biased towards any vertical the document is known to be in *a priori*, the set of candidate latches from the entire document, and the set of candidate
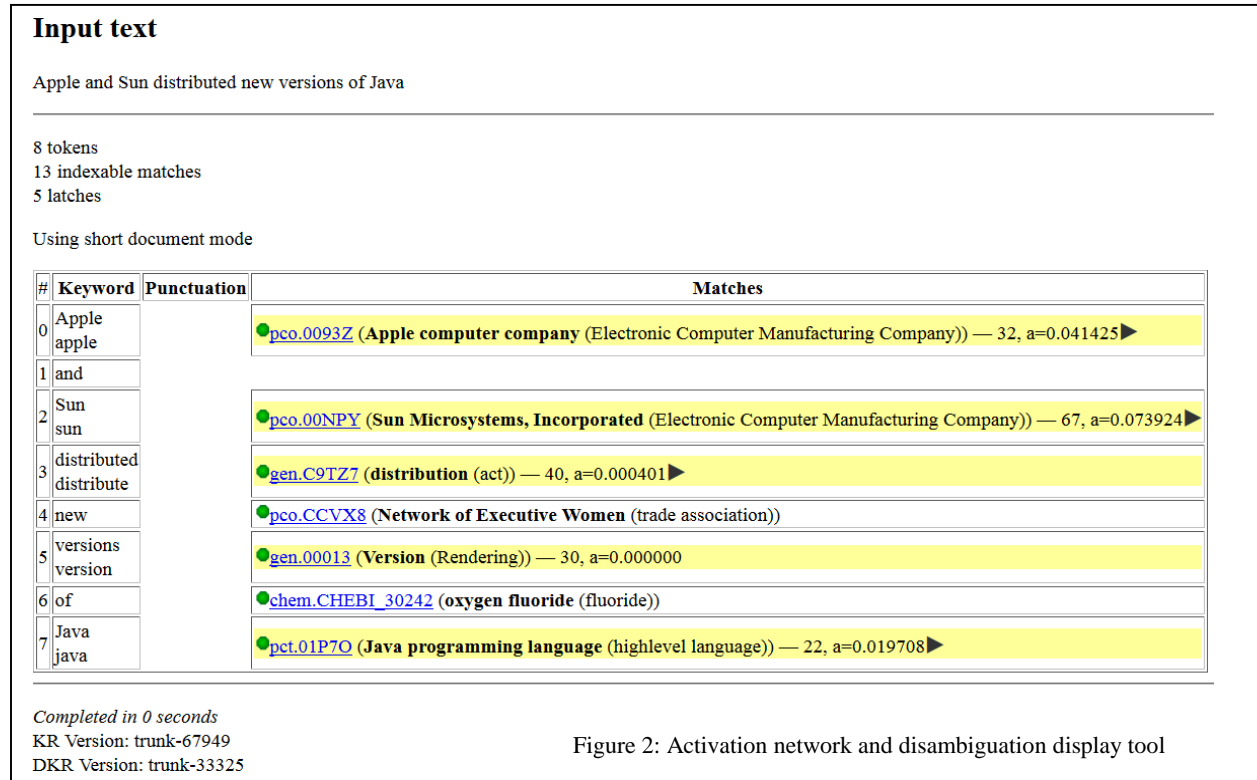
Input text

Apple and Sun distributed new versions of Java

8 tokens
13 indexable matches
5 latches

Using short document mode

#	Keyword	Punctuation	Matches
0	Apple apple		●pco.0093Z (**Apple computer company** (Electronic Computer Manufacturing Company)) — 32, a=0.041425 ▶
1	and		
2	Sun sun		●pco.00NPY (**Sun Microsystems, Incorporated** (Electronic Computer Manufacturing Company)) — 67, a=0.073924 ▶
3	distributed distribute		●gen.C9TZ7 (**distribution** (act)) — 40, a=0.000401 ▶
4	new		●pco.CCVX8 (**Network of Executive Women** (trade association))
5	versions version		●gen.00013 (**Version** (Rendering)) — 30, a=0.000000
6	of		●chem.CHEBI_30242 (**oxygen fluoride** (fluoride))
7	Java java		●pct.01P7O (**Java programming language** (highlevel language)) — 22, a=0.019708 ▶

Completed in 0 seconds
KR Version: trunk-67949
DKR Version: trunk-33325

Figure 2: Activation network and disambiguation display tool

latches in immediate proximity. A set of primary and secondary latches are selected and ranked.

Figure 2 above illustrates the "activation network" generated by propagating initial and progressively aggregated weights via ontological links. Resulting activation weights exceeding a threshold determine the "latching" to the selected concept, which is further indexed and also reinforces surrounding "latch" candidates. The input sentence is "Apple and Sun distributed new versions of Java", and the ambiguous proper nouns are successfully "latched" to appropriate concepts (highlighted) by gaining mutual support from proximate "latch" candidates. The activation link chains, including link types and directions, are available for viewing via collapsible widgets. For the tokens "new" and "of", two senses with case-sensitive expressions NEW for the synset "Network of Executive Women" and OF for the synset "oxygen fluoride" were initially considered as latch candidates but discarded due to the lack of activation support.

A key feature of this disambiguation process is that it is applied to all candidate latches, not just those that are known to be ambiguous in the knowledge resource. The need to be able to process a large amount of potentially unattested input (product, brand, titles, band names, etc.) during run time dictates a number of assumptions about the nature of data and the functionality of the text processing architecture. First, we assume that our KR will always be incomplete, because any otherwise unambiguous lexical item or expression may also appear as the title of a movie or the name of a band. Second, it is assumed that incomplete coverage in the knowledge resources should be supplemented dynamically, i.e. the system should have the functionality required to detect and resolve this situational ambiguity when no adequate expression could be found in the knowledge resources. As a result, during text processing the system is given considerable leeway in selecting potential latch candidates and may even choose to ignore an expression available in the knowledge resources if there are strong contextual considerations against it.

As a limited form of relevance blurring, ancestor synsets (collections and super-collections) of the selected latches are also added to the document as indirect latches. For example, if a document mentions "poodle" then we would recognize that the document was (at least weakly) about dogs, mammals, animals, *etc.*

Dynamic entity extraction

Dynamic entities such as telephone numbers, email addresses, and recipe preparation times are extracted automatically. The system not only detects a particular telephone number but also segments it into meaningful "roles", i.e. area codes, domain names, etc.

More sophisticated entity extraction rules, developed in an in-house language, allow for programmatic gathering of named entities and concept-concept relations based on ordered sequences of concept expressions and key lexical items. To illustrate, in sentences like "Richard Reid was a member of Al-Qaeda" and "Jerry Stackhouse played for the Pistons" the membership relation between the individuals and respective groups would be extracted automatically based on the same triple rule. Triple rules are a powerful means to extract information from unstructured natural language text: resulting triples can further be stored in an RDF repository for reasoning, search, lookup, classification and retrieval purposes.

Document classification

The Excalibur indexer also executes a large number of facet rules. These are defined in a similar way as entity extraction rules, but the goal here is to provide high-level categorizations about the document. An example would be a facet rule detecting news article, academic article, FAQ, clinical guidelines that describe tobacco use in a positive light. Facet rules are very powerful, and can match syntactic sequences, mere proximity of terms, or specific parts of a document.

3.2 Using ontology during query analysis

The indexing process described above extracts the concepts underlying the natural language text of document, but the typically user query is also expressed in natural language. In the simple case where the query refers to a single concept within the application vertical, then it can be interpreted directly using an ontology-based lexicon. For more complex cases, disambiguation is required.

VSW's patented approach ([7], [8]) is to use the query as a keyword search against a vertically-restricted corpus. The results of that search are then used to translate the query into synsets, or into a hybrid of synsets and residual keywords. The system examines the locations in the result documents where the query terms appear. These are scored for length, proximity, and corpus statistics. A semantic interpretation of the user query is then selected, possibly with alternative interpretations for interactive

clarification dialogue. This approach allows Excalibur to interpret multiple concepts in the user query in accordance with their relation to each other and, when appropriate, to fill in missing words. There are other layers in the query pipeline concerned with business logic and presentation, but these are beyond the scope of this paper.

4 Performance evaluation and validation

The evaluation cycle of the ontology parallels that of Excalibur's software development. It comprises five key elements: validation, black box end-to-end testing, glass box tests and desk-check QA, and reports.

Validation

This is a basic check of syntax and some simple inherent properties, such as acyclicity in transitive relations (see above). It is performed repeatedly throughout the cycle. Anyone editing the ontology will perform the test locally before committing changes to source control. In the same way that it checks that the source code will build, CruiseControl ensures that the ontology in source control passes validation. All components that use the ontology perform the same checks on every load, making it part of the indexing cycle.

Black box end-to-end

A key property of the ontology is that it performs well at its intended task within a software architecture. In order to test the search quality of the end-to-end system, Excalibur uses the TREC [9] approach of taking typical user queries and a set of documents that have been manually evaluated against those queries. These tests are not binary pass/fail, but rather measure the extent to which the system selects good documents as relevant and rejects bad documents, and to which it ranks good documents above bad ones.

Initially the GOV corpus was used, but this was subsequently supplanted by an internally-developed test corpus focused on typical user queries against typical web documents in the verticals of interest. Bad, good, and very good results are distinguished, and the votes of multiple evaluators are combined. The TREC measure "prefs_simp", ignoring unevaluated documents, was found to be most useful. Results can be broken down by vertical and examined at the level of queries and individual search results. The low level of data means that fine-grained results must be used

with caution. The absolute results are less important than relative changes seen as the code and ontology are modified day-to-day.

The test corpus used for this end-to-end testing is included in every index, from development to production. The tests are executed daily against all indexes and the results reviewed frequently.

Glass box semantic tests

As part of the general tests of the Excalibur platform, there are automated pass/fail tests that evaluate specific semantic applications of the ontology, in performing query expansion, synset latching, or document classification. These are performed as part of daily builds, and every deployment of a production index. These tests are developed either as regression tests for issues resolved, or as functional tests of enhancements.

Desk-check QA

Desk-check QA (manual verification of completed bug fixes) is routinely performed as part of knowledge resource development and maintenance work at VSW. Most ontology revisions relate to reported bugs or enhancement requests, and are therefore subject to the usual software defect tracking and change management processes. They are therefore reviewed and tested in the same way as the software. Automated glass-box tests may be developed at this point. Desk-check QA practice has proven useful in identifying broader classes of problems from specific instances. For example, identifying missing classes of child concepts after observing isolated (i.e. reported separately) instances of reifying siblings has been instrumental in ontological acquisition.

Reports and reviews

A final area of quality assurance in the KR is that when issues are detected, the team brainstorms ways to detect other instances of the same issue automatically. Typically such new intrinsic tests will expose many existing problems (and benign exceptions), so they cannot immediately be made part of the core intrinsic tests used for validation. Instead they are first presented as reports, and used to sweep the ontology clean over time.

5 Conclusion

This paper described the large-scale ontology used by VSW as part of a web-scale semantic search engine. Unstructured documents and user queries are both

assigned meaning using the lexical knowledge embedded in the ontology. Tools and process for knowledge acquisition, ontology maintenance, validation, and visualization were presented. In particular, certain principles were outlined for ensuring the consistency of the ontology and its usefulness for text processing. The process of disambiguation was discussed, with particular reference to relevant in-house visualization tools.

6 References

[1] Borgo, A., and C. Masolo. Foundational Choices in DOLCE: In: S. Staab, S Studer (eds.) Handbook on Ontologies. Springer, 2009

[2] Crestani, F. Application of Spreading Activation Techniques in Information Retrieval. *Artificial Intelligence Review* 11: 1997, 453-482.

[3] Lenat, D., and R. V. Guha. Building Large Scale Knowledge Systems. Reading, Addison-Wessley, 1990

[4] S. Nirenburg and V. Raskin, Ontological Semantics. Cambridge, MA: MIT Press, 2004.

[5] Pease, A., and C. Fellbaum. Formal Ontology as Interlingua: The SUMO and WordNet: Linking Project and GlobalWordNet, In: Huang, C. R. et al (eds.) Ontology and the Lexicon. Cambridge: Cambridge University Press, 2010

[6] Waltz, D. L., and J. B. Polack 1985. Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation. *Cognitive Science* 9:1, 1985, 51-74.

[7] Convera Corporation (2010). Search System and Method. US Patent us 7,668,825

[8] Convera Corporation (2010). SEARCH SYSTEM AND METHOD. US Patent us 7,783,668

[9] http://trec.nist.gov/

[10] http://www.graphviz.org/

Association-Based Identification of Internet Users Interests

M. Charnine,[1] **A. Petrov,**[2] **and I. Kuznetsov**[1]
[1]Institute for Informatics Problems, Russian Academy of Sciences, Moscow, Russia
[2]Tinkoff Digital, Moscow, Russia

Abstract – *The method to provide the Internet user with useful information, while using search engines is presented. Here we mean the systematization of search results according to user's interests, and also showing advertisement which could be interesting for the user. We introduce concept of "user profile", consisting of keywords/terms, reflecting user interests. The discovering of such keywords is done by parsing user queries and visited websites. The proposed method uses a tree of categories linked to related websites and to the advertising. From these websites we retrieve primary keywords characterizing categories. The primary keywords are extended with new associated ones (called secondary) which were obtained by the methods of distributive semantics. By comparing keywords of user's profile and keywords of the categories we determine relevant categories and useful information (including advertising). The use of distributional semantics methods allows us to obtain good results even on short search queries.*

Keywords: user behavior analysis, user interests; cognitive technologies; semantic analysis of Internet texts; text classification.

1 Introduction

One way of increasing the quality of Internet search engines and other Internet systems is the use of knowledge about the characteristics of users. Such characteristics may be gender, age, interests of users in the form of keywords, etc. Having more complete information about the user, the search engines can provide more "personalized" results, as follows:

- Show to user the content which he/she is interested in (articles, music, movies, books, etc.);
- Adjust the sequence of the search results for the user. For example, the search engine can show more interesting to the user web page in a higher position.
- Optimize the selection of advertising, showing to the user only those web pages and pictures that can be interesting to him/her.

Typically, data about the features and characteristics of a particular user are not explicit in the Internet. At the same time, the Internet has a large set of data about the user's actions while using the computer. By using such data it is possible to recover the missing information about users with sufficient accuracy. The data about the user actions includes:

- history of visits of the various Internet pages;
- user search history;
- history of clicks on hyperlinks and banners;
- history of purchases from online stores;
- video viewing history;
- history of listening audio files.

In this article, to determine the characteristics of a user we consider only two types of actions: the history of visits and the search history. We introduce a method that gives the possibility to automatically find the characteristics of users by linguistic analysis of search queries and visited web pages.

2 Representation of user interests

To solve the problem of automatic identification of user interests we chose an approach based on automatic analysis of the history of user queries and visited websites. We also use a training set of categorized webpages to build classifier that contains not only keywords/terms of the training set, but also associatively related terms. The analysis of the user history includes automatic classification of visited webpages and discovering/mining keywords/terms that characterize the current user.

The user's interests can be represented as a set of pairs <keyword, weight>. For example, a shopper can have the following interests: <dress, 70> <cosmetics, 40> <handbag, 20>, and the interests of the sports fan can be represented as follows: <sports, 30> <football, 60> <volleyball, 20>. Here the numbers are representing the weight of the keyword, which expresses a particular interest in a certain scale. Note that a specific user interest may also be represented as a point (or vector) in a multidimensional vector space of keywords/terms. This way a vector of user interests is formed. This representation allows us to introduce a measure of semantic similarity between interests of different users (proximity of points in keyword/term space). Also this representation helps us to automatically identify the groups of users with similar interests.

To classify the interests of users we use the category tree (taken from the Internet), where each category corresponds to a specific theme, expressed by the keywords or terms (football, indoor football, American football) and has links to the relevant explanatory texts (websites, articles). These texts are used as a training set for the process of automatic categorization and discovering keywords/terms that are related to each category. These category keywords/terms are represented as triples   <keyword, category, weight>, where weight is a number that describes the significance of the keyword for identification of the category. Let's call such keywords as primary. The number of such triples can be extended by including of keywords/terms that are associated with the name of category or with a set of primary keywords of the category. These associated keywords/terms are discovering from the variety of Internet texts where the name of the category, for example, football, is mentioned. Let's call such associated keywords/terms as secondary keywords. This way we get triples <secondary keyword, category, weight> where the weight describes the strength of the association.

In proposed method each category in the category tree is associated with relevant content or advertising. So it's very important to determine the appropriate categories for the user. This determination is done by comparison of the keywords of the user and the category. The categories, to which the user is belong/relevant, are chosen because of their semantic similarity. As a result, the user receives relevant content or advertising.

Note that if we know the vector of user interests, then it is possible to make an assumption about the gender and age of the person. For example, men are more interested in football, and women are more interested in cosmetics. Young people are interested in contemporary music and at the same time old people are interested in a retro. We call vectors of user interests, which are supplemented by information on the categories of user, the user's profile.

3   Building of the vector of user interests

The calculation of vector of user interests is done by automatic analysis of the history of user queries and visited websites. This user history determines the collection of user related texts including texts of queries and texts of visited web pages.
Vector of user interests is constructed by statistical analysis of user related texts and calculation of frequencies of the various keywords/terms that are mentioned in these texts. These data forms a multidimensional vector of user interests, the components of which are the frequencies of the various keywords. This vector also can be represented as a point in a multidimensional vector space of keywords/terms.

A set of primary and secondary keywords of each category and their weights also can be presented as a point (or vector) in this multidimensional vector space. This vector representing weights of primary and secondary keywords of category we called as vector of category features.

This representation allows us to introduce a measure of semantic similarity between vector of user interests and vector of category features as proximity of points in keyword/term space. So this representation helps us to automatically identify the most relevant categories for the user.

Vector of category features depends from the method of calculation of weights of primary and secondary keywords. So weights of primary and secondary keywords of category must be carefully calculated to provide the most accurate classification of text documents and users histories.

4   Calculation of the optimal weights

The task of automatic classification of visited documents (their relation to the categories), can be reduced to the question of collective decision of experts, each of them only responds to the document in the presence of one particular keyword/term associated with the category. The experts have to answer the question: is the document belongs to category or not? Thus, the number of experts is equal to the number of keywords and each expert is responsible for one specific keyword. If the keyword exists in the document, the expert makes decision about belonging of the document to the specified category. The probability of correct expert decision is equal to conditional probability:
$Pi = p(\text{Category} \mid \text{Keyword})$.

The main result of Nitzan and Paroush (1982) [16], and Shapley and Grofman (1984) [17] was the claim that if we have two alternatives and the probability of correct decision for each expert is known, then the linear combination rule of their decisions is optimal, and the maximum probability of correct collective decision is reached when the weights (shares) of experts calculated by the formula:

$Wi = \log(Pi/1\text{-}Pi)$,   (1)

Where, Pi – is the probability of correct solutions of expert with number "i". This formula (1) is optimal when we have two alternatives, for example, gender: man and women.

In the case when we have more than two alternatives we can use another formula (2) that is discussed below.

Let Keyword represent the problem keyword and {category-1, category-2, …, category-n} represent the alternatives. The PMI-IR algorithm assigns a weight/score to each category, weight(category-i), and selects the category that maximizes the weight/score.

The PMI-IR algorithm, is based on co-occurrence. The core idea is that "a word is characterized by the company it keeps" [21], that is another version of distributional hypothesis discussed later. There are many different measures of the degree to which two facts co-occur. PMI-IR uses Pointwise Mutual Information (PMI) [19, 20], as follows:

weight(keyword) = weight(category-i) =

Log(p(keyword&category-i)/(p(keyword)p(category-i)))   (2)

Here, p(keyword&category-i) is the probability of co-occurrence of two facts: keyword exist in the document and document belongs to category-i. If keyword and category-i are statistically independent, then the probability that they co-occur is given by the product p(keyword)p(category-i). If they are not independent, and they have a tendency to co-occur, then p(keyword&category-i) will be greater than p(keyword)p(category-i). Therefore the ratio between p(keyword&category-i) and p(keyword)p(category-i) is a measure of the degree of statistical dependence between keyword and category-i. The Log of this ratio is the amount of information that we acquire about the presence of keyword when we observe document of category-i. Since the equation is symmetrical, it is also the amount of information that we acquire about belonging the document to the category-i when we observe keyword, which explains the term mutual information.

These formulas (1) and (2) allow us to calculate weight of keyword associated with a given category if we know the probability of the presence of this keyword in the documents of the category. The method of calculation of probabilities is discussed in the next section.

5   Calculation of the probability of finding given keyword in the context of the category

Let's define the context of the category as a set of documents (or sentences) associated with this category. In the simplest case, the probability of presence of given keyword/term in the context of the category is calculated as the number of occurrences of this keyword in the context divided by the number of occurrences of keyword in all analyzed texts.

This method is similar to the wellknown measure TF-IDF [5], which takes into account not only the frequency of the presence of keyword/term in the documents associated with the category, but also the frequency of documents containing given keyword/term, which reduces weight of generally used and insignificant words such as interjections, prepositions, etc.

A more accurate calculation of the probability of occurrence of the keyword/term in the context of the category should consider the probability error that is discussed in section 9.

6   Calculation of the weights of primary keywords of the category

To calculate the weight of primary keyword of the category we should know the probability of presence of this keyword in the context of category. We use category tree with related documents to calculate the context of category.

As a training set for the creation of the initial version of the category tree we can use any online directory that contain links to Web sites, for example, Google, Yahoo, Yandex. In our research we use Yandex Catalogue as a reference catalogue of Russian sites. Yandex Catalogue consists of a set of tree-like categories; each of category contains a brief description, subcategories and links to Web pages. Thus one web page can be included into several different categories. In this case, if the web page belongs to the category and it also belongs to all parent categories.

For example, a category FOOTBALL (upper level - SPORT) consists of subcategories:

RUSSIAN FOOTBALL

WORLD FOOTBALL

MINI FOOTBALL. . .,

And links to sites:

FOOTBALL ON PORTAL "CHEMPIONAT.COM"

Russian Football Union. . .

Websites have supplementary texts and pictures which are used as a reference to the sites.

We use the program in python and library nltk to collect text information from web pages. Text of web pages has been cleared of html-markup and broken into strings of words by using standard library functions "nltk". We divided such strings into various keywords. For each keyword we have counted the number of occurrences in the document. As a result, for each web page from the referenced file directory has been created a file, which consists of triplets
        <keyword, category, the number of occurrences>.

On the next step with the data processing program map-reduce [6] for each pair <keyword, category> was counted the total number of occurrences of the keywords in each of the parent categories. The resulting file was used to assign keywords to a category.

These data were used for calculation of the probabilities of occurrence of keywords in the documents of particular category. By using these probabilities and formulas (1) and (2) we calculated the weight of primary keyword of the category.

7 Calculation of the weights of secondary keywords of the category

The weight of a secondary keyword, which is associated with the name of the category, is calculated using the probability of occurrence of the keyword in the context of the category. Context is the set of all documents (or phrases) containing the name of the category and some of its primary keywords.

Discovering of the association from Internet texts is based on the distributional hypothesis, which states that semantically similar (or related) keywords/terms have a similar context, and, conversely, keywords/terms with similar context are semantically close.

The discovery of secondary keywords and their weights involves the use of various methods, including:

-    methods of detection of Internet texts of specific subject areas;

-    methods of detection of significant texts keywords, terms, and their ranking;

-    methods of identifying and ranking the associative relationships between important keywords and terms.

The processing of large volumes of texts that are constantly updated in the Internet allows us to collect all the necessary statistical data to generate a fairly complete picture of the subject area that can be represented as a set of associative relationships. The ability to use machine learning techniques on a large number of examples gives the system flexibility and improves the results.

8 Aligning of weights of primary and secondary keywords

The weight of the keyword that is associated with the category may be calculated by different methods, for example, based on the probability of occurrence of the keyword in the documents of category in training set. The next possible method for calculation of the weight is based on the probability of finding keyword in the Internet context of the category. Saying context here we mean all the Internet documents (or phrases) that contains category name and some of the primary keywords of the category.

The weights of the same keyword calculated by the different methods should be similar as much as possible. To achieve that, we proposed the following method of weights aligning.

Suppose, by using two different methods, we have found two groups of keywords T1 and T2 that are associated with a particular category. For example, a set T1 consists of primary keywords that are included in the training set of the category, and a set T2 consists of secondary keywords associated with

the category and discovered from Internet context of the category. The Internet context of the category usually is much bigger than training set, so the set T2 is usually bigger than T1. The weights for T1 and T2 are calculated independently by using two different algorithms. Usually the set T1 is included into the set T2, so the weights for T1 are calculated by both algorithms, and we can select the weight coefficient for T2 to align them with the weights of T1. We use this coefficient for calculating all the weights of secondary keywords.

9 Determining the user's interests using history of his visits and inquiries

Previously, it was discussed the method of calculation of the category of each of the visited by user web pages. This method can serve as the basis for construction a user profile that will reflect his interests. Such profile can be represented as a plurality of pairs   <category, weight>. Here the weight can be calculated, for example, as the number of the web pages in the category «C» divided to the total number of web pages, visited by the user.

Suppose that the user's interests are unchanged. Then we can assume that the number of web pages of the category «C» in comparison with the total number of visited web pages obeys the binomial distribution. Let's take «N» as the total number of pages that were visited by the user, and «n» - the number of pages category «C». In this case, the weight will be calculated according to the formula: $W = n/N$. Then confidence interval (with 95% confidence level) may be calculated by the formulas:

$Wmin = W - 1.96 * SQRT(W*(1-W)/N)$.

$Wmax = W + 1.96 * SQRT(G2*(1-G2)/D2)$

For example, if $n = 2$ and $N = 4$, then in this case $Wmin = 0.01$, and $Wmax = 0.99$, then there we can say nothing specific about the value of W. If $n = 100$ and $N = 200$, $Wmin = 0.43$, and $Wmax = 0.56$ - in this case, it is safe to say that the category is found and that it is not a "noise" (i.e., web pages that were visited by accident). In order to exclude the "noise" from the user's interests we included into the user profile only those keywords which have the minimum value of the confidence interval greater than a certain threshold.

10 Conclusions

This paper describes the method of determination of user's interests from the history of visited websites and user search history. We have developed a program that can successfully determine the interests of the user even from the short texts that do not contain keywords/terms from training set for calculation of the classifier. Note that standard algorithms usually didn't work well with such texts. In the future we plan

to develop an algorithm that will allow us to get the user profile, which takes into account a number of factors, including the user gender, age, interests, intentions, region of residence, income level, marital status and other useful information.

11 Acknowledgements

12 References

[1] Greg Linden, Brent Smith, Jeremy York "Amazon.com recommendations. Item-to-item collaborative filtering". EEE Internet Computing, Los Alamitos, CA USA, 2003 http://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf.

[2] Brian McFee, Luke Barrington, Gert Lanckriet "Learning Similarity from Collaborative Filters", ISMIR 2010, http://cosmal.ucsd.edu/cal/pubs/ISMIR2010_learnCF.pdf.

[3] M.C. Agueyev, "Methods of automatic text categorization based on machine learning and expert knowledge", PhD Thesis: 05.13.11, Moscow, 2004.

[4] V.E. Abramov, "Automatic classification and abstracting of textual information: including in foreign languages," PhD Thesis: 05.25.05, Moscow, 2008.

[5] Salton, G. and McGill, M. J. 1983 Introduction to modern information retrieval. McGraw-Hill, ISBN 0-07-054484-0.

[6] J Dean, S Ghemawat, «MapReduce: simplified data processing on large clusters», Communications of the ACM, 2008 - dl.acm.org.

[7] E.Baharad, J.Goldberger, M.Koppel и S.Nitzan, "Beyond Condorcet: Optimal Aggregation Rules Using Voting Records", CESifo München, 2011.

[8] A. Lenci, "Distributional semantics in linguistic and cognitive research", Rivista di Linguistica, 1, 2008, pp.1-30.

[9] M.Baroni, A.Lenci, "Distributional Memory: A General Framework for Corpus-Based Semantics", Computational Linguistics. V.36, Issue 4, 2010, pp. 673-721.

[10] Peter Turney, "A uniform approach to analogies, synonyms, antonyms and associations", Proceedings of COLING, Manchester, 2008, pp. 905–912.

[11] Peter Turney, "Mining the web for synonyms: PMI-IR versus LSA on TOEFL", Proceedings of the Twelfth European Conference on Machine Learning (ECML-2001), Freiburg, Germany. September 3–7, 2001. pp. 491–502.

[12] M.Charnine, I.P.Kuznetsov, E.B.Kozerenko, "Semantic Navigator for Internet Search", Proceeding of International Conference on Machine Learning, 27-30, 2005 Las Vegas, USA, CSREA Press, pp. 60-65, 2005.

[13] Michael Charnine, Vladimir Charnine. Keywen Category Structure.// Wordclay, USA, 2008, pp.1-60.

[14] Michael Charnine, "Keywen Automated Writing Tools", Booktango, USA, 2013, ISBN 978-1-46892-205-9.

[15] Nitzan, S., and J. Paroush. 1985. "Collective Decision Making: An Economic Outlook", Cambridge University Press, Cambridge, England.

[16] Nitzan, S., and J. Paroush. 1982. "Optimal Decision Rules in Uncertain Dichotomous Choice Situations." International Economic Review 23(2): 289-97.

[17] Shapley, L. and B. Grofman. 1984. "Optimizing Group Judgmental Accuracy in the Presence of Interdependencies." Public Choice 43: 329-343.

[18] R. Diamond, "Designing and improving courses and curricula in higher education: A systematic approach: Jossey-Bass," SF,1989.

[19] K.W. Church, P. Hanks, "Word association norms, mutual information and lexicography", Proceedings of the 27th Annual Conference of the Association of Computational Linguistics, 1989, pp. 76-83.

[20] K.W. Church, W. Gale, P. Hanks, D. Hindle, "Using statistics in lexical analysis", In: Uri Zernik (ed.), "Lexical acquisition: Exploiting on-line resources to build a lexicon", New Jersey, Lawrence Erlbaum, 1991, 115-164.

[21] Firth, J.R.: "A synopsis of linguistic theory 1930-1955", In Studies in Linguistic Analysis, pp.1-32, Oxford, Philological Society (1957).

Automatic Processing and Acoustic Normalization of Vowels

Lai, Yi-hsiu[*]

Department of Western Languages & Literature, National University of Kaohsiung

Abstract - *This paper addressed the issue of automatic processing of phonetic cues and acoustic normalization of Hakka vowel production in Mainland China and in Taiwan. An acoustic study of Hakka vowels, produced by native speakers of Hakka, was conducted. Fifty participants (twenty from Mainland China and thirty from Taiwan) were invited to read a wordlist of Hakka with six vowels embedded in different contexts (i.e. in different tones, in different phrase positions). An automatic approach was then adopted to analyze the collected acoustic data. Acoustic cues of F0, F1, F2, F3 and F4 were automatically fetched, and these cues were further examined in the normalization procedures. Based on the discussion on the normalization of Hakka vowels, implications for relevant linguistic issues and applications of machine learning to speech and language processing were provided.*

Keywords: Automatic processing, acoustic properties, vowel normalization, Taiwan Hakka

1   Introduction

Recently, acoustic reports of vowels have been increasingly made. Acoustic cues are often fetched manually to examine formant averages, vowel distribution or vowel plotting. To minimize the danger of being unreliable or inconsistent in the manual operation, an automatic approach, also one kind of intelligent linguistic technologies, which follows certain formula and criteria, is suggested in the researcher's previous report [1]. This paper follows Lai [1] to examine automatic processing of vowel acoustic properties, which may set a model of spectral analysis for future studies.

Additionally, due to personal variables of vowel production, it remains controversial and limited if the discussion on vowels is made merely on the formant averages. To reduce the interference of personal variables, vowel normalization has been conducted and has attracted greater attention in the western reports [2, 3, 4, 5]. To lower the variables of individual voice quality, over the past decades, the vowel normalization -- through the comparison of acoustic data of speakers from different regions -- has increasingly received the attention of western phoneticians and sociolinguists. The so-called phonetic normalization is of two types, the acoustic normalization or the perceptual normalization [6]; the former refers to the transformation of raw acoustic data into the normalized values so as to reduce the influences of raw acoustic properties and the latter refers to the transformation of hearers' phonetic perception data to exclude the individual and relevant variables in anticipation of sketching the characteristics of phonetic perception. Many western reports have pointed out the importance of acoustic normalization [3, 4, 5]. The advantages of normalization are, firstly, to minimize the possible influences of speakers' physiological differences, secondly, to retain the speakers' sociolinguistic information, and thirdly, to retrieve the phonemic information.

Over the past fifty years, there have been fruitful results in the studies of phonetic description, phonology, morphology and syntactic structures in Taiwan Hakka [7, 8, 9, 10]. In the early twentieth century, there was a lack of Hakka acoustic research [11, 12, 13]. While in the recent years, the acoustic research of Hakka segments receives more attention than it used to be. Nevertheless, most studies of Hakka vowels are merely limited to the presentation of formant averages, to the distribution of vowels or to the description of vowel patterns [14, 15, 16]. To date, none of the acoustic studies on Hakka vowel normalization has been conducted, not to mention on such relevant linguistic issues as the sociolinguistic issues (i.e. Hakka vowel qualities of different areas or of different accents), the physiological issues (i.e. Hakka vowel qualities of different genders), and the linguistic issues (i.e. Hakka vowel qualities in different phonetic contexts).

To fill these academic gaps, this paper addressed the issue of automatic processing of phonetic cues and acoustic normalization of Hakka vowel production in Mainland China and in Taiwan. An acoustic study of Hakka vowels uttered by native speakers of Hakka was conducted. Fifty participants (twenty from Mainland China and thirty from Taiwan) were invited to read a wordlist of Hakka with six vowels [i, e, a, o, u, ɨ] embedded in different contexts (i.e. in different tones and in different phrase positions). An automatic approach was then adopted to analyze the collected acoustic data. Previous acoustic reports were primarily made manually and might pose greater degrees of variance in fetching the acoustic values. The current automatic processing aimed to demonstrate a way of conducting the acoustic analysis with greater consistency, but smaller deviance. Acoustic cues of F0, F1, F2, F3 and F4 were automatically retrieved with the help of computer script programs in *PRAAT* [17], and these cues were further examined in the normalization procedures [18, 19, 20, 21, 22].

Possible variables, including the sociolinguistic issues (i.e. Hakka vowel qualities of different areas or of different accents), the physiological issues (i.e. Hakka vowel qualities of different genders), and the linguistic issues (i.e. Hakka vowel qualities in different phonetic contexts), are discussed in the present study. On the basis of discussion on the normalization of Hakka vowels, implications for relevant linguistic issues and applications of machine learning to speech and language processing are elaborated upon.

2   Acoustic Normalization of Vowels

Individual differences frequently exist in the production of vowels. To lower the variable of individual characteristics, over the past decades, vowel normalization and statistical comparison of acoustic values of speakers from different regions has increasingly received the attention of phoneticians and sociolinguists. Acoustic normalization indicates that the normalized values will be less influenced by the raw acoustic qualities after the transformation of normalization procedures. A number of scholars [3, 4, 5] have pointed out three main goals of vowel normalization: (a) to reduce speakers' physiological and structural differences, (b) to retain the distinctive sociolinguistic information of speakers from different regions, and (c) to obtain the phonemic information. In the past decades, disputes concerning the classification of vowel acoustic normalization remain. Classification centers either on the intrinsic qualities of vowels/formants, or on the extrinsic qualities of vowels/formants. The most discussed one is the intrinsic as well as extrinsic qualities of vowels [23, 24]. The normalization of vowel intrinsic qualities depends primarily on a single vowel acoustic cue as a norm in the procedures, and it often includes the non-linear conversion of frequency values, such as log, mel, bark, or the conversion of several frequency waves, such as F1-F10.The normalization procedures of vowel intrinsic qualities are specified in Syrdal and Gopal [22]. However, the normalization procedures of vowel extrinsic qualities refer to more than one vowel acoustic characteristics of speakers. The normalization procedures of vowel extrinsic qualities can be referenced to the studies of Gerstman [18], Lobanov [19], Nearey [20] and Nordström [21].

In recent years, a number of researchers in the western countries examined similarity and differences of accents from different regions by means of vowel acoustic normalization procedures. Adank, Smits and van Hout [2] analyzed the vowel acoustics of two communities of Dutch native speakers through the normalization procedures. The subjects, consisting of eighty women and eighty men, were required to read out a word list containing nine Dutch vowels in the context of /sVs/. The vowel normalization procedures were divided into two categories: intrinsic and extrinsic qualities of vowels, both of which involved six normalized procedures. The results indicated that the most successful three indicators were LOBANOV, NEAREY1, GERSTMAN (Table 2), all of which were subsumed under the extrinsic normalization procedure. Recently, Flynn and Foulkes [3] collected the

vowel sounds from twenty native speakers of British English. They retrieved the acoustic values of F1 and F2, and conducted twenty normalization procedures, including six vowel intrinsic qualities and fourteen vowel extrinsic qualities. The results showed that the normalization procedures of vowel extrinsic qualites were better than those of intrinsic ones. This finding further supported that of Adank, Smits and van Hout [2].

Up to the present time, literature on Chinese vowel normalization is rarely found, let alone the studies of vowels of Hakka, one dialect in the Chinese language family. Huang and Tseng [25], for example, examined vowel formant frequency of twenty Mandarin-Min bilinguals in Taiwan in two phonetic contexts: the Mandarin context /lV/ and the Min context /kV/, using the Bark-distance measures, F1-F0, F2-F1, F3-F2, F4-F2, F4-F3. The results indicated that the formant frequency was higher in women than in men. It was found that F1 value of high vowels was lower than that of low vowels and that F2 value of front vowels was higher than that of back vowels. Furthermore, two Bark-distance measures, F1-F0 and F3-F2, echoed the high and low positions of vowels and the place of articulation. The distance F1-F0 of high vowels and the distance F3-F2 of front vowels were within the range of 3.5Bark. The findings confirmed that the vowel normalization procedures significantly reduced the variability of speakers of different genders. Huang and Tseng [25] was, however, limited to the Bark-distance measures without the involvement of other normalization procedures－which left the room for future research.

3   Method

In this section, the design of the current research is introduced. Description of participants, materials and data analysis is explicitly presented.

3.1   Participants

Fifty native speakers of Hakka (twenty from Mei-Xian, Mainland China and thirty from Taiwan) participated in the current study. Due to the geographical reasons, speakers from Taiwan were further divided into two subgroups: fourteen participants in Miao-Li (in the northern part of Taiwan) and sixteen participants in Mei-Nong (in the southern part of Taiwan). There were twenty-four male participants and twenty-six female ones. Their ages approximately ranged from thirty to fifty-five with the average of forty-three years old.

Rationales for the selection of participants in three different areas (i.e. Mei-Xian in Mainland China, Miao-Li, Mei-Nong in Taiwan) are three-fold. First of all, it was historically recorded that native speakers in these three areas spoke the same dialect of Hakka, that is, Si-Xian Hakka [26]. Mei-Xian Hakka in Mainland China was regarded as the source region, where Miao-Li Hakka and Mei-Nong Hakka in Taiwan were originated. It is worthy of investigation whether language migration of Hakka from Mainland China to Taiwan

results in significant changes in their accents. Secondly, it is worthwhile to examine whether Hakka in Taiwan is influenced by the local languages in Taiwan, leading to language integration or language mixing, which may make the accent of Hakka in Taiwan different from the source accent of Hakka in Mainland China. Thirdly, Si-Xian Hakka in Taiwan is spoken in two regions due to some geographical reasons. It is interesting to address whether language separation makes Hakka people in these two areas of significantly different accents.

3.2   Materials

Participants in the present study were invited to read a wordlist of Hakka with six vowels [i, e, a, o, u, ɨ] embedded in six tones: Tone 33, Tone 11, Tone 31, Tone 53, Tone 3 and Tone 5. And, the target vowels were then put in two different positions in two-word phrases: the first word (e.g. 西瓜) and the second word (e.g. 東西). In total, there were seventy-two target tokens (6 vowels*6 tones*2 positions) in the wordlist.

Participants completed the production task individually in a quiet room free from ambient noise. After a brief introduction of the goal and procedure of recording, they were invited to read aloud the wordlist, with a mini-microphone pinned on their collar, for three times in succession with normal speech rate.[1] The researcher monitored the recording session to avoid weak or overloaded signals. The recording process for one participant lasted approximately fifteen minutes.

3.3   Data Analysis

3.3.1   Acoustic Analysis of Vowel Production

To analyze the production data, a notebook computer and a *PRAAT* program [17] were utilized. The collected speech data were converted into wave files in *PRAAT*. The sound recorder of *PRAAT* was set with the 22050Hz sampling rate and 16-bit resolution of quantization.

The current study followed Lai [1] to pave an automatic way for vowel research in the field of experimental phonetics, which used to be a laborious and time-consuming job. The acoustic values of participants' production, as reflected in formant frequencies, were fetched with efficient and precise computer processing. Both rationales and procedures for this automatic computation are elaborated below.

3.3.2   Rationales for Automatic Processing

An automatic approach to analyze the acoustic data in the current study was adopted for three main reasons. In the first place, automatic processing of the acoustic data is more time-saving than manual processing. Since fifty subjects and a long inventory of speech tokens were involved in the present research, it would be a time-consuming job to manually

---

[1] In the current vowel production task, participants were asked to produce the wordlist for three times. But, only the middle token was analyzed acoustically in the present investigation.

operate the enormous amount of spectral data. Secondly, reliable and accurate results are best ensured in the automatic approach, which follows certain formula and criteria. By contrast, there is always a danger of being unreliable or inconsistent in the manual operation, even if the researchers have received a good training on the acoustic analysis of spectrograms. Thirdly, the automatic approach may set a model of spectral analysis for future studies. Previous acoustic reports were primarily conducted manually and might pose greater degrees of variance in fetching the acoustic values. The current automatic approach aimed to demonstrate a way of conducting the acoustic analysis with greater consistency, but smaller deviance.

In the automatic approach, three *PRAAT* script programs, "fetch_pitchlist.praat," "fetch_formant_list.praat" and "fetch_formant_mean_max_min.praat" were written by Huang [27] to obtain the speech data – pitch (F0) and formant (F1-F4) -- from the speech files of each participant. Additionally, an *ACCESS* program was adopted to make database files, to handle the tables of speech stimuli, and to store the acoustic data gathered by the *PRAAT* scripts. With the help of the *PRAAT* scripts and the *ACCESS* program, the program-controlled database management was made possible and this further laid the cornerstone of the automatic computation in the present research.

3.3.3   Automatic Computation Procedures

Before running the computation procedures, each vowel production by participants was saved in an organized way. Totally, there were 3600 speech files produced by fifty participants (72*50).

Table 1 *Eleven computation procedures*

Formant analysis	
1	Check Subject & Stimuli
2	Run *PRAAT* script: fetch_pitch_list. praat
3	Transfer Pitch List
4	Edit pitchList tables if necessary (to wipe out invalid pitch sections)
5	Get Pitch Duration & Percentage Time
6	Run *PRAAT* script: fetch_formant_list. praat
7	Transfer Formant List
8	Edit formantList tables if necessary (to wipe out no-value starting/ending formants)
9	Get Formant Duration & Percentage Time
10	Run *PRAAT* script: fetch_formant_mean_max_min_list. praat
11	Transfer Formant Mean & Max & Min

Following the construction of the database of wave files, the researcher proceeded to the formal computation procedures, inclusive of eleven individual steps (as indicated in Table 1). The first step "Check Subject & Stimuli" was to specify the project directory of each design. This text offered a stimuli list and a subject list, on which the subsequent formula would be based.

Running *PRAAT* script: "fetch_pitch_list.praat" was the second step. The third step "Transfer Pitch List" transferred the voiced pitch in the files of "pitch_list" into the database

and filtered out the undefined pitch. Two kinds of tables would be made. One was "available_pitchlist_tables," which kept record of those data about pitch values. The spectrogram of valid pitch, as painted with blue pulse and pitch line. The other was "empty_pitch_tables," which recorded those spectrograms without pitch values. These data in the "empty_pitch_tables" were either considered invalid (e.g. no pitch line in Figure 1) or worthy of manual operation to fetch the pitch values (e.g. two non-continuous pitch lines as circled in Figure 2).
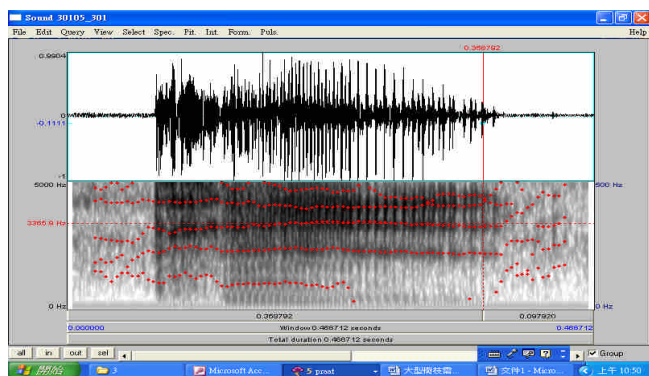
Figure 1 Spectrogram of invalid pitch for /t^{h}i/^{4}
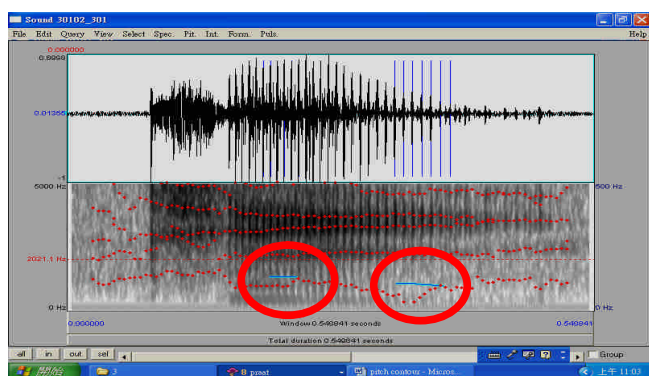
Figure 2 Spectrogram of non-continuous pitch lines for /t^{h}i/^{4}

"Edit pitchList tables if necessary (to wipe out invalid pitch sections)" was the fourth step. It aimed to examine whether the pitch contour was continuous or not. If there would be any pitch value more than 500Hz in the screen of pitch-list tables, it might result from the voiced consonants in the vowel-consonant transition. Then, the wave file in the *PRAAT* program should be double checked to examine whether such a discontinuity would influence the later fetch of vowel formants and whether this frame in the "pitch_lists" should be deleted.

The fifth step "Get Pitch Duration & Percentage time" helped to summarize the data of pitch values in the "available_pitchList_tables." The starting time as well as the ending time for each pitch would be respectively listed. And, the duration of each pitch would be recorded in the table of pitch_duration in the *ACCESS* program. The sixth and seventh

steps were to run "*PRAAT* script: fetch_formant_list.praat" in the *PRAAT* program and then to "Transfer Formant List." On the press of the button "Transfer Formant List," the files of formant lists would be created by computer itself and saved in the directory of "formant_list" in each stimulus.

"Edit formantList tables if necessary (to wipe out no-value starting/ending formants)" was the eighth step. If there would be any F1 value more than 1000Hz in the screen of pitch-list tables, it might be a result of voiced consonants in the vowel-consonant transition. Then, the wave file in the *PRAAT* program should be double checked and the frame with the extraordinary high formants in the "formant_list" should be deleted. Also, no-value starting/ending formants should be wiped out in this step. The ninth step was to "Get Formant Duration & Percentage Time." The tenth step was to run "*PRAAT* script: fetch_formant_mean_max_min.praat."

The last step was to "Transfer Formant Mean & Max & Min." The files of formant_list would then be surveyed, the maximum and minimum values would be fetched, and the mean formants would be calculated.

3.3.4 Acoustic Normalization of Vowels

Acoustic normalization of vowels followed the acoustic measurements of vowel production. *EXCELL* and *SPSS* software[2] were employed to compute the acoustic values (i.e. F0, F1, F2, F3, F4) in the normalizatoin procedures.

Table 2 *Normalization procedures in the current study*

Vowel-intrinsic procedures	
HZ	formant frequencies in Hz
LOG	log-transformation of the frequency scale
BARK	bark-transformation of the frequency scale
MEL	mel-transformation of the frequency scale
Vowel-extrinsic procedures	
LOBANOV	Lobanov's (1971) z-score transformation
NEAREY1	Nearey's (1978) single logmean procedure
NEAREY2	Nearey's (1978) shared logmean procedure
GERSTMAN	Gerstman's (1968) range normalization

Eight normalization procedures (Table 2), inclusive of four vowel-intrinsic ones and four vowel-extrinsic ones, were implemented in the present analysis. Vowel-intrinsic procedures included HZ, LOG, BARK and MEL, while vowel-extrinsic procedures referred to LOBANOV, NEAREY1, NEAREY2 and GERSTMAN. Rationales for these eight normalization procedures were two-fold. First of all, these procedures have been widely cited and discussed in the western research [3, 4, 5]. Secondly, these procedures included the significant indicators of Dutch vowel normalization, as found in Adank, Smits and van Hout [2]. It is hoped that these procedures may help significantly differentiate vowel production made by speakers of different accents, from different areas and of different genders and that the findings of these normalizatioin procedures in the present analysis may further contribute to a number of relevant

---

[2] A significant level 0.05 (*p*=0.05) was set in the current statistical analysis, for example, MANOVA test.

linguistic issues.

3.3.5   Research Questions

In the present study, an automatic approach was first adopted to fetch the acoustic cues of vowels. Then, acoustic normalization of vowels was implemented to address the following four questions:

(1)   To what extent did the acoustic normalization of vowels help differ vowel qualities of Taiwan Hakka from those of Mainland (Mei-Xian) Hakka, where Taiwan Hakka was originated?

(2)   To what extent did the acoustic normalization of vowels help differ vowel qualities of Miao-Li Hakka from those of Mei-Nong Hakka, both of which were spoken in Taiwan, but in different areas?

(3)   To what extent did the acoustic normalization of vowels help differ vowel qualities of male Hakka speakers from those of female Hakka speakers?

(4)   To what extent did the acoustic normalization of vowels help differ different phonetic variables (i.e. vowel types, tone types, positions)?

4   Results & Discussion

This section presents the results of acoustic normalization of vowels. Discussion was made concerning vowel-intrinsic procedures and vowel-extrinsic procedures. And, one normalization procedure is exemplified to illustrate the detailed statistics and the interaction between vowel patterns and relevant linguistic variables.

4.1.1   Acoustic   Normalization   of   Vowel-intrinsic Procedures

Vowel-intrinsic procedures in the current investigation referred to HZ, LOG, BARK and MEL. MANOVA tests were conducted to examine whether these procedures might help significantly differentiate (a) vowel qualities of different accents (i.e. Mainland Hakka vs. Taiwan Hakka), (b) vowel qualities of different areas (i.e. Miao-Li Hakka vs. Mei-Nong Hakka in Taiwan), (c) vowel qualities of different genders (i.e. male vs. female), and (d) different phonetic variables (i.e. vowel types, tone types, positions). Findings of the MANOVA tests are summarized in Table 3.

According to Table 3, significant results were found in all of the vowel-intrinsic procedures, including HZ, LOG, BARK and MEL. It can be argued that these vowel-intrinsic procedures help significantly differ vowel qualities of different accents (i.e. Mainland Hakka vs. Taiwan Hakka), vowel qualities of different areas (i.e. Miao-Li Hakka vs. Mei-Nong Hakka in Taiwan), vowel qualities of different genders (i.e. male vs. female), and different phonetic variables (i.e. vowel types, tone types, positions). To be brief, these vowel-intrinsic procedures were successful ass well as significant indicators of different linguistic variables.

Table 3 *Statistical results of vowel-intrinsic procedures*

	HZ	LOG	BARK	MEL
Accent	***	***	***	***
Area	***	***	***	***
Gender	***	***	***	***
Vowel	***	***	***	***
Tone	***	***	***	***
Position	***	***	***	***

Note: ***$p < .001$

4.1.2   Acoustic   Normalization   of   Vowel-extrinsic Procedures

Vowel-extrinsic procedures in the present analysis included LOBANOV, NEAREY1, NEAREY2 and GERSTMAN. MANOVA tests were implemented to investigate whether these procedures helped significantly differ (a) vowel qualities of different accents (i.e. Mainland Hakka vs. Taiwan Hakka), (b) vowel qualities of different areas (i.e. Miao-Li Hakka vs. Mei-Nong Hakka in Taiwan), (c) vowel qualities of different genders (i.e. male vs. female), and (d) different phonetic variables (i.e. vowel types, tone types, positions). Table 4 summarizes the findings of the MANOVA tests.

Table 4 *Statistical results of vowel-extrinsic procedures*

	LOBANOV	NEAREY1	NEAREY2	GERSTMAN
Accent	ns	ns	***	***
Area	ns	ns	***	***
Gender	ns	ns	***	***
Vowel	***	***	***	***
Tone	***	***	***	***
Position	***	***	***	***

Note: ***$p < .001$; ns: non-significant

Observation of Table 4 revealed that two of the vowel-extrinsic procedures (i.e. NEAREY2 and GERSTMAN) significantly differed vowel qualities of Taiwan Hakka from those of Mainland (Mei-Xian) Hakka, where Taiwan Hakka was originated. And, they succeeded in distinguishing vowel qualities of Miao-Li Hakka from those of Mei-Nong Hakka, both of which were spoken in Taiwan, but in different areas. Further, vowel qualities of different genders (i.e. male vs. female) and of different phonetic variables (i.e. vowel types, tone types, positions) can also be significantly detected via these two procedures.

As for the other two procedures, LOBANOV and NEAREY1 significantly differed vowels of different phonetic variables (i.e. vowel types, tone types, positions). But, they failed in the differentiation of vowel qualities of different accents (i.e. Mainland Hakka vs. Taiwan Hakka), vowel qualities of different areas (i.e. Miao-Li Hakka vs. Mei-Nong Hakka in Taiwan), vowel qualities of different genders (i.e. male vs. female). It could be inferred that LOBANOV and NEAREY1 served as two indicators less successful than NEAREY2 and GERSTMAN among the vowel-extrinsic procedures.

4.1.3   Vowel Patterns in One Normalization Procedure

This subsection presents one normalization procedure to

illustrate the detailed statistics and the interaction between vowel patterns and relevant linguistic variables. The statistics is taken from a baseline condition, that is, formant frequencies in Hz.

Table 5 reports formant frequencies made by participants of different accents (i.e. Mainland Hakka vs. Taiwan Hakka). The statistical results were indicative of significant differences in four formant frequencies (i.e. F0, F1, F3, F4), except for F2. Judging from the means, speakers of Mainland Hakka produced Hakka vowels with significantly higher values in F0, F1, F3 and F4 than those of Taiwan Hakka. The mean value of F2 was also higher for speakers of Mainland Hakka than for those of Taiwan Hakka, though not to a significant level.

Table 5 *Formant frequencies made by participants of different accents*

	Accent	N	Mean	SD	F
F0	Mainland Hakka	20	190.24	70.27	98.33***
	Taiwan Hakka	30	168.08	61.56	
F1	Mainland Hakka	20	563.68	165.08	57.96***
	Taiwan Hakka	30	521.71	157.94	
F2	Mainland Hakka	20	1654.35	427.86	2.964
	Taiwan Hakka	30	1627.86	462.52	
F3	Mainland Hakka	20	2810.13	349.18	12.04***
	Taiwan Hakka	30	2771.14	312.74	
F4	Mainland Hakka	20	3935.89	366.52	165.61***
	Taiwan Hakka	30	3774.48	365.48	

Note: N: number of participants; SD: standard deviations; ***$p < .001$

Table 6 displays the statistical results of formant frequencies produced by participants from different areas in Taiwan (i.e. Miao-Li Hakka vs. Mei-Nong Hakka). Significant differences could be found in three formants (i.e. F0, F3 and F4). According to the means, speakers of Mei-Nong Hakka uttered Hakka vowels with significantly higher F0 and F3, but lower F4, than those of Miao-Li Hakka. Non-significant distinction was present in the other two formants (i.e. F1, F2).

Table 6 *Formant frequencies made by participants from different areas*

	Area	N	Mean	SD	F
F0	Miao-Li Hakka	14	163.93	67.78	8.19**
	Mei-Nong Hakka	16	171.59	55.54	
F1	Miao-Li Hakka	14	526.24	147.43	1.48
	Mei-Nong Hakka	16	517.88	166.29	
F2	Miao-Li Hakka	14	1640.01	393.13	1.24
	Mei-Nong Hakka	16	1617.57	513.98	
F3	Miao-Li Hakka	14	2756.26	268.08	4.08*
	Mei-Nong Hakka	16	2783.74	345.72	
F4	Miao-Li Hakka	14	3802.86	293.93	10.90**
	Mei-Nong Hakka	16	3750.44	415.19	

Note: N: number of participants; SD: standard deviations; *$p < .05$; **$p < .01$

Formant frequencies made by participants of different genders are summarized in Table 7. Observation of Table 7 indicated that significant gender distinction was overwhelmingly present in five kinds of formant frequencies (i.e. F0, F1, F2, F3, F4). Judging from the mean values, female speakers of Hakka produced the vowels with significantly higher formant frequencies than male ones in F0, F1, F2, F3 and F4. This finding supported the report in Huang

and Tseng [25], in which the formant frequency was higher in women than in men.

Table 7 *Formant frequencies made by participants of different genders*

	Gender	N	Mean	SD	F
F0	Male	24	139.30	59.83	1512.04***
	Female	26	211.64	50.82	
F1	Male	24	500.08	139.86	193.49***
	Female	26	573.93	172.84	
F2	Male	24	1512.61	395.31	276.27***
	Female	26	1754.38	464.28	
F3	Male	24	2605.38	241.48	1384.98***
	Female	26	2953.79	308.57	
F4	Male	24	3635.42	295.73	1329.46***
	Female	26	4026.77	339.39	

Note: N: number of participants; SD: standard deviations; ***$p < .001$

Table 8 *Formant frequencies of different vowels made by participants*

	Vowel	N	Mean	SD	F
F0	ɨ	50	184.10	73.49	11.02***
	i	50	176.32	63.47	
	e	50	171.49	58.75	
	a	50	164.73	55.10	
	o	50	176.33	67.37	
	u	50	188.82	73.59	
F1	ɨ	50	467.74	118.96	757.24***
	i	50	401.30	117.83	
	e	50	528.21	118.32	
	a	50	745.23	120.14	
	o	50	617.13	112.65	
	u	50	472.40	108.37	
F2	ɨ	50	1629.12	241.08	1460.49***
	i	50	2221.05	299.78	
	e	50	1983.27	283.41	
	a	50	1511.95	201.03	
	o	50	1201.08	238.44	
	u	50	1280.93	296.28	
F3	ɨ	50	2814.29	276.89	152.46***
	i	50	3017.54	296.66	
	e	50	2751.59	280.57	
	a	50	2556.83	297.09	
	o	50	2746.99	327.66	
	u	50	2831.85	311.35	
F4	ɨ	50	3998.34	347.02	58.74***
	i	50	3955.66	363.51	
	e	50	3824.57	399.71	
	a	50	3725.03	392.20	
	o	50	3740.39	319.84	
	u	50	3792.29	330.67	

Note: N: number of participants; SD: standard deviations; ***$p < .001$

Table 8 lists formant frequencies of six vowels [ɨ, i, e, a, o, u] produced by participants in the present analysis. Statistical results showed that these vowels differed significantly in their formant frequencies (i.e. F0, F1, F2, F3, F4). It was interesting to note that F1 value of high vowels [ɨ, i, u] was generally lower than that of low vowels [a] and that F2 value of front vowels [ɨ, i, e] was higher than that of back vowels [o, u]. These patterns exemplified the interaction between the location of vowels and their acoustic qualities [28, 29, 30]. The vertical dimension (F1) is inversely equivalent to the tongue height, while the horizontal dimension (F2) corresponds closely to the tongue advancement.

5   Conclusion

This paper addressed the issue of acoustic normalization of Hakka vowel production in Mainland China and in Taiwan. The vowel-intrinsic procedures, including HZ, LOG, BARK and MEL, were successful indicators of different linguistic variables. These vowel-intrinsic procedures helped significantly differ vowel qualities of different accents (i.e. Mainland Hakka vs. Taiwan Hakka), vowel qualities of different areas (i.e. Miao-Li Hakka vs. Mei-Nong Hakka in Taiwan), vowel qualities of different genders (i.e. male vs. female), and different phonetic variables (i.e. vowel types, tone types, positions). As for the vowel-extrinsic procedures, LOBANOV and NEAREY1 served as two indicators less successful than NEAREY2 and GERSTMAN. This finding, however, is somewhat in disagreemet with that in Adank, Smits and van Hout [2], in which the most successful three indicators were LOBANOV, NEAREY1 and GERSTMAN.

Additionally, this study contributed to a number of linguistic issues, inclusive of language migration, nativization and language separation. To begin with, Taiwan Hakka differs significantly from Mainland Hakka in most of the normalization procedures (i.e. HZ, LOG, BARK, MEL, NEAREY2, GERSTMAN), implying the fact that language migration from Mainland Hakka to Taiwan Hakka has brought about certain language changes between these two accents. Secondly, due to the effect of language contact, Taiwan Hakka has been influenced by some other languages in Taiwan (e.g. Taiwan Mandarin or Taiwan Min), leading to a process of nativization (Chung [13]). It is this kind of language integration or mixing that makes the accent of Hakka in Taiwan different from the source accent of Hakka in Mainland China. Thirdly, the current analysis confirms the sutble, but distinctive, patterns between Miao-Li Hakka and Mei-Nong Hakka, both of which belong to Si-Xian Hakka, but are spoken in two regions. In the present investigion, Miao-Li Hakka and Mei-Nong Hakka are significantly different from each other in several normalization procedures. Speakers of Mei-Nong Hakka produced Hakka vowels with significantly higher F0 and F3, but lower F4, than those of Miao-Li Hakka. It is interesting to note that language separation makes Hakka people in these two areas of significantly different accents. This finding echoes the fact that Ministry of Education (MOE) in Taiwan published respective versions for these two sub-types of Si-Xian Hakka in Taiwan (MOE [31]), that is, Si-Xian (for the Miao-Li accent) and Southern Si-Xian (for the Mei-Nong accent).

Moreover, this paper adopted an automatic approach to analyze the acoustic data and set a model of spectral analysis for future research. Further, the sound analyzer or the spectral display, as shown in *PRAAT*, is also a useful instrument in the process of pronunciation instruction. With the visual cues in productive and receptive training, language learners may be provided the solid evidence of their pronunciation in comparison with the native pronunciation. A visual display of the F1 versus F2 plane [32, 33] may be instructive in promoting accurate vowel production or perception. Learners may also check on themselves and train the auditory judgments as precisely as the analytical data in the instruments of modern technology.

6   References

[1] Lai, Y. H. (2006). *A phonetic and phonological study on English vowel acquisition for Taiwanese EFL learners*. Unpublished doctoral dissertation. National Kaohsiung Normal University, Taiwan, ROC.

[2] Adank, P., Smits, R., & van Hout, R. (2004). A comparison of vowel normalization procedures for language variation research. *Journal of the Acoustical Society of America*, 116(5), 3099-3107.

[3] Flynn, N., & Foulkes, P. (2011). Comparing vowel formant normalization methods. *Proceedings of ICPhS XVII* (pp. 683-686), China: Hong Kong.

[4] Thomas, E. R., & Kendall, T. (2007). NORM: The vowel normalization and plotting suite. Online Resource. http://ncslaap.lib.ncsu.edu/tools/norm, accessed:17/11/08.

[5] Watt, D. J. L., Fabricius, A. H., & Kendall, T. (2010). More on vowels: Plotting and normalization. In Di Paolo, M., & Yaeger-Dror, M. (Eds.), *Sociophonetics: A student's guide* (pp. 107-118). London: Routledge.

[6] Adank, P. (2003). *Vowel normalization: A perceptual-acoustic study of Dutch vowels*. Ponsen & Looijen bv, Wageningen.

[7] Chang, P. S. (1997). Relevant problems in trasncriptions of Hakka segments. *Proceedings of Conference of Taiwan Language Development*. National Hsinchu Normal College.

[8] Huang, Y. H. (2004). *On Si-hai Hakka in southern Taiwan*. Unpublished M.A. thesis. National Kaohsiung Normal University.

[9] Luo, Z. J. (1998). *Taiwan Hakka*. Taipei: Tai-Yuan.

[10] Yang, S. F. (1971). Mei-nung Hakka dialect in Taiwan. *Bulletin of the Institute of History and Philology*, 42, 405-456.

[11] Chung, R. F. (1990a). On the status of Hakka prevocalic glides. *Taiwan Culture*, 40(4), 189-198.

[12] Chung, R. F. (1990b). The structure of Hakka rime. *The Sinology Review*, 8(2), 57-78.

[13] Chung, R. F. (2003). *Studies on Hakka Phonology*. Taipei: Wunan Publishing Company.

[14] Cheng, M. C. (2010). Generational variation of the apical vowel in Hakka and its phonological explanations. *Journal of Taiwanese Languages and Literature*, 5, 103-119.

[15] Liang, C. W. (2005). *The acoustic characteristics of Hakka consonants and vowels*. Unpublished M.A. thesis. National Kaohsiung Normal University, Taiwan, ROC.

[16] Tan, Y. C. (2010). *Vowel pattern of sixian hakka: An acoustic perspective*. Unpublished M.A. thesis. National United University, Taiwan, ROC.

[17] Boersma, P., & Weenink, D. (1999-2000). *Praat, a system for doing phonetics by computer* [Computer program]. Web site: www.praat.org.

[18] Gerstman, L. (1968). Classification of self-normalized vowels. *IEEE Transactions of Audio Electroacoustics AU-16*, 78-80.

[19] Lobanov, B. M. (1971). Classification of Russian vowels spoken by different speakers, *J. Acoust. Soc. Am. 49*, 606-608.

[20] Nearey, T. M. (1978). *Phonetic feature systems for vowels* (Indiana University Linguistics Club, Indiana).

[21] Nordström, P. E. (1976). Female and infant vocal tracts simulated from male area functions. *Journal of Phonetics*, 5, 81-92.

[22] Syrdal, A. K., & Gopal, H. S. (1986). A perceptual model of vowel recognition based on the auditory representation of American English vowels. *Journal of the Acoustical Society of America*, 79, 1086-1100.

[23] Ainsworth, W. A. (1975). Intrinsic and extrinsic factors in vowel judgments. In Fant, G. & Tatham, M. A. A. (Eds.) *Auditory analysis and perception of speech* (pp.103-113). London: Academic.

[24] Nearey, T. M. (1989). Static, dynamic, and relational properties in speech perception. *Journal of the Acoustical Society of America*, 85, 2088-2113.

[25] Huang, K. U., & Tseng, C. H. (1995). Variability of vowel formant frequencies and speaker normalization: The cases of Mandarin and Taiwanese. *Journal of the Acoustical Society of America*, 98(5), 2966-2966.

[26] Chung, R. F. (2011). Language contact and language change in Taiwan Hakka. An invited speech at the study seminar. *Southern Underlying Effect from the Perspective of Language Contact*. National Tsing-Hua University, Taiwan, ROC.

[27] Huang, Y. H. (2009). *Praat scripts for fetching pitch and formants of vowels*. ms.

[28] Ladefoged, P. (1993). *A course in phonetics* (3rd edition). N.Y.: Harcourt Brace Jovanvich Inc.

[29] Ladefoged, P. (2001a). *Vowels and consonants*. Mass: Blackwell.

[30] Ladefoged, P. (2001b). *A course in phonetics* (4th edition). Orlando: Harcourt Brace College.

[31] MOE (2011). *Graded Hakka materials*. Complied by the National Academy for Educational Research. Taipei: MOE.

[32] Chen, Y. (1999). *Acoustic characteristics of American English produced by native speakers of Mandarin*. Unpublished doctoral dissertation. University of Connecticut.

[33] Chen, Y., Robb, M., Gilbert, H., & Lerman, J. (2001). Vowel production by Mandarin speakers of English. *Clinical Linguistics & Phonetics*, 15(6), 427-440.

The Information Structure of the Domain "Intellectual Technologies and Computational Linguistics": Texts, Corpus, Context

Elena Yagunova and Anna Savina

St. Petersburg State University, St. Petersburg, Russia

Abstract — *The paper presents the study of the terminology, subdomain interaction, information structure of the text and the corpus. The goal of this research is to determine the distribution features of lexis that help to distinguish common and subdomain terminology. Our objective is to identify keyword features as the most informative structural elements, describing the scientific domain of the corpus and the text position within this corpus.*

Keywords: information structure, domain "intellectual technologies and computational linguistics", texts, corpus, context

1   Introduction

Our aim is to determine the information structure of the following:

- the interdisciplinary domain "Intellectual technologies and computational linguistics" (*for short — IT&CL*);
- the scientific text and corpus.

Contemporary terminology researches have many focuses (see [1]). Within this study we deal with questions about terminology, subdomain interaction, information structure of the text and the corpus, especially the degree of their homogeneity, the core and peripheries.

Is the terminology for this interdisciplinary area stable (in Russian and other languages other than English)?

The goal of this research is to determine the distribution features of lexis that are able to distinguish common and subdomain terminology. Our objective is to identify keyword features as the most informative structural elements, describing the scientific domain of the corpus and the text position within this corpus.

We restricted the task to the study of the following:

- the text as a context for the keyword;
- the corpus as a context for the text.

The scientific domain of the corpus (corpora) varied from "Corpora Linguistics" (monothematic) to "Theory of Language & Computational Linguistics" (polythematic).

2   Data and Methods

2.1   Data

The data for the investigation of the information structure for such domain as "IT&CL" is four corpora based on proceedings of the four most representative Russian international conferences dedicated to IT&CL. We assume that each corpus represents its own subdomain of the IT&CL topic. So, proceedings of such conferences as "Corpora linguistics" of 2004-2008 (*for short — CL*) and "Dialogue" of 2003-2009 represent more linguistic topics, while proceedings of RCDL and CAI — more technical topics.

The material used for the study of the text structure and corpus structure consists of (1) the CL corpus (i.e. monothematic and homogeneous corpus of the analyzed domain) and (2) separate texts from the CL corpus.

All the corpora have a morphological annotation.

2.2   Methods

The express analysis of the corpora information structure is based on several measures:

- TF-iDF for keywords;
- MI (Mutual Information) for collocations.

The additional study includes the comparison of the frequency of a particular term a particular corpus and the

frequency of the same term in all the corpora and weighting of these terms using the weirdness measure (KWw) (see [2], [3]).

The keyword detection was verified by the result of a computational experiment (KWc) and the experiment with the informants (KWi). We had the comparative analysis of these two sets: KWc & KWi.

The visual analysis of the distribution of words (for keywords) (1) in the space of the text (for 10 texts of a subset) and (2) in the space of CL corpus is made with D. Lande's tools [4,5].

The computational experiment is based on the TF-iDF [6] as a traditional measure of the importance of words in the most simple (mathematical or calculation) form. We have defined the TF (term frequency) in the simplest case as the occurrence count of a term in a document (see [7]). The iDF (inverse document frequency) is the measure of the general importance of the term.

A high value of TF–iDF is reached by a high term frequency (in the given document) and a low document frequency of the term (in the whole collection).

We use comparative criterions to discover the words with the most keywordness degree. First, we extract the set of terms using the method, which is based on comparison of a term frequency in a particular corpus and the frequency of the same term in all the corpora [8]. Second, we rank and weight these terms using the weirdness measure [9].

The experiment with informants was conducted in the traditional method with the instruction. The group of informants consisted of more than 21 students ("Mathematical Linguistics" and "Applied Informatics in Arts and Humanities" programs), who were familiar with the domain "Corpus linguistics". All the informants were given the task to read the text and write down 10-15 of the most important (for its content) words.

Each keyword selection (KW) was evaluated according to its weight (coefficient of importance):
- the value of TF-iDF (KWc);
- the amount of informants, who had written the word as a keyword (KWi).

3   Results. Discussion

3.1   Corpora in the context of the interdisciplinary domain "IT&CL"

We have received the data, verifying our hypothesis of the role of context in the analysis of texts and collections. The proposed method allows us to study the basic features of the information structure of scientific texts (subject area). We evaluated the sets of keywords (it helps to evaluate the degree of compactness of the information structure of texts and collections).

What is the domain specific of CL? The set of words with the most keywordness degree for CL (weighted using the weirdness measure in decreasing meanings of measure) includes: *корпусной* /corpus/, *национальный* /national/, *параллельный* /parallel/, *неоднозначность* /ambiguity/, *формат* /format/, *буква* /letter/, *омонимия* /homonymy/, *частотный* /frequency/, *падеж* /case/, *частота* /frequency/, *критерий* /criterion/, *ошибка* /mistake/, *категория* /category/, *текстовой* /text/, *предлог* /preposition/, *специальный* /special/, *база* /base/.

The set of keywords (lemmata) of CL (extracted based on the TF-iDF in decreasing meanings of measure) is: *текст* /text/, *корпус* /corpus/, *слово* /word/, *словарь* /dictionary/, *семантический* /semantic/, *система* /system/, *единица* /unit/, *предложение* /sentence/, *русский* /Russian/, *тип* /type/, *анализ* /analysis/, *глагол* /verb/, *лингвистический* /linguistic/, *работа* /work/, *следовать* /to follow/, *форма* /form/, *информация* /information/, *создание* /creation/, *контекст* /context/, *исследование* /research/, *класс* /class/, *речь* /speech/, *часть* /part/, *морфологический* /morphologic/, *признак* /feature/, *результат* /result/, *корпусной* /corpus/, *лексический* /lexical/, *структура* /structure/, *грамматический* /grammatical/, *описание* /description/, *задача* /task/, *синтаксический* /syntactic/, *пример* /example/, *словоформа* /wordform/, *поиск* /retrieval/, *лингвистика* /linguistics/.

The intersection between CL and other corpora: *текст* /text/, *значение* /meaning/, *система* /system/, *тип* /type/, *анализ* /analysis/, *работа* /work/, *информация* /information/, *результат* /result/, *структура* /structure/, *задача* /task/.

The top of keywords (lemmata) of the other three corpora (extracted based on the TF-iDF in decreasing meanings of measure):
CAI — *система* /system/, *решение* /decision/, *знание* /knowledge/, *модель* /model/, *задача* /task/, *множество* /multitude/, *объект* /object/;
RCDL — *система* /system/, *ЭБ* /digital library/, *документ* /document/, *ресурс* /resource/, *работа* /work/, *информация* /information/, *поиск* /retrieval/;
Dialogue — *слово* /word/, *текст* /text/, *значение* /meaning/, *система* /system/, *русский* /Russian/, *случай* /case/, *предложение* /sentence/, *пример* /example/;
so CAI and RCDL are more "technical", Dialogue — "linguistic" conferences. The words related to the intersection

of the sets of CL and others corpora have underlying font selection.

According to our results the IT&CL domain has rather technical than linguistic terminology: 87% of keywords extracted from the two "technical" proceedings are unique, i.e. not being presented in the "linguistic" proceedings; whereas only 67% of keywords extracted from "linguistic" conferences are unique. The number of shared keywords within one subdomain — either "technical" or "linguistic" — is rather small and do not exceed 10% of keywords that demonstrates heterogeneous nature of the research area. CL and Dialogue have 29% "technical" terminology.

What is the domain specific of the other corpora? What is the main difference of CL from the other corpora? The best answer is the sets (the tops of the sets) of words with the most keywordness degree for these – others – corpora (weighted using the weirdness measure in decreasing meanings of measure):

CAI — *нечеткий* /fuzzy/, *следующий* /next/, *принятие* /acceptance/, *среда* /environment/, *обучение* /learning/, *интеллект* /intellect/, *заключение* /conclusion/, *логический* /logical/, *моделирование* /modeling/, *критерий* /criterion/, *эксперт* /expert/, *искусственный* /artificial/, *определять* /to define/, *атрибут* /attribute/, *рассуждение* /reasoning/, *робот* /robot/, *программный* /programmatic/, *обучать* /to learn/;

RCDL — *ЭБ* /digital library/, *ВЕБ* /web/, *метаданные* /metadata/, *сервис* /service/, *каталог* /catalog/, *библиотека* /library/, *архив* /archive/, *сервер* /server/, *коллекция* /collection/, *доступ* /access/, *публикация* /publication/, *распределенный* /distributed/, *интеграция* /integration/, *индикатор* /indicator/, *стандарт* /standard/, *ссылка* /link/, *обеспечение* /support/, *Россия* /Russia/;

Dialogue — *посессор* /possessor/, *аллофон* /allophone/, *акцентоноситель* /word marked with accent/, *тринотация* /tree-annotation/, *инфопортрет* /infoportrait/, *селькупский* /Selkup/, *фонограмма* /phonogram/, *непереходный* /intransitive/, *дейктический* /deictic/, *фразема* /phrase/, *жестовый* /sign/, *намек* /hunt/, *гласный* /vowel/, *нарратив* /narrative/, *повествование* /narration or narrative/, *невербальный* /nonverbal/, *указательный* /indicating/, *пауза* /pause/.

It is important that the sets of words with the most keywordness degree – weighted using the weirdness measure in decreasing meanings of measure – have the minimum of the intersections in contradistinction to the sets of keywords weighted using TF-iDF measure

3.2 The text in the context of corpus "CL"

What texts are thematically central (in context of collection) and what texts are in periphery? Characteristics of

the information structure of the text were analyzed by comparing KWi and KWc sets.

We define the main topics both with the help of the keywords and the terminology bigrams (the top of MI-collocations): *речевая деятельность* /speech perception and speech production/, *художественная литература* /fiction/, *корпусная лингвистика* /corpus linguistics/, *математическая лингвистика* /mathematical linguistics/, *словарная статья* /dictionary entries/, *предметная область* /domain/, *машинный перевод* /machine translation/, *семантическое состояние* /semantic state/, *разрешение неоднозначности* /disambiguation/, *английский язык* /English/, *грамматическая категория* /grammatical categories/, *база данных* /database/, *лексическая единица* /lexical item/, *русский язык* /Russian/, *часть речи* /part of speech/, *морфологическая разметка* /morphological annotation/, etc. ([10], [11]).

Examples of the text thematically central to the collection and the text on the periphery may serve as a "Text1" and "Text 2". The assignment of texts to these groups should be based on (1) the analysis of each of the two sets of keywords (KWc and KWi), (2) comparing the KWc & KWi sets and (3) comparing the characteristics of the text with the characteristics of the collection as a whole (sets of keywords and terminology collocations).

Text №1. KWi with a maximum weight (coefficient of importance) contain the terminology, which was determined by the terminology collocation: *словарь* /dictionary/, *корпус* /corpus/, *текст* /text/, *термин* /term/, etc. KWc set contains keywords that are relevant to a text in relation to CL. These keywords occur in this text very often, and it is enough to high values of TF–iDF.

At the intersection of the sets of KWc and KWi there are the following: *термин* /term/, *модуль* /module/, *обучение* /learning/, *тематизация* /theming/, *конкорданс* /concordance/, *иерархия* /hierarchy/, *встречаемость* /occurrence/, *наполнение* /content/, *статистический* /statistical/, *словарь* /dictionary/, *классификация* /classification/, *предметный* /object/. The intersection is about 50% of each list. This intersection is compatible with the information structure of the collection (see table I).

TABLE I.   KWc & KWi SETS FOR THE TEXT №1
(THE TEXT THEMATICALLY CENTRAL TO THE COLLECTION)

KWc	KWi
термин /term/	словарь /vocabulary or dictionary/
модуль /module/	корпус /corpus/
обучение /learning/	текст /text/

KWc	KWi
тематизация /theming/	**термин /term/**
конкорданс /concordance/	**конкорданс /concordance/**
статистика /statistics/	анализ /analysis/
иерархия /hierarchy/	**статистический /statistic/**
тема /topic/	автоматический /automatic/
неразмеченный /non-marked/	**модуль /module/**
просматривать /to look through/	**обучение /learning/**
обучать /to learn/	**тематизация /theming/**
сообщение /report/	разметка /markup/
встречаемость /occurrence/	**предметный /subject/**
наполнение /content/	обработка /processing/
статистический /statistic/	структура /structure/
выборка /sampling/	**иерархия /hierarchy/**
словарь /vocabulary or dictionary/	информация /information/
дообучение /learning/	лексический /lexical/
словокомплекс /word-complex/	**наполнение /content/**
терминообразовать /to form a term/	**классификация /classification/**
классификация /classification/	технология /technology/
наследование /inheritance/	**встречаемость /occurrence/**
механизм /mechanism/	выборка /sampling/
документооборот /documents circulation/	частота /frequency/
словарный /lexical/	создание /creation/
предметный /subject/	**словарь /dictionary/**

Words related to the intersection of the sets of KWc and KWi have bold font selection.

Text №2. KWi with a maximum weight rarely correspond to the terminology, which was determined by the collocation of terminology.

There are fewer such comparisons for a set of KWc (KWc represents a main part of the phonetic terminology). The intersection of the sets includes *звук* /sound/, *фонема* /phoneme/, *комбинация* /combination/, *сочетание* /combination/, at all — no more than 16% intersections (see table II).

TABLE II.   KWc & KWi SETS FOR THE TEXT №2
(THE TEXT ON THE PERIPHERY)

KWc	KWi
звук /sound/	язык /language/
фонема /phoneme/	**фонема /phoneme/**

KWc	KWi
комбинация /combination/	фонотактика /phonotactics/
согласный /consonant/	поиск /retrieval/
взрывной /plosive/	**иностранный /foreign/**
задний /back/	**звук /sound/**
преграда /barrier/	**транскрипция /transcription/**
сонант /sonant/	программа /program/
транскрипционный /transcriptional/	словарь /dictionary/
передний /front/	**комбинация /combination/**
позиция /position/	анализ /analysis/
редукция /reduction/	<звуковая> оболочка /<sound> shell/
помочь /to help/	**звуковой /sound/**
сочетание /combination/	система /system/
британский /British/	текст /text/
гласная /vowel/	статистический /statistic/
иноязычный /foreign language /	**сочетаемость /compatibility/**
иностранный /foreign/	поисковый /searching/
безусловно /certainly/	электронный /electronic/
английский /English/	**сочетание /combination/**
альвеолярный /alveolar/	перекодировка /conversion/
англичанин /Englishman/	модель /model/
апикальный /apical/	изучение /study/
аффриката /affricate/	слово /word/
боковой /side/	проблема /problem/

Words related to the intersection of the sets of KWc and KWi have bold font selection.

Keywords in the table III are in the decreasing order by their weight. Words related to the intersection have bold font selection.

TABLE III.   iDF FOR WEIGHTING KWC

Text 1		Text 2	
Word	*iDF*	*Word*	*iDF*
термин /term/	1.1	**звук /sound/**	2.6
модуль /module/	1.8	**фонема /phoneme/**	3.7
обучение /learning/	1.5	**комбинация /combination/**	1.9
тематизация /theming/	4.8	согласный /consonant/	3.2
конкорданс /concordance/	2.1	взрывной /plosive/	4.1
статистика /statistics/	3.2	задний /back/	4.8
иерархия /hierarchy/	2.1	преграда /barrier/	4.8

Text 1		Text 2	
Word	*iDF*	*Word*	*iDF*
тема /topic/	0.7	сонант /sonant/	4.8
неразмеченный /non-marked/	4.8	**транскрипционный /transcriptional/**	4.8
просматривать /to look through/	4.8	передний /front/	4.1
обучать /to teach/	2.3	позиция /position/	1.2
сообщение /report/	2.3	редукция /reduction/	3.7
встречаемость /occurrence/	2.2	помочь /to help/	3.4
наполнение /content/	2.6	**сочетание /combination/**	1.1
статистический /statistic/	1.0	британский /British/	3.0
выборка /sampling/	1.7	гласная /vowel/	3.0
словарь /dictionary/	0.3	иноязычный /lingual/	3.0
дообучение / additional training/	4.8	иностранный /foreign/	1.0
словокомплекс /word-complex/	4.8	безусловно /certainly/	2.6
терминообразовать /to form a term/	4.8	английский /English/	1.2
классификация /classification/	1.2	альвеолярный /alveolar/	4.8

We divide the terms into tree classes:
- iDF>4 => the term presents less than 20%,
- iDF<2 => more than 40%,
- iDF <=4 & >=2.

This division allows one to extract terminology:
- Specific terms for the certain text,
- Common terms for the corpora (the main topics),
- Terms intermediate nature

The interpretation of the results includes the spectrograms to visualize the density of occurrence of words in a text or a corpus, depending on the width of the observation window [5].

Examples of interpretation keywords within text №1 (the text thematically central to the corpus CL) are presented on spectrograms 1-5:
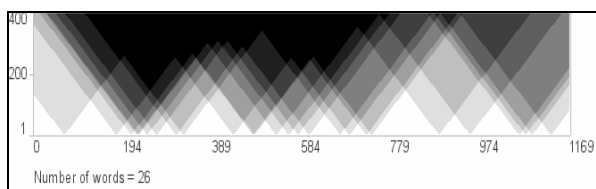
SPECTROGRAMM 1. "TERM (TERMINOLOGY)" DISTRIBUTION
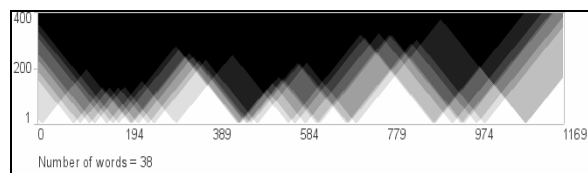
SPECTROGRAMM 2. "VOCABULARY" DISTRIBUTION
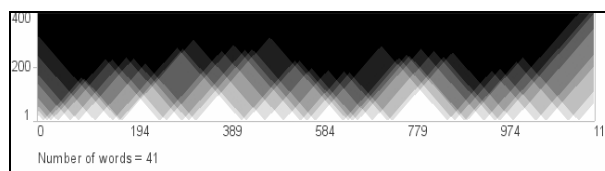
SPECTROGRAMM 3. "TEXT" DISTRIBUTION
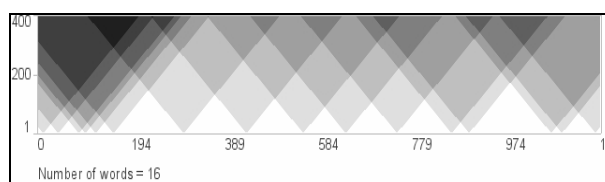
SPECTROGRAMM 4. "CORPUS (CORPORA)" DISTRIBUTION
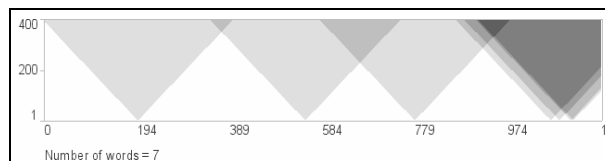
SPECTROGRAMM 5. "CONCORDANCE" DISTRIBUTION

Why these words became keywords for the text №1? The answer includes the information about not only frequency, but also about distribution of words: regular or irregular, concentration near the beginning or near the end of the text. Such analysis allows us to understand what the place of the keywords in the text information structure (especially for the scientific text) is. Moreover — the distributions specify, where it means object, or method, or result. It depends on a regular (or irregular) distribution, localizations in the certain composition segment (according to the principles of scientific academic text structuring), concentration near the beginning or near the end of the text and so on (see [12], [13]). Thus we can say that distributions specify some features of keywords and the most probable roles in the text frame.

We can say that such keywords as "terminology", "text" (or text structure) and "vocabulary" are the main objects for the text №1 and it is argued by their distribution in space of text. The other distribution would be characterized, for example, result or data.

Examples of interpretation keywords within the corpus CL are presented on spectrograms 6-10 (text №1 is the thematically central to this corpus and it is essential to see some similarity between spectrograms 1-5 spectrograms 6-10):
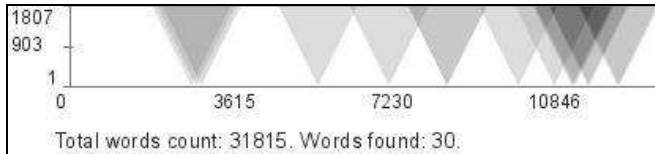
SPECTROGRAMM 6. "TERM (TERMINOLOGY)" DISTRIBUTION

SPECTROGRAMM 7. "VOCABULARY" DISTRIBUTION

SPECTROGRAMM 8. "TEXT" DISTRIBUTION

SPECTROGRAMM 9. "CORPUS (CORPORA)" DISTRIBUTION
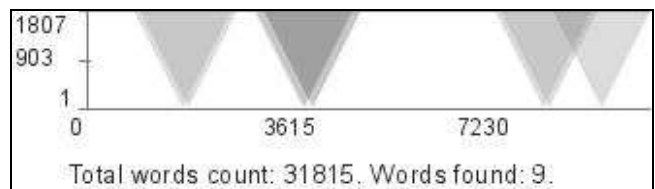
SPECTROGRAMM 10. "CONCORDANCE" DISTRIBUTION

Why these words became keywords for the monothematic corpus CL? The answer includes information about not only frequency, but also about distribution of words: regular or irregular, concentration near the beginning or near the end of the text (as it was for keyword in text context, certainly for the text in the nuclear of topic area (nuclear for the corpus). Such analysis allows us to understand what the place of the keywords in the corpus information structure (especially for the scientific text) is. Moreover — the distributions can specify, where it means object, or method, or result. It depends on the main features of distribution (more regular / more irregular distributions, specific localizations), but not — concentration near the beginning or near the end of the corpus.

4   Conclusions

We have determined the distribution features of lexis that distinguish common and subdomain terminology. We suppose that one of the most important results of our study is the methodology for identifying and extracting keyword features as the most informative structural elements, describing the scientific domain of the corpus and the text position within this corpus. Further directions of our research have the focus on the evaluation. Our plans include evaluation of the experimental series with informants and experts, where informants are master and PhD students studying computer science and/or linguistics. We want to collect informants for the different subdomains within the interdisciplinary domain "IT&CL".

5   Acknowledgment

6   References

[1]   F M.T. Cabré and R. Estopà On the units of specialized meaning used in professional communication in Journal of International Institute for terminology research (IITF) Vol. 14, 2003, pp. 16-27

[2]   E. Yagunova and L. Pivovarova "The Nature of Collocations in the Russian Language. The Experience of Automatic Extraction and Classification of the Material of News Texts" in Automatic Documentation and Mathematical Linguistics, vol. 44, No. 3, 2010, pp. 164–175.

[3]   L. Pivovarova and E. Yagunova "Izvlechenie i klassifikaciya terminologicheskih kollokacii na materiale lingvisticheskih nauchnyh tekstov (predvaritel'nye nablyudeniya)" in Materialy Simpoziuma "Terminologiya i znanie" (Moskow, 21-22 May 2010) Moscow, 2010, pp. 214-229.

[4]   D. Lande "Vizualizaciya statistiki vhozhdeniya slov" in MegaLing'2009. Gorizonty prikladnoi lingvistiki i lingvisticheskih tehnologii, Kiev, Ukraina, 2009, pp. 63-64.

[5]   Web   site   for   statistical   tools: http://ling.infostream.ua/jag/jag.html

[6]   G. Salton and C. Buckley "Term-weighting approaches in automatic text retrieval" in Information Processing and Management, № 24(5), 2010, pp. 513-523.

[7] Ch. D. Manning, P. Raghavan and H. Schütze "Introduction to Information Retrieval", Cambridge University Press, Cambridge, England, 2009.

[8] E. Yagunova and D. Lande "Dinamicheskie chastotnie harakteristiki kak osnova dlya strukturnogo opisaniya raznorodnix lingvisticheskix obektov" in Trudi 14-y Vserossiiskoi nauchnoy konfrencii RCDL-2012, Pereyaslavl-Zalesskii, Russia, 2012.

[9] I. Chetviorkin and N. Loukachevitch "Extraction of Domain-specific Opinion Words for Similar Domains" in Proceedings of the Workshop on Information Extraction and Knowledge Acquisition, Hissar, Bulgaria, 2011, pp.7-12.

[10] E. Yagunova and A. Savina "The Experimental Study of Terminology Collocations: Calculations and Experiments with Informants" in Proceedings of the NODALIDA 2011 workshop CHAT 2011: Creation, Harmonization and Application of Terminology Resources NEALT Proceedings Series, vol. 12, pp. 40-44

[11] L. Pivovarova and E. Yagunova "Informacionnaya struktura nauchnogo teksta. Tekst v kontekste kollekcii" in "Korpusnaya lingvistika–2011", Saint-Petersburg, Russia, 2011.

[12] E. Yagunova "Formal'nye i neformal'nye kriterii vychleneniya klyuchevyh slov iz nauchnyh i novostnyh tekstov" in Materialy IV Mezhdunarodnogo kongressa issledovatelei russkogo yazyka "Russkii yazyk: istoricheskie sud'by i sovremennost'", Moscow, Russia, 2010, pp. 533-534.

[13] E. Yagunova "Eksperiment i vychisleniya v analize klyuchevyh slov hudozhestvennogo teksta" in Sbornik nauchnyh trudov kafedry inostrannyh yazykov i filosofii PNC UrO RAN. Filosofiya yazyka. Lingvistika. Lingvodidaktika, Perm', Russia, 2010, pp. 85-91.

Optimal Automated Method for Collaborative Development of University Curricula

M. Charnine[1] and V. Protasov[2]
[1]Institute for Informatics Problems, Russian Academy of Sciences, Moscow, Russia
[2]The Russian Center of Computing for Physics and Technology, Moscow, Russia

Abstract – *Aspects of curriculum design automation using semantic analysis of Web texts to develop candidate keywords and voting alternatives, as well as methods of optimal aggregation of individual expert decisions are presented. It is shown how semantics analysis of online texts can help with: generating of initial version of the curriculum; generating of candidate keywords and alternative options for improving the curriculum; building hierarchy of categories of the subject area or domain; detecting the change of domain knowledge state and defining structural knowledge dependencies; monitoring terminology and knowledge base of the subject area or domain. It is shown how the techniques of optimal integration of individual expert decisions in selecting of automatically generated alternatives helps to create the highest quality curriculum.*

Keywords: curriculum design; semantic analysis of Internet texts, competency assessment, optimal aggregation rules, voting records

1 Introduction

At present, many educational institutions participate in intensive methodological work on development of educational standards, appropriate training programs and curriculums. The high education curriculum must meet the requirements of quality education and follow the speed of technological and scientific revolution boosted by progress in information and communication technologies. At the same time, the study of the semantics of terabytes of Internet texts can help to upgrade the curriculum, making possible fully automated monitoring of the knowledge base.

Next generation of curriculum development systems requires:

•   revision of the curriculum development process, including information processing scheme and the use of semantic web searches based on keywords;

•   the use of semantic analysis of large volumes of online texts to determine the key terms/keywords of the domain and their semantic relations;

•   the use of automated tools to maintain relevance of term/keyword to the specific subject area of curriculum;

•   participation of a large number of experts in curriculum development;

•   the use of automated tools for collective decision support.

2 Automated tools for curriculum design

The university curriculum is a document which summarizes the structured content, knowledge, skills and abilities for learning by students. The material is usually divided into sections and themes, contains a hierarchy of sections and the sequence of their study.

In many cases, curriculum can be formally presented as a list of keywords/terms of the subject area and a set of hierarchical relationships between them that form a tree-like structure. Such hierarchical structure of the curriculum can be created by a group of experts that have to:

- select keywords/terms for inclusion in the curriculum from a list of automatically generated candidate terms;

- build a hierarchy of selected terms by specifying hierarchical (parent-child) relationships between them.

Experts can use so-called associative portrait of subject area that can be generated automatically to reduce the amount of work on keyword/term selection.

Associative Portrait of Subject Area (APSA) – is a set of associative relationships between keywords/terms that are relevant to the subject area. Automatic methods of finding associative relationships between terms are based on the calculation of their semantic similarity. If semantic similarity between two terms is high then we consider that these terms have an associative relationship. Semantic similarity is also called as semantic distance or connectivity. The method of calculation of semantic similarity is presented below.

2.1 Automatic method of analysis and calculation of semantic similarity

Most automated tools and technologies for curriculum design are based on the method for calculation of semantic connectivity and distance usually called "semantic similarity of terms".

Method of calculation of semantic similarity of terms is based on statistical analysis of large bodies of text from the Internet. In our work the method has been successfully used for keyword research, identification of related and core keywords, calculation of keyword/term clusters, finding categories of clusters and their hierarchical relationships.

The proposed model presents keywords/terms of the curriculum in a form of several clusters of terms that dynamically change over time in quantity, structure and even meaning.

To define, analyze and "calculate" clusters there is a need to understand a measure of semantic similarity between terms also called as semantic distance or connectivity.

The most efficient approach to calculate semantic similarity of terms is based on the use of distributional semantic models [3].

Distributional semantic models were successfully used for the following tasks: word similarity, word clustering, automatic thesaurus generation, word sense disambiguation, query expansion; enabling their application for automated curriculum design.

Distributional semantic models, variously known as vector spaces, semantic spaces, word spaces, corpus-based semantic models, distributional memory, all rely on some version of the distributional hypothesis, stating that semantic similarity between two words/terms can be modeled as a function of the degree of overlap among their linguistic contexts.

In other words distributional hypothesis [3] is claiming that semantically related lexemes/terms have similar context and, conversely, in a similar context, the lexemes are semantically close. The proposed model uses advanced hypothesis, suggesting that context similarities exist not only for semantically similar individual lexemes, but also for semantically similar phrases containing multiple lexemes and terms.

Web contains astronomical and constantly growing amount of textual documents and linguistic contexts, enabling calculation of semantic similarity of terms and curriculum keywords migration with growing precision over time.

The most modern versions of Distributional semantic models are Word Space Models and Distributional Memory Models.

Most of the criticism of these models stems from the fact that term-document and word-context matrices typically ignore word order.

In contrast with Word Space Model and Distributional Memory our approach uses more precise method of analysis and calculation of semantic distance and connectivity accounting "fine-tuning" of semantic relations and presence of word combinations and as well as keywords in a context.

Advanced method applied in our research is presented below.

2.2 Advanced method for calculation of semantic similarity

Advanced method for calculation of semantic similarity/distance uses statistical methods for selection of important terms from the body of texts.

These selected terms can be words, or word combination, or named entities. The word order within selected terms is very important and is not ignored.

Let n is the number of selected terms.

Then we create symmetric term–context matrix (n-by-n), which elements are calculated based on co-occurrence of selected terms in a corpus of texts.

When we calculate the matrix elements, we can use special lexico-syntactic patterns (e.g., x "is a" y | y "including" x | x "such as" y) to extract specific types of semantic relationships between terms (hierarchical, genus-species, the part-whole).

Each row of the matrix can be interpreted as a vector in n-dimensional coordinate space corresponding to certain selected term. We call this vector a Context Vector corresponding to certain term.

The set of such context vectors together forms a high dimensional Semantic Vector Space.

Semantic similarity between the terms x and y is calculated based on the cosine measure between the corresponding vectors according to the following formula:

$$\frac{x \cdot y}{|x||y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}}$$

The terms x and y are considered to be similar/connected/associated if the cosine measure of their vectors has a large value.

2.3   Formal definition of APSA

Formally, the APSA is defined as a graph G = (V, E) with nodes v of V, representing significant keywords/terms and links of the graph (v-i, v-j, Link, w-ij) of E, describe the relationship between terms. Here w-ij is the weight that expresses the measure of semantic similarity between terms. Link represents the type of the context of terms v-i and v-j. This context type is related to type of relationship between terms v-i and v-j. The context type is determined by the parameters of the algorithm for calculation context vectors, such as the size of the context window, or type of syntactic template binding the terms.

2.4   Automatic methods of APSA creation

The process of APSA creation implied a set of methods, including:

-identification of Internet texts related to the specified subject area or domain;

-identification of relevant keywords/terms and their ranking;

-identification and ranking of associative relationships between keywords/terms.

These methods are based on the preliminary studies on texts, including those taken from various Internet resources.

Processing large volumes of texts, constantly updated in the Internet, allows us to collect the necessary statistical data to generate a fairly complete picture of the subject area, presented in the form of APSA. The ability to perform machine learning on a large number of examples gives the system flexibility and improves the results.

Thus automatic nature of APSA creation helps to monitor terminology and knowledge base of the subject area or domain and helps to detect the change of domain knowledge state and define structural knowledge dependencies.

2.5   Method for curriculum development using APSA

APSA technology helps to identify the list of relevant keywords/terms that can be considered as candidate keywords for inclusion in the curriculum. The list of keyword candidates can be also considered as a list of voting alternatives. Experts can vote for every keyword candidate and make collective decision: include it or not in the curriculum.

APSA technology also helps to identify the list of associative relationships between keywords/terms in curriculum. These associative relationships can be considered as candidates for hierarchical relationships between terms. Experts can vote for every candidate relationship and make collective decision: include it or not in the curriculum as hierarchical relationship.

This way experts can create a hierarchy of categories/sections included in the curriculum.

2.6   Algoritm for curriculum development

Formally, the actions of experts to build the initial version of the curriculum and to identify possible changes of its keywords/terms can be described by the following algorithm.

1. Set of initial terms/keywords is given defined manually or taken from the old curriculum

2. Collect text from Internet with predefined keywords (formation of body of texts for various time slots)

3. Design context vectors that determine meaningful closeness and dependency between keywords at various moments of time and time slots (point and evolution dependency)

4. Searching clusters of closely related keywords

5. Searching of the hierarchical structure of cluster dependency (hierarchical clustering).

6. Calculate the center of the cluster of keywords/terms related to the domain or subject area.

7. Those new terms that are closer to the cluster center are candidate terms for inclusion in the curriculum.

8. Those keywords that are more distant from the cluster center are candidate terms for exclusion from

the curriculum. After deletion these candidate terms the cluster center changes its position and thus motion of the curriculum is defined.

3 Automated tools for collaborative decision making

The speed of scientific and technological revolution and the amount of knowledge in every domain is constantly growing. As a result we have a growing number of specific terms in the domain, as well as increasing rate of changes in curriculum terminology.

All this leads to the fact that we need more and more experts working together to maintain up-to-date curriculum.

In a large expert team the diversity of expert opinions and their competences can vary widely and it makes more difficult for them the quick creation of collective decisions. In other words, the larger expert team, the harder natural process of creating timely collective decisions without using special automated tools.

That's why there is a need for automated tool combining individual decisions of experts. This collaborative decision-making (CDM) tool coordinates the functions and features required to arrive at timely collective decisions, enabling all relevant experts to participate in the process. The core output of CDM tool is making better decisions.

This CDM tool can help to make better the following two types of collective decisions:

- whether or not to include a particular term in the curriculum, as well as

- which of the APSA relationships between terms can be considered as hierarchical relationships.

The CDM tool can use Internet for communication purposes and a special optimal algorithm described below for collective decisions support.

4 Algorithm for optimal aggregation of expert's individual decisions

E. Baharad, J. Goldberger, M. Koppel and S. Nitzan in the article "Beyond Condorcet: Optimal aggregation rules using voting records" describe the following optimal algorithm for combining expert decisions.

1) Select some initial weights (shares) for decisions of each expert.

2) For each expert, using the story of his decisions, calculate the probability of convergence of his individual decisions with the weighted collective decision.

3) Calculate the new weighs for each expert based on his probability calculated in step 2.

4) Repeat steps 1-3 until the new weights converge with those calculated in the previous iteration.

Baharad, Goldberger, Koppel and Nitzan indicated that the above algorithm for combining expert decisions works better than any other judgment aggregating rule; in particular, it is better than collective decision making based on a simple majority rule.

The above algorithm uses linear combination of expert decisions. The main result in Nitzan and Paroush (1982), as well as Shapley and Grofman (1984) was the claim that if the probability of a correct decision of each expert is known, then a linear aggregation rule of their decisions is optimal, and the maximum probability of correct collective decision is reached when the weights (shares) of experts are calculated by the formula $W_i = \log (P_i/1-P_i)$, where P_i is the probability of a correct decision of the expert with the number i.

Thus, the algorithm described above optimally integrates individual expert decisions into optimal collective decision.

This algorithm provides the highest quality curriculum that can be created by given group of experts with different skills.

5 Conclusions

In this paper we describe the algorithm for the optimal integration of individual expert decisions in selecting automatically generated alternatives, which were taken from the associative portrait of subject area (APSA) that was automatically calculated. The above algorithm provides the highest quality curriculum for a given group of experts with different skills.

6 Acknowledgements

7   References

[1]   E.Baharad, J.Goldberger, M.Koppel и S.Nitzan, "Beyond Condorcet: Optimal Aggregation Rules Using Voting Records", CESifo München, 2011.

[2]   A. Lenci, "Distributional semantics in linguistic and cognitive research", Rivista di Linguistica, 1, 2008, pp.1-30.

[3]   M.Baroni, A.Lenci, "Distributional Memory: A General Framework for Corpus-Based Semantics", Computational Linguistics. V.36, Issue 4, 2010, pp. 673-721.

[4]   Peter Turney, "A uniform approach to analogies, synonyms, antonyms and associations", Proceedings of COLING, Manchester, 2008, pp. 905–912.

[5]   M.Charnine, I.P.Kuznetsov, E.B.Kozerenko, "Semantic Navigator for Internet Search", Proceeding of International Conference on Machine Learning, 27-30, 2005 Las Vegas, USA, CSREA Press, pp 60-65, 2005.

[6]   Michael Charnine, Vladimir Charnine. Keywen Category Structure.// Wordclay, USA, 2008, pp.1-60.

[7]   Michael Charnine, "Keywen Automated Writing Tools", Booktango, USA, 2013, ISBN 978-1-46892-205-9.

[8]   Nitzan, S., and J. Paroush. 1985. "Collective Decision Making: An Economic Outlook", Cambridge University Press, Cambridge, England.

[9]   Nitzan, S., and J. Paroush. 1982. "Optimal Decision Rules in Uncertain Dichotomous Choice Situations." International Economic Review 23(2): 289-97.

[10] Shapley, L. and B. Grofman. 1984. "Optimizing Group Judgmental Accuracy in the Presence of Interdependencies." Public Choice 43: 329-343.

[11] R. Diamond, "Designing and improving courses and curricula in higher education: A systematic approach: Jossey-Bass," SF,1989.

[12] P. Ramsden, "Learning to teach in higher education," London: Routledge, 1992, DOI: 10.4324/9780203413937.

[13] S. Toohey, "Designing courses for higher education," Buckingham, UK: The Society for Research into HE & Open University Press, 1999.

[14] M. Robbins, I. Schagaev, J. Yip, "Effective teaching in theoretical IT modules," 33rd Annual Frontiers in Education Conference," (FIE'03), 2003, http://doi.ieeecomputersociety.org/10.1109/FIE.2003.126328 0.

[15] E. Bacon, N. Folic, N. Ioannides, I. Schagaev, "Curriculum design, development and assessment for computer science and similar disciplines," FECS'12 - The 2012 Int'l Conference on Frontiers in Education: Computer Science and Computer Engineering, Las Vegas, July 2012, Paper ID: FEC4130.

[16] E. Bacon, N. Folic, N. Ioannides, I. Schagaev, "Multiple choice answers approach: Assessment with penalty function for computer science and similar disciplines," International Journal of Engineering Education Vol.28, No.6, pp. 1-7, 2012.

[17] E. Bacon, G. Hagel, N. Folic, M. Charnine, R. Foggie, B. Kirk, I. Schagaev, G. Kravtsov, "Web-enhanced design of university curricula," FECS'13 - The 2013 Int'l Conference on Frontiers in Education: Computer Science and Computer Engineering, Las Vegas, July 2013.

Database of Equivalent Verbal Forms in a Russian-French Multivariant Parallel Corpus

Anna A. Zalizniak[1], **Dmitri V. Sitchinava**[2], **Sylvain Loiseau**[3], **Mikhail Kruzhkov**[4] **and Igor M. Zatsman**[4]

[1]Institute of Linguistics, Institute of Informatics Problems, Russian Academy of Sciences Moscow, Russia
[2]Institute of the Russian Language, Russian Academy of Sciences Moscow, Russia
[3]Institute Laboratoire LDI (Lexiques, Dictionnaires, Informatique), Université Paris 13, Paris, France
[4]Institute of Informatics Problems, Russian Academy of Sciences Moscow, Russia

Abstract - *The Russian-French parallel corpus as a part of the Russian National Corpus is being transformed into a multivariant corpus where several translation variants correspond to one and the same original text. On the basis of this corpus, a Database of functionally equivalent lexico-grammatical verbal forms is being created. The main purpose of database creation is to calculate statistical estimates of equivalences between Russian and French verbal forms. The paper discusses information technology for creating the Russian-French multivariant parallel corpus and the database.*

Keywords: parallel multivariant corpora; Russian National Corpus; XML markup for Russian-French parallel texts; lexico-gramatical forms; functional equivalence; statistical estimates of equivalences

1. Introduction

Creating databases for cross-linguistic studies is a time-consuming process, but it is justified by the richness of linguistic data, including information on lexical units and grammatical structures they provide. Data can be loaded into such databases from off-line sources or digital resources such as parallel corpora [1]. The use of parallel corpora to create databases for cross-linguistic and contrastive studies allows to streamline actual research [2-5]. Contrastive study is very productive for development of conceptualization models. Functional motivation of cross-linguistic models for machine translation and parallel texts alignment was presented in the works [6-9].

The parallel corpora within the Russian National Corpus (RNC) have been developed since 2005 [2, 3]. At the present time, within the RNC there are eight bilingual parallel corpora, with Russian as the source language and eight other languages (English, German, French, Spanish, Italian, Polish, Ukrainian, Belarusian) as target languages and one multilingual parallel corpus. The RNC includes Russian-French and French-Russian parallel corpora with a single translation for each textual pair. In 2013, the Russian-French parallel corpus will be transformed into a multivariant corpus with one Russian original and several French translations for each text. The multivariant corpus of parallel texts will be formed on the basis of the preexisting monovariant parallel corpus. In this corpus the sentence-alignment association between an original text and translated texts should be maintained.

The purpose of this paper is to outline the research questions related to the creation of the Database of functionally equivalent lexico-grammatical verbal forms using the Russian-French multivariant parallel corpus. One of the main objectives of the database creation is to obtain the statistical estimates of different types of equivalences between Russian and French verbal forms. In this paper we outline our experience with the development of information technologies for creating the Russian-French multivariant parallel corpus and the database simultaneously.

The information technology for creating the corpus and the database should support the following basic functions:

- alignment of the parallel texts with several variants of translation and XML markup for parallel texts;

- morphological annotation and lemmatization of the parallel texts;

- including the parallel texts with several variants of translation into the corpus;

- development of the database of the equivalent lexico-grammatical verbal forms;

- establishing correspondences between Russian lexico-grammatical verbal forms and their French translational equivalents, i.e. the *functionally equivalent fragments*[1];

- calculating the statistical estimates of these equivalences between Russian and French verbal forms.

The main research question is how to align and mark up the parallel texts in order to establish equivalences between Russian and French verbal forms.

---

[1] The notion "functionally equivalent fragment (FEF)" was introduced in [2] for resolving the tasks of computer linguistics within the domains of contrastive lexicology and bilingual lexicography.

2. Different translations of the same text: methodological tasks

We propose a framework for aligning several French translations of the same Russian text within the French-Russian parallel corpora. The Russian prose has been translated into French for more than 150 years, and translations of different periods greatly vary as to their accuracy, choice of lexicon and grammar, and style. A comparison of these translations can be a valid task *per se*. Lexical and grammatical studies performed on parallel corpora can reflect objective variability in the target language and can be a valuable resource for contrastive grammatical description of both languages. For instance, one can expect to observe greater variability amongst the translations when there are structural differences between the two languages (a Russian «signifié» with no clear correspondence in the French lexicon is more likely to exhibit variation in its translations). Several fruitful avenues of research are opened:

- Research may provide valuable insight into phenomena of short-term diachronical variation: some variation in the translation of the same source texts may be related to diachronic variation in the target language;

- When lexical and grammatical values differ between the two languages, the inventory of the choices made by the translators may help uncovering the values of the elements translated. For instance, how the Russian aspectual system is reflected into the French system? Which are the various options, and how are they conditioned by the context? This line of research will bring new elements to the contrastive grammar of Russian and French;

- The variability of translations may help discern the intricate parameters – cultural, stylistic, lexical and grammatical – at work in the translation process, since local choice may be interpreted in the wider context of the choices made by the translator.

Since the corpus is built on texts for which several translations are available, the corpus will rely mainly on the classical Russian fiction writers of the last 150 years. It will be based on the larger Russian-French corpora previously built for the RNC, which also contains law texts, movie scripts, and newspapers. In the literary classics, special attention will be given to specific axes of variation, in particular the ones of diachrony and genres (novel, short story, children's literature).

Quantitative methodologies for contrastive grammatical description based on aligned corpora are being elaborated. Quantitative methods may be used for identifying the most frequent cases of variability. If there are some well known cases of discrepancies between the Russian and the French linguistic systems, a systematic outline of the importance of variation in the translation according to grammatical and lexical features is still to be done and may prove to be a valuable resource. This quantitative investigation will provide an empiric basis for building a typology of translation discrepancies.

Quantitative analyses may also be used for analyzing the regular correspondences between the two linguistic systems. Factor analysis, and in particular Correspondences analysis, is well suited for this task of identifying the stable groups of contextual features across numerous instances of variants. Statistical analysis is used as a mean for providing a summary of numerous translations under examination. While each single context is of no use for drawing conclusions, and while it is also impossible to figure out the big picture when dealing with numerous collected contexts, statistical analysis is useful for mapping variants.

3. Equivalences between Russian and French verbal forms

The entry of the database is an equivalence, i.e. a pair of functionally equivalent Russian and French forms. Within the information technology for creating the Russian-French parallel corpus two types of equivalences should be distinguished:

- "translation models" – the set of French translations $\{F_1, ... F_n\}$ for a Russian verbal form R;

- "translation stimuli"– the set of Russian "stimuli" $\{R_1, ... R_n\}$, to which a French form F is a "reaction".

The equivalences of the first type (including their statistical characteristics) provide us relevant information on the structure of the Russian language, whereas the equivalences of the second type specify characteristic features of the French language. Since the main object of our interest is Russian, on the first stage we take into consideration only the equivalences of the first type, i.e. the "translation models". A further expansion of the database is possible in the direction of including data from the parallel French-Russian corpus; they will answer the question "What are the French 'stimuli' that provoke as 'reaction' to a specific Russian form?".

In our database we explore the cluster of grammatical categories known in typological studies as TAM (Tense-Aspect-Mood, see [10-14]). We impose the following restrictions: we include only finite verbal forms, and we do not consider the category of voice and other divergences due to the conceptualization modes. The set of TAM values for the languages in question is generally well established (although there is a number of controversial issues). Our list was created on the basis of the existing Russian and French grammars and the researches on Russian grammar, including corpus-based ones [15-22].

The conceptual category of modality is generally considered to be composed of two parts, viz. objective and subjective modality. Objective modality is represented by the grammatical category 'mood' and enters the TAM cluster. The set of values of the category of subjective modality constitutes another cluster (Mod): it was elaborated especially for the purposes of creating our database on the basis of grammatical descriptions [15, 16, 23, 24].

We have introduced the notion "lexical-grammatical form" (LGF); it is understood as a combination of grammatical features as such, or ascribed to a class of lexemes or even to an

individual lexeme (cf. the term "construction" in Construction Grammar [25-27]). For an equivalence as a database entry, the set of Russian LGFs constitutes the Source domain, the set of French LGFs – the Target domain (which is naturally not limited by verbal forms).

A database entry is conceived as having a three-level structure. The upper level is designed to search for information objects, which we call "hyperequivalences", the medium level – search for "polyequivalences", the lower one – for "monoequivalences".

Monoequivalence is a binomial tuple (an ordered pair) of the form $<Rn(i); Fm(j)>$, where the first position is occupied by the i-th occurrence of a LGF Rn. Each occurrence $Rn(i)$ is provided with its "address" in the Russian part of the database (all occurrences of each form, in both Russian and French parts of the database are indexed). The second position is occupied by the j-th occurrence of the French LGF Fm, functional equivalent to $Rn(i)$ in one of the French translations. The Table I contains a preliminary list of monoequivalence types and examples of their realizations (which are partially taken from comparative Russian-French grammars [28-30]).

TABLE I.   MONOEQUIVALENCE TYPES

Monoequivalence type	Russian sentence	French translation
<Fut-PF; Pr>	Я *вернусь* через 10 минут; Что я ему *скажу*?	Je *reviens* dans 10 minutes; Qu'est ce que je lui *dis*?
<Pres-PF; Pr>	не *пойму*	Je n'y *comprends* rien
<Past-PF(= 'fut'); Pr>	Ну я *пошел*	Eh bien, je *pars*
<Past-PF; Pr>	Он только что *вышел*	Il *sort* à l'instant
<Imperat; Pr>	А ты *помолчи!*	Toi, tu *te tais*!
<Fut-PF; Pr>	Если я его *увижу*, я ему это скажу	Si je le *vois*, je le lui dis
<Fut-PF; Pr>	Если я его увижу, я ему это *скажу*	Si je le vois, je le lui *dis*
<Pr; Imparf>	Он понял, что она его *любит*	Il a compris qu'elle l'*aimait*
<Pst-PF: PqParf>	Он понял, что она его *любила*	Il a compris qu'elle l'*avait aimait*
<Conj; Imparf>	*если бы я знал, я бы сказал*	*si je savais je dirais*
<Fut-PF; FutAnt>	я *закончу* через час	*J'aurai fini* dans une heure
<Fut-PF; PasCom>	Если в понедельник ничего от меня не *получишь*, возвращайся	Si lundi *tu n'as* rien *reçu* de moi, reviens
<Vfin, *что* Past-PF; Vfin+InfPas>	сожалею, *что не сказал*	je regrette *de n'avoir pas dit*
<Vfin *что/как*; Acc.c.Inf>	Я *видел, что* он вернулся	je *l'ai vu revenir*
<Vfin + Inf; Vfin + Inf>	Майор Ковалев *имел обыкновение* каждый день *прохаживаться* по Невскому проспекту	Le major Kovaliov *avait l'habitude de se promener* chaque jour dans l'avenue Nevski.
<Vfin + Inf; Vfin >	Он *начал щупать* рукою, чтобы узнать: не спит ли он?	*il tâta* avec la main, se pinça pour se convaincre qu'il ne dormait pas
<Vfin + DET; Vfin + Inf>	Послушай, голубушка, - *говорил* он *обыкновенно*, встретивши на улице бабу, продававшую манишки	Écoute un peu, ma chère, *avait*-il *coutume de dire* lorsqu'il rencontrait une de ces femmes qui vendent des chemises dans les rues
<Vfin + DET; Vfin + Inf>	Бедный Ковалев *чуть не сошел с ума.*	Le pauvre Kovaliov *faillit devenir fou*.

Polyequivalence is a binomial tuple of the form $<Rn(i); \{Fm(j), Fk(r), ...\}>$, which represents a combination of several monoequivalences with the identical first position ($<Rn(i); Fm(j)>$, $<Rn(i); Fk(r)>$ etc.), i.e. the set of French translations of the given Russian form in the given Russian sentence. So, for the i-th occurrence of the Russian LGF Rn the French form Fm appears as its functional equivalent in the first translation, the form Fk appears in the second French translation, and so on. The index j for the form Fm indicates that the i-th entry of the form Rn corresponds to j-th occurrence of the form Fm in the first French translation, index r denotes that to the i-th occurrence of the Russian LGF Rn corresponds the r-th occurrence of the French form Fk in the second translation, and so on; cf. a polyequivalence type and an example of Russian sentence and two French translations in the Table II.

TABLE II.   POLYEQUIVALENCE TYPE

Polyequivalence type	Russian sentence	French translations
<Conj;{CondPas, VDisp(CondPas) + Inf}>	То есть Иван Яковлевич *хотел бы и того и другого*, но знал, что было совершенно невозможно требовать двух вещей разом, ибо Прасковья Осиповна очень не любила таких прихотей.	*First translation*: ...Ivan Iakovlévitch *se serait volontiers régalé* de café et de pain frais […] *Second translation*: ...Ivan *aurait bien voulu goûter* de l'un et de l'autre […]

Hyperequivalence is a binomial tuple of the form $<Rn; \{F\}>$, which comprises one Russian LGF and a multitude of its functionally equivalent French LGFs which enter into one or more monoequivalences with Rn: ($<Rn(i); Fm(j)>$, $<Rn(i); Fk(r)>$ etc. So, a hyperequivalence represents an aggregation of all the possible types of functionally equivalent fragments of a given Russian LGF that exist in our database. It is important to note that if a new monoequivalence type appears in the database (when a new translation or a new pair of parallel texts enter into the corpus), the content of the corresponding hyperequivalence is automatically updated.

Three-level structure of the database serves to provide the possibility of a cross-identification on all three levels, each of them being source of relevant linguistic data. The database schema we developed allows to store information on the authors of original texts, their translators and on the time of creation of each text. It includes tables for morphological

properties of each word, which are used to retrieve targeted LGFs in original texts and their functionally equivalent fragments (FEFs) in translations. It also includes information about translation discrepancies or "mismatches" between original texts and their translations. The schema of the database has no restrictions concerning language pair (several French translations of a single Russian text); it may be used as well to store data from any polyvariant parallel corpus.

In the process of developing the schema of the database we decided to store information about each pair "source-translation" separately, which allows to preserve the original segmentation of the aligned texts. Based on these pairs we will be able to produce monoequivalences. Polyequivalences are constructed by joining two or more monoequivalences having the same first entity. The advantage of separate storage of each translation pair consists in the fact that it allows to keep analyzed fragments of text as short as possible, which facilitate linking LGFs in Russian text to their FEFs.

4. Information technologies for database creating

On the basis of the database architecture described above a technology of developing the database using the parallel Russian-French texts was set up. The technology includes the following stages:

- selecting multivariant parallel texts (multiple French translations for each single Russian text);

- alignment of the parallel texts with several variants of translation and XML markup for parallel texts (see section 5);

- morphological annotation and lemmatization of the aligned parallel texts;

- including the parallel texts with several variants of translation into the corpus;

- constructing the database of the equivalent lexico-grammatical verbal forms;

- calculating the statistical estimates of these equivalences between Russian and French verbal forms.

At the stage of selecting multivariant parallel texts the existing translations of Russian literary texts into French language are analyzed. For the multivariant corpus only texts that exist in no less than two French versions are selected. Moreover, these translations should be created not earlier than the middle of the 20th century. The earlier French translations, and particularly those created in the 19th century, are obsolete linguistically and, which is even more important, include a great deal of mistakes and abridgements.

Many Russian literary works (by Gogol, Goncharov, Leo Tolstoy, Dostoevsky and others) exist in different modern French versions translated by skilful experts. They exhibit both different approaches to translating fiction and different choices of linguistic means including those relevant to the grammar.

The alignment algorithm provides a separate alignment for each pair of a text and its translation. After that, a bilingual expert checks and corrects the results of alignment, and moves, merges or separates sentences in order to make fragments of the translated text correspond the respective fragments of the original text.

The same pattern is followed for each new translation, and afterwards all the translation pairs are merged automatically into a multivariant array of three or more texts. The resulting texts are saved as an XML file. The technology of alignment and markup is described in the next section of this paper in more detail.

At the stage of the alignment of the parallel texts and XML markup the expert adds a piece of relevant metatextual information to each aligned text. At the stage of the morphological markup the French and Russian texts are automatically marked with a morphological analyzers for these two languages. Each word is tagged by a set of morphological properties. The parallel texts with a morphological markup are saved into an XML file and uploaded to the corpus.

To grant an access to the corpus online the aligned texts with metatextual and morphological markup are made searchable within the multivariant Russian-French parallel corpus. The parallel texts are made available via a search interface that supports the standard search functions of the RNC by lexical and/or grammatical features and their combinations. A subcorpus can also be customized by metatextual attributes, the texts can be searched within a subcorpus; the query result can be sorted and exported.

As we have noted above, the database is built on the basis of the bilingual multivariant aligned texts of the corpus. The main *raison d'être* of this database is the possibility of a bilingual grammatical search for mono- and polyequivalences and calculation of statistical estimates for their types based on of Russian LGFs and their French FEFs.

The methods we use are based on formalized descriptions of each monoequivalence type *<Rn; Fm>* (see Table I). These descriptions result from ascribing to each LGF *Rn* and its respective LGF *Fm* a combination of morphological properties, which have been automatically, assigned to individual Russian and French word forms during the preceeding morphological analysis step. The database is built in iterations. At each iteration one search query is built according to Russian and French LGFs composing a selected monoequivalence type *<Rn; Fm>*. Each search query is executed on bilingual multivariant aligned texts of the corpus.

The retrieved fragments of the multivariant aligned texts are analyzed in order to identify and mark all monoequivalences of the type *<Rn; Fm>* in these fragments. After the search of all types of monoequivalences from the Table I is completed, equivalences that are absent from this table remain unmarked in the corpus. The search for those equivalences is performed using queries that include only grammatical features of the Russian LGFs. At this stage all types of monoequivalences listed in the Table I, which contain the searched Russian LGFs, are excluded. If new types of monoequivalences appear during the analysis of the found

fragments of aligned texts, they are added to the Table I. After completing the search, all marked monoequivalences are automatically grouped into polyequivalences according to their Russian LGFs *Rn(i)*.

After the mono- and polyequivalences' search is over, this database can be used for calculating type-frequencies. Frequency of a monoequivalence type *<Rn; Fm>* (in percents) are is defined by the following formula:

$$Freq(t, n, m) = 100*Q(t, n, m)/Q(t),$$

where *Q(t, n, m)* is the number of monoequivalences of type *<Rn; Fm>* at time *t*; *Q(t)* is the overall number of monoequivalences of all types in the database at time *t*.

The frequency of a polyequivalence type with two FEFs *<Rn; {Fm, Fk}>,* which we will call the second-order type, is calculated according to the following formula:

$$Freq(t, n, m, k) = 100*Q(t, n, m, k)/Q(t),$$

where *Q(t, n, m, k)* is the number of polyequivalences of type *<Rn; {Fm, Fk}>* at time *t*; *Q(t)* is the overall number of second-order polyequivalences of all types in the database at time *t*.

The three- and *n*-order polyequivalences are introduced in a similar manner. Their frequencies are found by analogous formulae with the variable of *t*, and *n*+1 indices in the formula for the *n*-th order of polyequivalences. All the type-frequencies depend on *t*, as the corpus and the database are constantly updated. Consequently, frequencies of all the mono- and polyequivalence types can fluctuate accordingly.

5. Aligning and XML marking

The technology of developing the database described above includes aligning and XML marking processes. The XML markup currently in use allows for building bi- or polylingual corpora with multiple translations of original texts (including the option of multiple translations into the same language).

The parallel texts in the RNC are aligned sentence-by-sentence. A lot of texts kindly offered for the use in the RNC by Adrian Barentsen and included into his ASPAC multilingual corpus are already aligned paragraph-by-paragraph [31]. We have additionally refined this segmentation semi-automatically introducing boundaries between sentences within these pre-defined paragraphs. For the majority of texts the alignment is made exclusively by the RNC team. The original text's sentence segmentation overrides the segmentation in the translation. So, each single element of parallel tagging corresponds to the original sentence, whilst its translated counterpart(s) may well be either a part of a sentence starting with a space or a comma or more than one sentence.

Multiple alignment tools have been used for the RNC parallel corpus. The procedure of alignment consists of two stages: introducing sentence boundaries into texts and alignment in the narrow sense of the word. Some alignment programs do not have an embedded sentence-splitting algorithm (HunAlign by Andras Farkas, http://mokk.bme.hu/resources/hunalign; LeoBilingua by Leonid Brodsky, www.hot.ee/bclogic/) while others enable

sentence-boundaries markup, for example TextAlign (http://www.englishhelp.ru/soft/soft-for-translator/151-textalign.html) or *Parallelnye Texty*, a program developed for the RNC by A. A. Kretov's team in the Voronezh University and used for markup of some English-Russian and German-Russian texts.

The algorithm of breaking a text into sentences is straightforward in programs of both types; it uses the punctuation marks like exclamation mark, quotation mark and full stop without taking into consideration the initial letters, abbreviation, quotation and parenthesis marks, or the rules of direct speech (for problems of using TextAlign in an Ukrainian-Polish parallel corpus see [32]). Segmentation, in programs of both types, can be corrected manually, although the algorithm itself cannot be corrected, and some general mistakes are to be treated each time they occur. In TextAlign, additionally, the automatic sentence-breaking is obligatory, and one cannot escape it by creating a standalone program for this purpose. For LeoBilingua and HunAlign this is optional, and it is possible to elaborate rules of sentence-breaking and change them as the new texts bring new challenges.

The alignment process for all the four tools is automatic with possible manual control. However two possible modes of alignment can be further distinguished – step-by-step (with corrections possible in the course of the simultaneous alignment) and full alignment with post-correction. The first approach is embraced by LeoBilingua and Parallelnye Teksty, while the other is chosen by TextAlign and HunAlign. The last two programs, therefore, call for a re-reading of an already aligned text with correction of the wrongly aligned sentences. While in TextAlign a GUI interface is provided for this (however a single correction calls for re-aligning the whole text), in HunAlign only a manual editing of the output file is possible.

This all said, LeoBilingua and HunAlign seem to be the best choice for the RNC and both are used in it currently, both allowing for user-defined sentence splitting and using statistical mechanisms of alignment. Both have their advantages. While LeoBilingua allows for a slow well-controlled process, with a possibility to split sentences manually in all tricky places and correct possible text misprints in a GUI, sending the results directly into a Unicode XML file, HunAlign aligns the whole text quickly with few mismatches, marked and discerned. The latter is currently used by Ruprecht von Waldenfels in ParaSol, a project close to the tasks of the RNC parallel corpora (earlier aka Regensburg Parallel Corpus, see [33]). However the material of Slavic corpora offers some challenges, including the dictionary problems; as the languages with rich inflection demand including most forms of the paradigm into the dictionaries used in alignment.

In order to deal with some of the issues pointed above, another strategy is currently explored. It involves using a part-of-speech tagger for annotating the corpora in both languages before the alignment step. A part-of-speech tagger TreeTagger [34] is available for both Russian and French languages and provides, beside categorization, sentence-breaking as well as lemmatization. The sentence-breaking is based on linguistic rules and is more elaborated that *ad-hoc* scripts mentioned above. However, it may be used only with alignment tools that

allow for external sentence-breaking. The lemmatization helps to reduce the morphological diversity: each original sentence (made of inflected forms) can be transformed into a sentence made of lemma. The task of aligning sentences of lemmata is easier, since a bilingual dictionary of lemmata is easier to produce. Once aligned, sentences of lemmata can of course be replaced with original, inflected sentences.

While the quality and accuracy of the alignments rely heavily on the quality and richness of the bilingual lexicon, it was found that a simple, word-by-word general lexicon was not available for Russian and French. Fortunately, the HunAlign program may produce, after an alignment task, a bilingual lexicon it built in memory while aligning the sentences.

6. Format and morphological tagging

The parallel texts in the RNC are presented in an XML format where sentences are paired by the <para></para> tag. Each sentence has an attribute indicating the language (this may be changed when bi- or polylingual texts are inaugurated). If a sentence is not translated, an omission is marked by three dashes.

The texts are automatically annotated using the morphological analyzers designed by Yandex search engine. The lexical and grammatical annotation is included into <ana></ana> tag. The tags are not currently disambiguated: however, some Russian texts selected for parallel corpora are already manually disambiguated for the monolingual corpus and may be later included into parallel corpora as well.

There is also a possibility for tagging translation discrepancies, i.e. adding of new text or omitting of the source fragments in the translated text. This kind of discrepancies is not a rare case in translations of fiction. They should be taken into consideration in every serious linguistic study based on parallel texts, as the translation accuracy can never be taken for granted. These discrepancies are also a valid object of study in their own right, concerning the theory of translation as such.

The parallel texts are made available for search online at the www.ruscorpora.ru website. Due to copyright reasons, no text is available for full view; the search results are always presented in the form of separate sentences with minimal context (so-called "snippets"). The following parameters are searchable:

- any combination of lemmata, exact word forms, and morphological tags within a 10-word combination (e. g. "avoir" in the Imparfait tense + Past Participle yields the French Pluperfect);

- names of the author and the translator, language of the original text, language of the translation text.

These parameters are available by selecting a subcorpus for further textual or grammatical search. New software is under development, providing for semiautomated alignment of more than one translation texts. The result should be an XML document with multiple French phrases (marked as belonging to a given translation) corresponding to each Russian phrase of the source text. By the end of 2013, the first multivariant

aligned texts will be included into the Russian-French parallel corpus.

7. Concluding remarks

A user working with a multivariant parallel corpus typically intends to retrieve the occurrences of an original LGF and its FEFs in the translation variants present in the corpus. To solve such tasks a technology is set up for creating a database of binomial tuples <LGF; {FEF}> based on the multivariant corpus. This database should offer the capability to find LGFs and their respective FEFs in parallel texts.

The main difference between this database and the parallel corpus itself is that the former enables the search by LGFs, FEFs and the types of mono- or polyequivalences (see Tables 1 and 3). The corpus can be searched only by the morphological properties of words. The query results for the corpus consist of the aligned text fragments including the requested morphological properties. The query results for the database consist of the LGFs, FEFs, mono-, polyequivalences and their types, as well as the fragments of the parallel texts that include them. Moreover, the database allows for building hyperequivalences based on the retrieved monoequivalences.

While loading new data into the database the expert has to select and to mark up the monoequivalences relevant to the monoequivalence type in question. This selection is personalized in the database by the expert's ID. The morphological properties assigned by him/her are also recorded in the database. Currently the selection of monoequivalences relevant to the type in question is done by one expert only. According to our plans in the future multiple experts should work simultaneously, and the results of their collaboration should be processed using the methods and models of coordinating personal expert knowledge [35, 36].

References

[1] The use of databases in cross-linguistic studies, M. Everaert, S. Musgrave, and A. Dimitriadis, Eds. Belin-New York: Walter de Gruyter GmbH & Co., 2009.

[2] D.O. Dobrovolsky, A.A. Kretov, and S.A. Sharoff, "Korpus parallel'nyx tekstov [Corpus of parallel texts]," Nauchnaja i tekhnicheskaja informacija, Serija 2 Informacionnye processy i sistemy [Scientific and technical information. Ser. 2: Information processes and systems], № 6, pp. 16-27, 2005.

[3] D.O. Dobrovolsky, A.A. Kretov, and S.A. Sharoff, "Korpus parallel'nyx tekstov: arkhitektura i vozmozhnosti ispol'zovanija [Corpus of parallel texts: Architecture and Usage]," in Nacional'nyj korpus russkogo jazyka [Russian National Copus]: 2003-2005. Moscow: Indrik, 2005, pp. 263-296.

[4] E.G.Andreeva and V.B. Kasevich, "Grammatika I leksika (na materiale anglo-russkogo korpusa parallel'nykh tekstov) [Grammar and lexicon in the English-Russian corpus of parallel texts]," in Nacional'nyj korpus russkogo jazyka [Russian National Copus]: 2003-2005. Moscow: Indrik, 2005, pp. 297-307.

[5] D.O. Dobrovolsky, "Korpus parallel'nyx tekstov v issledovanii kul'turno-specifichnoj leksiki [Study of culture-specific lexicon using a corpus of parallel texts]," in Nacional'nyj korpus russkogo jazyka. Novye rezul'taty i perspektivy [Russian

National Copus. New results and prospects.]: 2006-2008. St.Petersburg: Nestor-Istoriya, 2009, pp. 383-401.

[6]   E.B. Kozerenko, "Cognitive Approach to Language Structure Segmentation for Machine Translation Algorithms," in Proceedings of the International Conference on Machine Learning, Models, Technologies and Applications, June, 23-26, 2003. Las Vegas: CSREA Press, 2003, pp. 49-55.

[7]   E.B. Kozerenko, "Linguistic Filters in Statistical Machine Translation Models," Informatika i ee primenenija [Informatics and its Applications], vol. 4, No 2, pp. 83-92, 2010.

[8]   E.B. Kozerenko, "Parallel Texts Alignment Strategies: the Semantic Aspects," Informatika i ee primenenija [Informatics and its Applications], vol. 7, No 1, pp. 82-89, 2013.

[9]   E.B. Kozerenko, "Syntactic Transformations Modelling for Hybrid Machine Translation," in Proceedings of ICAI'11, WORLDCOMP'11, July 18-21, 2011. Las Vegas: CRSEA Press, 2011, pp. 875-881.

[10]   B. Comrie, Aspect. An introduction to the study of verbal aspect and related problems. Cambridge: Cambridge Univ. Press, 1976.

[11]   B. Comrie, Tense. Cambridge: Cambridge Univ. Press, 1985.

[12]   Ö. Dahl, Tense and aspect systems. Oxford: Blackwell, 1985.

[13]   Tense and aspect in the languages of Europe, Dahl Ö., Ed. Belin-New York: Mouton de Gruyter, 2000.

[14]   J.L. Bybee, R.Perkins, and W. Pagliuca, The evolution of grammar: Tense, aspect and modality in the languages of the world. Chicago: University of Chicago Press, 1994.

[15]   Grammatika sovremennogo russkogo literaturnogo jazyka [Grammar of contemporary Russian]. In 2 vols. Moscow, 1954.

[16]   Russkaja grammatika [Russian grammar]. In 2 vols. Shvedova N.Ju., Ed. Moscow, 1980.

[17]   V.V. Vinogradov, Russkij jazyk [Russian language]. Moscow, 2001.

[18]   V.G. Gak, Teoreticheskaja grammatika francuzskogo jazyka [Theoretical grammar of French]. Moscow, 2000.

[19]   Nacional'nyj korpus russkogo jazyka: 2003-2005 [Russian National Corpus: 2003-2005]. Moscow: Indrik, 2005.

[20]   Nacional'nyj korpus russkogo jazyka: 2006-2008 [Russian National Corpus: 2006-2008]. St-Petersbourg: Nestor-Istorija, 2009.

[21]   Korpusnye issledovanija po russkoj grammatike [Corpus studies on Russian grammar]. Moscow: Probel-2000, 2009.

[22]   Lingvistika konstrukcij [Construction linguistics], E.V. Rakhilina, Ed. Moscow: Azbukovnik, 2010.

[23]   A.V. Bondarko et al., Teorija funkcional'noj grammatiki. Temporal'nost'. Modal'nost' [The theory of functional grammar. Temporality. Modality]. Leningrad, 1990.

[24]   E.V. Paducheva, Modal'nost' [Modality] - http://rusgram.ru.

[25]   A. Goldberg, Constructions: a construction grammar approach to argument structure. Chicago: Univ. of Chicago Press, 1995.

[26]   A. Goldberg, Constructions at work. The nature of generalization in grammar. Oxford: Oxford Univ. Press, 2006.

[27]   M. Tomasello, Constructing a language. A Usage-based theory of language acquisition. Cambridge, Mass. and London: Harvard Univ. Press, 2003.

[28]   V.G. Gak, Russkij jazyk v sopostavlenii s francuzskim [Russian language compared to French]. Moscow, 2006.

[29]   V.G. Gak, Sravnitel'naja tipologija francuzskogo i russkogo jazykov [Comparative typology of French and Russian]. Moscow, 2010.

[30]   I.N. Kouznetsova, Grammaire contrastive du français et du russe. Moscow: Nestor Academic Publishers, 2009.

[31]   Information on Amsterdam Slavic Parallel Aligned Corpus - http://   www.uva.nl/over-de-uva/organisatie/medewerkers/content/b/a/ a.a.barentsen/a.a.barentsen.html.

[32]   N. Kotsyba, "The Current State of Work on the Polish-Ukrainian Parallel Corpus (PolUKR)," in Problems of Slavic Lexicography, Proceedings of the International workshop within MONDILEX project, Kiev, 2-4 February 2009 - http://www.domeczek.pl/~natko/papers/ NKotsyba_Kyiv2009.pdf.

[33]   R. v. Waldenfels, "Compiling a parallel corpus of slavic languages. Text strategies, tools and the question of lemmatization in alignment," in Beiträge der Europäischen Slavistischen Linguistik (POLYSLAV), B. Brehmer, V. Zdanova, R. Zimny, Hrsg. München, 2006, pp. 123-138 (available at: http://www-nw.uni-regensburg.de/%7E.war05297. slavistik.sprachlit.uni-regensburg.de/pub/WaldenfelsParallelCorpora 2006.pdf)

[34]   H. Schmid, "Probabilistic Part-of-Speech Tagging Using Decision Trees," in Proceedings of the International Conference on New Methods in Language Processing, D. Jones, Ed. Manchester, UK: UMIST, 1994, pp. 44-49.

[35]   I. Zatsman, "Nestastionarnaja semioticheskaja model' komp'juternogo kodirovanija kontseptov, informacionnykh ob'ektov i denotatov [Time-dependent semiotic model of computer coding of concepts, information objects and denotata]," Informatika i ee primenenija [Informatics and its Applications], vol 3, No. 4, pp. 87–101, 2009 - http://www.ipiran.ru/journal/issues/2009_03_04/Vol3_Issue4.pd f (in Russian).

[36]   I. Zatsman, "Tracing Emerging Meanings by Computer: Semiotic Framework," in Proceedings of the 13th European Conference on Knowledge Management, vol. 2. Reading: Academic Publishing International Limited, 2012, pp. 1298–1307.

SESSION

NATURAL LANGUAGE PROCESSING AND RELATED METHODS

Chair(s)

TBA

A Knowledge-Based Approach to Word Sense Disambiguation

W. Faris and K.H. Cheng

Computer Science Department, University of Houston, Houston, Texas, USA

Abstract – *Lexical ambiguity, words with multiple meanings, are prevalent in all natural languages. In most cases, an individual can determine the meaning of a word in a sentence using available information. Similarly, a natural language processor should use as much information as possible to disambiguate a word's meaning. In this paper, we propose a robust knowledge-based solution to the word sense disambiguation problem. Ambiguous words are resolved using not only the word's part-of-speech, but also the contextual information found in the sentence. We classified ambiguities into two categories: one for words whose multiple meanings may be used as the same part-of-speech and have the same spelling, while the rest fall into the other category. We divided the process of handling ambiguities into two distinct phases. Given a sentence, the first collects all possible meanings of the words, and the second, done distributively based on the ambiguity's category, determines the intended meaning.*

Keywords: Natural Language Processing, Word Sense Disambiguation, Parser, Artificial Intelligence

1 Introduction

An ideal artificial intelligence program is one that could communicate with humans using a natural language. One problem for these programs is disambiguating the proper meaning for words that have multiple meanings. For instance, in "John seals the envelope", the word 'seal' means an action, while it is an animal in "John feeds the seal", a man-made object for making an imprint image in "The seal of the company is on the contract", and a device used to join two objects in "The seal is leaking oxygen". For humans, the ambiguities are resolved by using conversational context and the position of the word in a sentence. A reasonable solution to resolve ambiguities in a program is to mirror the approach used by humans. However, as the knowledge base of the program grows, the number of multiple meaning words and the number of ways to use them in a language also increases, possibly requiring the existing solution to be changed. In this paper, we describe how our program handles the word sense disambiguation (wsd) problem in a robust way. Our solution is knowledge based, depending on both the knowledge referred to by the context of the sentence and the grammatical knowledge of the natural language. We separate the ambiguities into two distinct categories. The first category involves knowledge that have the same base form name and all conflicting knowledge may be used as the same part-of-speech (*pos*). For instance, 'pry' could be for inquiring too much or alternatively, using force to open an object, but both meanings are used as an action verb. All other situations belong to the second category. It includes cases where conflicting knowledge are used in different *pos*, irrespective of having the same or different base form names. For example, 'desert' could be the noun of an arid region or the verb to abandon. It also includes cases where conflicting knowledge are used in the same *pos*, but they have different base form names. For example, 'painting' represents a noun or the action verb in progressive form that could be stemmed to 'paint', which in-turn could be mistaken as a noun by a parser.

Regardless of the category, the disambiguation process is accomplished in two distinct phases. The first phase involves recognizing and collecting the several knowledge objects that may be identified by the word. This is influenced to some extent by how words are stemmed, as well as how our program maintains knowledge. Different kinds of knowledge are maintained by different classes, and each knowledge object is indexed according to its name. The second phase of the disambiguation process is responsible for selecting the correct sense of the ambiguous word. In this phase, each category of ambiguity is resolved at a different stage of handling a sentence. Resolving ambiguity for the first category can only be resolved by using context available from other parts of the given sentence, and so has to be handled at a stage after the sentence has been parsed successfully. For the other category, the disambiguation is accomplished at the stage of parsing the given sentence by using the grammar and the classifications of the various knowledge objects for the ambiguous word. Occasionally, words may have several meanings with conflicting senses in each category of ambiguity, as in 'seal'. For these words, the respective ambiguities are resolved appropriately and successively in each stage.

The *wsd* problem has been heavily studied and summarized in surveys [1, 2]. Algorithms for the *wsd* problem can be either supervised or unsupervised. Supervised algorithms such as [3, 4] rely on machine learning techniques and need a trained corpus. One advantage of supervised algorithms is that the accuracy is relatively high in the designated knowledge domain. The disadvantage to purely supervised systems is the problem of

data sparseness and the lack of sufficient ambiguous situations in standard texts. In other words, if the program is to work on all domains, it is unlikely that the corpus is large enough to be trained on all ambiguous terms. Unsupervised algorithms use lexical and worldly knowledge, like those in [5, 6]. Although they generally are not as accurate as supervised algorithms, they are not limited to the scope of the corpus. Some algorithms such as those in [7, 8] try to take advantage of the strengths of supervised and unsupervised methods by combining both. Regardless of the algorithms used, most disambiguation solutions use statistical methods in resolving ambiguity. In an attempt to follow a more natural approach similar to how humans disambiguate, our solution does not perform any statistical calculation and uses context information available in the given sentence. Specifically, it makes use of the word's *pos* in the given sentence, selectional restrictions [9] that use semantic restrictions placed on a word's sense, neighboring words, syntactic features [10] that utilize the word's grammatical role in the sentence, and the domain knowledge acquired by our system. Another difference of our solution from most methods is that we resolve ambiguity in a distributed manner rather than in a single disambiguation module. This allows disambiguation to occur once enough knowledge is available, meaning some ambiguous terms can be determined earlier and may potentially be used to help disambiguate other ambiguous terms if multiple ambiguous terms exist in the sentence. Finally, our solution to the *wsd* problem is robust since it depends on acquired knowledge to make sense of the context information available in the sentence rather than being designed for specific ambiguous words. In addition, as knowledge continues to be gained by the system, the same context information may then be used to resolve unforeseen ambiguities.

We have proposed A Learning Program System (ALPS) [11] with the goal of learning knowledge learned by a human. The initial focus of ALPS had been on identifying the composition of the various kinds of knowledge and their relationships with one another. For example, basic capabilities such as creating a new category and adding objects, attributes, and properties to a category are provided. Two useful knowledge components of categories, important in solving the disambiguation problem, have also been developed: hierarchy [12] and definition [13]. In addition, new knowledge classes have recently been developed to acquire the grammar of a natural language, which allows ALPS to communicate with humans in English [14]. Two main stages in the process of handling a sentence are parsing and understanding. The parsing stage is accomplished in two steps: a syntax step and a semantic step. The syntax step analyzes the words of the given sentence, stemming words if necessary and identifying a word or group of words as a term, where each term corresponds to at least one piece of knowledge. To accommodate all possible ambiguities, the syntax step has to identify and keep all possible knowledge alternatives

identified for each term. The semantic step interprets the high level intent of a grammar term by recognizing the correct knowledge object to represent a particular *pos* and relating the found knowledge to others. At the end of this step, an appropriate thought is produced at the sentence level. The understanding stage uses the knowledge it has learned about the contents of the thought to decide the thought's sensibility. For example, the action knowledge identified by the action verb in a sentence is defined to make use of the action knowledge class. This class can identify any action known by the program given its name and understand the relations among the other knowledge objects in the thought, such as the specific knowledge objects for the subject and the direct object. Ambiguity that cannot be resolved by the semantic step of the parsing stage, those with the same name and same *pos*, are resolved in the understanding stage of the process.

The remainder of this paper is organized as follows. Section 2 describes briefly some background information about ALPS which includes how the program handles knowledge, the major components of the grammar of a natural language, and the two stages in processing a given sentence. Section 3 provides examples of how knowledge is identified, i.e. how different types of ambiguous terms are treated to retrieve all the possible knowledge that they could represent. Section 4 discusses how each category of ambiguity is resolved. Finally, section 5 discusses some complications associated with our solution and concludes the paper.

2   Background Information on ALPS

Different kinds of knowledge are implemented in ALPS by different classes of objects: such as category, action, and description. In order to reduce the amount of knowledge stored within ALPS, each piece of knowledge is identifiable by a unique name, its base form name. For example, the word 'eat' is the base word for the words 'eats', 'eaten', 'ate', and 'eating'. As a result, only a single action knowledge object representing 'eat' exists in the knowledge base of the program. However, knowledge of different kinds may share the same name. For examples, the word 'meter' identifies both an object of a category and a unit of measurement for the concept of distance; while 'paint' may refer to the liquid substance or the spreading action. They are represented internally by ALPS using knowledge objects of the appropriate kind. A search on the given name will yield a list of all knowledge objects, but a search on a specific kind using the name will only yield a single knowledge object of that kind. In addition, one name may have several meanings within a single kind of knowledge. For instance, there are several distinct pieces of knowledge within the action 'receive': a means of communication such as a receiving a phone call, the obtaining of tangible object such as a gift, etc. A search with that name will yield one single knowledge object but with several internal knowledge objects containing their meanings. Hierarchy and definition

are two knowledge components of categories whose functionalities are used extensively in solving the disambiguation problem. Hierarchy allows both generalization and containment relationships among categories to be specified and stored (e.g., Houston is a city, which implies that it is a location to live, whereas IBM is a company, which implies that it is a place to work). Definition specifies the necessary and sufficient condition that objects of a category must satisfy, and therefore may be used for classification. The ability to infer categories and to classify knowledge allows our program to disambiguate multiple meanings for ambiguous words. It also allows our solution to be applicable to any new knowledge that can be classified or inferred as the appropriate category required by a specific meaning.

Before ALPS can communicate with humans using a natural language, the grammar of the language must first be learned. The grammar terms (or *pos*) are composed of up to five components: structure, kind, role, control and semantic rule. The structure of a grammar term defines the grammatical format of the term using either a sequence or an alternative set of terms. A term may have multiple kinds, which are subsets that may share the same structure but must have different roles. The role of a grammar term defines its purpose. The idea of the control is to notify the parser which role from within a sequence of terms is responsible for organizing the identified knowledge during the parsing process. A rule specifies a condition to be satisfied by either the grammar term or its structure. Grammar rules are used to either guide the parser towards the correct grammar structure or restrict certain terms from being accepted. The use of rules in a grammar can enhance the scope of the grammar while maintaining a manageable size for parsing. They can also be used to aid in identifying the proper knowledge of a multiple meaning word by specifying conditions that must be met for a specific sense of a word, as will be seen in this paper.

The grammar learned by ALPS includes not only grammar terms and their associated components but also morphology rules. Morphology rules expand the set of legal words by specifying different forms and meanings of a word. Two categories of morphology rules are used in English: regular and irregular. A regular rule specifies how an affix is added to a base word in order to express a property. For example, the suffix –ing is added to action verbs to indicate a progressive tense such as 'eating'. However, the same affix can mean different properties depending on the part-of-speech of the word. For example, the suffix –s is used to indicate a plural number property when applied to nouns, but singular for verbs. An irregular rule is used for words whose transformation from the base word into another form does not adhere to any consistent rules. Our program is taught both types of rules to transform a word to and from its base form, and each rule identifiable by the property it represents. Regular rules are defined by conditions and an action to be carried out if those conditions are met, e.g., if a word ends in –x, -s, -ch, or –sh,

it can be made plural by adding –es. On the other hand, specialized lexicons are provided to our program to map a word between its base and irregular forms. For instance, a plural lexicon contains a set of words and their irregular plural forms such as goose/geese, ox/oxen, and radius/radii.

Morphology rules are applied during the syntax step to convert (stem) a word to its base form and recognize any properties it may reflect. The syntax step will first attempt to identify knowledge based on the un-stemmed words. Identification is not limited to a single word, but also recognizes knowledge represented by multiple word term (*mwt*) such as 'least common multiple' and 'surface area'. The stemming process is done in two parts, and the exact order of applying the two parts is irrelevant for English. In one part, the multiple lexicons are checked to see if the word is in an irregular form. Whether or not it is in an irregular form, the other part applies each regular transformation rule to the original word. When a morphology rule is applied to a word, the newly formed word is checked against the knowledge base as well as a dictionary to validate its legitimacy as a true word. For the dictionary, we direct ALPS to the words file that comes standard on the UNIX operating system. At the end of the syntax step, each term is represented by a unit that contains the original word, its base(s), identified properties, and all the identified knowledge for that term. This sequence of word units is then passed to the semantic step. Note that for a term unknown to our program, a "dummy" knowledge object is created.

The semantic step deals with recognizing the various *pos* and relationships among them in order to produce an appropriate thought reflecting the meaning of the sentence. It is based on a depth-first top-down template parser, and its execution consists of processing the grammar structures in two major manners: top-down and bottom-up. Starting from the sentence, the top-down processing traverses the grammar structures and calls upon each grammar term's internal parser, unique by its structure, to determine if the current grammar term matches the next word unit. One subtype of the alternative grammatical structure is the alternative knowledge structure, which defines a set of ALPS specific knowledge kinds that can categorize the given grammar term. For examples, the grammar term action verb is taught to use action, adjective to use description, and noun to use category, object, concept, or an unknown knowledge of those kinds. The internal parser for the alternative knowledge structure works by iterating through the list of knowledge kinds to determine if the knowledge in the word unit is a member of that kind. Consider parsing a word unit that contains the simple knowledge of the individual John. Assuming John is taught as a human, the alternative knowledge structure for noun will iterate through the list of knowledge kinds until it recognizes John as an object of the human category. The top-down processing also checks the grammar rules to validate the results. After all the subparts of a grammar term have been correctly identified, the bottom-up

processing organizes all these identified subparts into a meaningful result and returns that to a higher-level term. By the time the semantic step finishes, the result is a thought that reflects what the sentence wants to express, and all identified knowledge are stored as the appropriate parts of this thought.

The thought produced by the semantic step can now be interpreted by the understanding stage of ALPS. The produced thought may provide a high-level intention of the given sentence such as it may declare a fact, inform an action performed by a specific object, or pose a question. However, the exact meaning of the sentence is based on the relationships among the various knowledge objects referred to within the given sentence. For instances, a declaration using a forms-of-be verb may declare the hierarchical relationship between two categories or the state/value of an aspect of an object [15]. In addition, the declaration of an action may imply the occurrence of an event or activity, and may also imply several sub-actions that are intuitive in nature but not explicit in the sentence [16]. For instance, the act of buying includes at least two transactions: the transfer of money and the transfer of the product. Furthermore, this stage uses knowledge learned about the objects identified by the sentence to determine the sensibility of the given sentence. A declarative sentence is determined to be non-sensible if it contradicts with what the system concludes. For example, stating that 3 is a factor of 7 will be considered as not sensible by ALPS. Finally, it is also responsible to disambiguate the meaning of a certain term if the term is known to have many interpretations, specifically, the term for the action verb.

3   Knowledge Collection of Words

The primary objective of the syntax step in the parsing stage is to separate the given sentence into terms and associate each term with the knowledge representing the term. If a term has multiple meanings, the proper meaning may be identified by its *pos* or context information. Since neither has been established during the syntax step, ambiguities cannot be resolved in this step and all alternatives have to be stored and passed onto later steps. Note that for an ambiguous word that is part of a *mwt*, resolving its ambiguity may be avoided once the knowledge representing the *mwt* has been identified. For example, the words 'bank' and 'factor' are not ambiguous once the knowledge 'Bank of America' and 'highest common factor' have been identified in conjunction with neighboring words. However, the ambiguities of these words are not resolved if they do not form a *mwt* with their neighboring words.

Recall that knowledge in ALPS is indexed by its name in its base form; and each name may reflect one or more knowledge objects. However, since words within a sentence may not be in their base forms, locating all the potential knowledge referred to by a term requires stemming words using morphology rules. In addition, there exist cases where a word may be both in its base form as well as the transformed form of another, e.g., 'painting', 'drawing',

and 'trimming'. The term 'painting' could be either the art work or the progressive form of the action 'paint'. For this reason, even though a term may already identify a piece of knowledge in its original form, morphology rules still need to be applied in the event that it can be stemmed to reveal other knowledge objects. In this case, 'painting' can be reduced to 'paint', which, as shown earlier, represents two additional meanings. As a result, the final word unit that is created for this term contains three alternative definitions: the art work 'painting', the substance 'paint', and the action 'paint'. Note that although it may be clear to the human that the liquid paint is not a plausible option, this is not distinguishable at the syntax step, which returns all knowledge under that name.

Arriving at the base form of a word can occur in multiple ways. It is possible in English to have a transformed form of one word be identical to an irregular form of another. For example, the term 'lives' is both the plural form of the object 'life' as well as the singular third-person form of the action 'live'. As a result, both regular and irregular morphology rules have to be applied in order to recognize both knowledge objects. Note that which kind of morphology rules is applied first is unimportant, but both have to be applied to the original word. In our program, the lexicon of irregular word forms is searched first. Here, it will find the base 'life' and retrieve the knowledge representation from ALPS. Then when one of the regular rules is applied, the suffix –s is removed to reveal the action 'live'. By the end of the syntax step, the original word 'lives' will contain the two possible knowledge objects.

Finally, only one regular morphology rule is used per word; in other words, a stemmed word will not be stemmed a second time using a different stemming rule. This avoids issues with words like 'paintings' that can be stemmed to form the word 'painting', which can then be stemmed to get to the word 'paint'. Clearly, 'paintings' is not a form of the word 'paint'. The same principle applies for determining which rules can be applied. For instance, the definition for producing a plural is to add either an –s or –es to the end of a word. However, when asked to stem a word like 'cares', both versions could produce the legal words 'care' and 'car'. For this reason, regular morphology rules are applied in an order specified in the grammar. With this rule, the affix –s is removed first. If it produces a valid word, the stemming is complete. Otherwise, the affix –es is removed from the word. All words that are stemmed by regular words are tested against a dictionary for legitimacy to avoid creating false words, e.g., 'dea' will not be accepted although it is created from 'dead' by removing –d based on the past tense morphology rule.

As pointed out earlier, ambiguous words such as 'pry' that have multiple meanings of the same knowledge kind will have their various definitions stored within the same knowledge object. This single knowledge object is returned when searched using the ambiguous word. On the other hand, an ambiguous word that represents knowledge of different kinds, like 'paint', will return a list of knowledge

objects when it is searched. Occasionally, a term may have multiple meanings that can be defined by multiple knowledge objects of different kinds, some of which may be single knowledge objects with multiple meanings. For example, searching for the different senses of 'lie' will return a single knowledge representing a lie along with a knowledge object representing the multiple action definitions: the giving of a false statement or being in a reclined position. Consequently, the syntax step identifies all senses of the word and includes them in the word unit. For instance, if the original sentence was "Mark lies about painting the patio.", the word unit for the ambiguous term 'lie' would include the multiple knowledge representations. The word unit for 'painting' would include the singular knowledge of the category painting, the knowledge of the action paint (marked as progressive), and the knowledge of the category paint (marked as singular). The full list of word units will then be sent to the semantic step of the parser, where the ambiguity is resolved based on the specific kinds of knowledge capable of fulfilling the particular *pos* in the grammar structure.

4 Resolving Ambiguity

As mentioned earlier, there are two categories of ambiguities and each is resolved at a different point in the process. In this section, we will first explain how to resolve ambiguities that belong to the second category since it occurs first in the process, specifically during the semantic step of the parsing stage. After that, we will cover how ambiguities of the first category are resolved, which occurs during the sentence understanding stage.

Recall that grammar terms like noun and action verb can be defined in the language as alternative knowledge structures. By defining a grammar term by the kinds of knowledge that can fulfill it, ambiguous knowledge with different *pos* can be resolved by the internal parser of the alternative knowledge structure. This is accomplished by identifying and isolating the knowledge objects in the word unit that belong to the classification specified by the *pos* definition. If exactly one meaning meets this criterion, the ambiguity is resolved. For example, assume a word unit contains the three knowledge objects resulting from stemming the term 'painting'. If the parser were trying to identify an action verb in the sentence 'The artist is painting a new painting', it would narrow down the list to just the action knowledge. The other two would fail since they are both category objects. However, when the internal parser tries to identify the noun in the same sentence, it will isolate two possible word senses, 'painting' and 'paint'. These are knowledge of the same *pos* but with different names. In situations like this, both options are semantically correct. The artist could be working on a painting or applying another layer of paint. However, since one knowledge object matches the original spelling, this is most likely the intended meaning, while the other is derived from the stemming of the original word. As a result, the alternative

knowledge structure will select the knowledge object for 'painting'.

Another situation of ambiguity may arise from the following circumstance. The learning of the grammar of a natural language has the flexibility to define a grammar term using a less restrictive alternative and rely on grammatical rules to prevent the exceptions from being accepted. For example, in noun phrase, pronouns and proper nouns cannot be preceded by an article, while a determiner is required if the noun is singular such as 'the dog' and 'a pavilion'. Assume noun phrase is taught as a sequence of an optional determiner followed by a noun to account for these possibilities, and a grammatical rule is used to decide the necessity of a determiner. Now consider the following two grammar terms: infinitive and prepositional phrase. An infinitive uses 'to' but requires the base form of an action verb such as 'to achieve' and 'to pry', while a prepositional phrase with the preposition 'to' contains a noun phrase such as 'to Houston', 'to the mall', and 'to me'. There is no problem in recognizing a phrase as an infinitive instead of a prepositional phrase if the involved word can only assume one *pos*, such as the verb 'achieve'. However, for ambiguous words such as 'store' that may be used as both a noun and an action verb, the phrase 'to store' could be recognized as an infinitive or a prepositional phrase without the optional determiner. The presence of the grammatical rule that requires a determiner for singular nouns can prevent the phrase from being fulfilled as a prepositional phrase but instead recognized correctly as an infinitive. In [17], we describe how grammatical rules such as this can be softened to tolerate grammatical incorrectness when ambiguity does not exist, but are necessary to resolve ambiguities, if found. For example, the grammatically incorrect phrase 'to dog' may be accepted to mean 'to a dog' or 'to the dog'; but 'to store' will not be parsed to create the meaning of 'to a store' and must mean 'to store an item'.

Next, consider ambiguities of the first category, i.e., knowledge that have the same base form name and all conflicting knowledge may be used as the same *pos*, such as the two meanings for the action 'pry'. Since the same name of the knowledge is used for the same *pos*, the responsibility to identify the intended meaning is no longer that of the semantic step of the parsing stage, but belongs to the understanding stage. This stage resolves ambiguity by combining three knowledge sources used in unsupervised algorithms: selectional restrictions, syntactic features, and the classification of domain knowledge. The classification of an object can be inferred using hierarchical relationships such as an object of a sub-category is also an object of an ancestor category. In [16], we resolve this category of ambiguity for action verbs by examining the category of the direct object only. For example, with the verb 'receive', one could physically receive an item, such as a gift, or receive a message, like a phone call or an email. Therefore, when the direct object is a physical object, the chosen definition of 'receive' would be an increase in the set of possessions. On

the other hand, if the direct object was a communication medium, the definition of 'receive' would be a gain of the knowledge conveyed in the communicated message.

Besides the direct object, the multiple definitions of an action may be differentiated by the category of other objects involved in the action such as the subject. We have now extended the disambiguation to include other context information provided by the given sentence, and process them in the following order: the classification of the direct object, the category of the subject, the presence of and knowledge within each prepositional phrase attached to the verb. As described, the category of the direct object is first used to disambiguate. If there is no direct object in the given sentence, then a similar approach can be used to disambiguate using the category of the subject. If the subject in the sentence belongs to a category that is only capable of carrying out one sense of the action but not the others, then the correct meaning may be identified. For example, the correct meaning of the action in "the tree lies on the forest floor" can be determined since a tree (a non-human physical object) cannot tell lies but can be in a resting position. However, if the subject can perform more than one meaning of an action, using the category of the subject will fail to disambiguate. For instance a human can both tell a lie and lie down. For such cases, we then use the information obtained from prepositional phrases attached to the verb. Each meaning of an action may be taught with a list of prepositions that are unique to that definition. For example, telling a lie can be directional ("John lies to his coworkers") or topical ("John lies about his coworkers") using the preposition 'to' and 'about', respectively. On the other hand, the act of lying down may be in relation to another object ("John lies on the bed"). When the preposition used in a phrase matches one of these prepositions of the action, the correct meaning of the action is uniquely identified. However, the category of the object of the preposition may also be needed to disambiguate the proper meaning for the following two reasons. First, many prepositions such as 'at' and 'on' are used to specify both location and time, regardless of the verb being used. For example, looking only at the phrases 'on the desk' and 'on Sunday', it is obvious they represent a location and a time, respectively. Second, the same preposition may be used with multiple meanings of an action. Take for instance the sentences "John moves to Houston" and "John moves to IBM." A location (Houston) as the object of the preposition signifies that the move is a physical change in residence, while a move to a company (IBM), signifies a change in employment. Consequently, the category information of the object of the preposition must be used to disambiguate the various meanings. Furthermore, information about the category may be used to recognize non-sensible sentences. For example, John cannot lie to a measurement ("John lies to 4 meters"). It may also be used to guess the correct preposition when an improper one is used and there is no ambiguity (ex., the queen to England). Incorrect preposition usage is a common grammatical mistake among non-native speakers. Note that situations can occur where a sentence alone will not provide enough information to disambiguate a verb sense, such as "John lies." Situations like this will require context from the current discourse, and will be used when our system is extended to handle multiple sentences.

A similar approach may also be implemented to resolve noun ambiguity in the future. For example, consider the multiple meanings of the noun 'seal'. The correct meaning of the word will rely on the context of the sentence, specifically the verb and either the predicate or subject of the sentence. If the sentence contains an action and the ambiguous noun is the actor, the type of action can determine which word sense is more appropriate. For instance, verbs that represent a movement, such as 'eat', 'swim', 'yell', cannot be applied to inanimate objects such as the seal that joins two objects. Similarly, animals cannot perform verbs that represent a connection, such as 'closed', 'leaks', 'binds', 'applied'. If the sentence contains a linking verb and the ambiguous noun is the subject or predicate, the determining factor would be the predicate or subject of the sentence, respectively. The predicate of a sentence defines a characteristic or state of being of the subject. Based on learned knowledge of higher-level classes, the more logical word sense can be used. For example, in the sentence "The seal was hungry", it is obvious that an inanimate object cannot feel hunger. However, just like ambiguous verbs, there are times in which not enough information can be provided in the sentence. "The seal is worn out" could reflect a tired sea mammal or a bond that has lost its adhesiveness. In situations like this, the program has the following options to resolve the ambiguity. It could look for context provided in the discourse in the hopes that prior statements specify the type of object. Another alternative would be to use other forms of input, such as visual images, if applicable. Finally, it could take the same approach as their human counterparts and simply ask for clarification.

5   Conclusion

We have described in this paper a two phase word-sense disambiguation solution. The first phase is responsible for identifying and locating all the possible knowledge objects corresponding to each term in the given sentence. It uses morphology rules taught to the system to convert words into their base forms, and happens at the first step of parsing a sentence, namely, the syntax step. The second phase is responsible for resolving the ambiguity among all possibilities to correctly identify the intended meaning. Due to the nature of the ambiguities, we classify ambiguities into two categories. One category may be resolved based on the grammar's requirement that a certain *pos* be at a specific place of the given sentence, and hence can be resolved during the parsing stage of a sentence. The other category is resolved during the understanding of the thought, which uses the context information available from the rest of the sentence. Our solution depends on the capability of the program to infer the categories of various objects. This makes our solution applicable to a huge amount of

knowledge and robust in its ability to disambiguate unforeseen cases as more knowledge is acquired.

Resolving an ambiguous word based on the word's *pos* is possible when the parse tree is unambiguous. However, problems may arise when multiple parse trees can be formed due to the absence of an optional term and the presence of a term with an ambiguous *pos*. For instance, when an action verb accepts indirect objects but one is not provided; instead a time adverb, which can serve as a noun or an adverb, follows the direct object. An example sentence would be "John bought the TV yesterday." Since 'bought' can accept indirect objects, the parser will tag the TV as the indirect object and yesterday, serving as a noun, as the direct object. Consequently, the sentence is understood incorrectly as the object yesterday being purchased by John for the TV.

When ambiguous terms were resolved using the categories of various objects, relying on the classification does present the problem of determining their most appropriate category. For example, it would be ideal to say that when defining the act of eating, humans and animals can only eat edible objects, such as fruits, meats, plants and vegetables. This definition of 'eat' implies an act done for nourishment. However, an object may be considered edible by one animal, but not by another. For instance, wood can be eaten by termites but is not considered edible by humans. Bamboo, a staple of the Panda diet, is not eaten by cows, although both eat plants. In addition, animals and babies are always capable of eating non-edible items like paper or small toys. In situations like this, a natural category may cover all sensible objects but not plausible ones. Another problem with categorizing is that a category may not even naturally exist. For example, the act of telling a lie should be towards a category of objects that can receive and interpret information. This category could include individuals, a collective group like the public, entities like the government, etc. Defining a category for all the possibilities appears to be very difficult. These issues may be tackled through teachings that allow for exceptions, multiple category associations, and specialized categories to replace generic teachings when available. Both problems will be addressed in future versions of ALPS.

6 References

[1] R. Navigli. "Word sense disambiguation: A survey"; ACM Computer Survey, 41, 2, Article 10, 2009.

[2] X. Zhou & H. Han. "Survey of Word Sense Disambiguation Approaches"; Proceedings of the 18th International FLAIRS Conference, 2005.

[3] G. Escudero, L. M`arquez, & G. Rigau. "Naive Bayes and exemplar-based approaches to word sense disambiguation revisited"; Proceedings of the 14th European Conference on Artificial Intelligence, 421–425, 2006.

[4] Y. K. Lee & H.T. Ng. "An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation"; Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, 41–48, 2002.

[5] O.Y. Kwong. "Word Sense Disambiguation with an Integrated Lexical Resources"; Proceedings of the NAACL WordNet and Other Lexical Resources Workshop, 2001.

[6] R. Mihalcea & D. Moldovan. "An Iterative Approach to Word Sense Disambiguation"; Proceedings of the 13th International FLAIRS Conference, 219-223, 2000.

[7] E. Agirre, et al. "Combining supervised and unsupervised lexical knowledge methods for word sense disambiguation"; Computer and the Humanities, 34, 103-108, 2000.

[8] R. Mihalcea. "Co-training and self-training for word sense disambiguation"; Proceedings of the 8th Conference on Computational Natural Language Learning, 33–40, 2004.

[9] M. Stevenson, & Y. Wilks. "The Interaction of Knowledge Sources in Word Sense Disambiguation"; Computational Linguistics, 27, 3, 321 – 349, 2001.

[10] P. Hastings, et al. "Inferring the meaning of verbs from context"; Proceedings of the 20th Annual Conference of the Cognitive Science Society, 1142-1147, 1998.

[11] K. Cheng. "An Object-Oriented Approach to Machine Learning"; Proceedings of the International Conference on Artificial Intelligence, 487-492, 2000.

[12] K. Cheng. "The Representation and Inferences of Hierarchies"; Proceedings of IASTED International Conference on Advances in Computer Science and Technology, 269-273, 2006.

[13] K. Cheng. "Representing Definitions and Its Associated Knowledge in a Learning Program"; Proceedings of the International Conference on Artificial Intelligence, 71-77, 2007.

[14] W. Faris & K. Cheng. "An Object-Oriented Approach in Representing the English Grammar and Parsing"; Proceedings of the International Conference on Artificial Intelligence, 325-331, 2008.

[15] W. Faris & K. Cheng. "Understanding and Executing a Declarative Sentence involving a forms-of-be Verb"; Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 1695-1700, 2009.

[16] E. Ahn, W. Faris, & K. Cheng. "Recognizing the Effects caused by an Action in a Declarative Sentence"; Proceedings of the International Conference on Artificial Intelligence, 149-155, 2009.

[17] W. Faris & K. Cheng. "A Grammatical-Error Tolerant Parser", Proceedings of the International Conference on Artificial Intelligence, 711-717, 2012.

Computer-based Method for Association Response in Autonomous Conversation

Eriko Yoshimura[1]**, Misako Imono**[2]**, Seiji Tsuchiya**[1] **and Hirokazu Watabe**[1]

[1] Dept. of Intelligent Information Engineering & Sciences, Faculty of Science and Engineering
Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2] Dept. of Knowledge Engineering & Computer Sciences, Graduate School of Engineering,
Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *In this paper, the authors propose a method that incorporates mechanisms for handling ambiguity in speech and the ability of humans to create associations, and for formulating replies based on rule base knowledge and common knowledge, that go beyond the level that can be achieved using only conventional natural language processing and vast repositories of sample patterns. In this paper, the authors propose a method for associated replies elicited from information relating to the place as an example of how the common knowledge and associative ability described earlier are applied.*

Keywords: Computer interface human factors, Knowledge engineering, Knowledge representation, Natural languages

1   Introduction

While it is easy for human beings to provide autonomous responses during conversations, it is extremely difficult for machines to do the same. For the most part, machines are capable of performing predetermined actions, but are less able to respond to varying conditions. In addition to the lack of predetermined rules, autonomous responses require the ability to handle new expressions and thoughts. However, such an ability requires an understanding of natural language, making autonomous judgments concerning a reply, and then formulating an expression for the reply. Thus, mechanisms need to be developed that can provide machines with a basic understanding of grammar and conversational rules, as well as the ability to handle ambiguous language, and to create associations like humans do.

One area of study that explores methods for processing replies that are specialized to the rules of a conversation is task-type conversation. This research guides the conversation toward fulfilling the target task and is extremely effective when used for carrying out clerical work. As a result, conversational systems have been developed for numerous specific purposes, with systems already in practical use that can grasp the state of the conversation as a transition diagram and develop effective conversational rules for accurately obtaining the information necessary for the task—without

allowing the conversation to wander to other topics. However, such systems are unable to handle any issues other than the target task or to understand words that have not been registered. Thus, they are exceptionally poor when it comes to the development of issues such as chatting or small talk.

One system that was developed specifically for chatting is Eliza [1], which is an interactive system. Eliza is a counselor system developed with the aim of eliciting responses from the conversation partner. To generate a sentence in reply, Eliza searches for examples of keywords and then returns questions in response to those keywords. The system also partially stores the contents of previous utterances and incorporates vocabulary from those sentences in its reply pattern. Because of this, Eliza is able to handle any type of conversation with users and is able to give the impression that it is producing autonomous replies. However, because the conversational content is not evaluated, the program is unable to comprehend what the conversational partner has said, nor can it be assured of generating an appropriate response.

Eliza's ability to provide the illusion of conversation with humans triggered the development of subsequent chat systems such as A.L.I.C.E., Reactor, Verbot, and Julia. These systems have been improved to the point where they "learn" responses from the conversational partner and, by being continuously placed in an environment in which they chat with humans, have come to possess a continually growing repository of response patterns.

A large volume of research in interactive conversation that is not task-driven is being carried out (see, for example, [2] [3] and the interactive voice system described in Ref. [4]). These and other similar systems acquire conversational rules through inductive learning using genetic algorithms based on an enormous repository of interactive conversation samples. As a result, they demonstrate better performance than Eliza-type systems.

These systems generate responses by imitating conversations contained in the abovementioned repositories and are very good at chat-type conversations. However, when a system

simply learns previously used patterns, it can only model conversations based on those patterns and is unable to formulate responses that reflect the volition of the machine itself. Thus, while it may appear capable of spontaneous chatting, there are limits to the autonomous responses that can be generated based on pattern learning alone.

To solve these problems, research has been conducted in natural language processing utilizing statistical processing [5] [6]. This can involve, for example, using the frequency of occurrence of words to analyze spoken sentences from the standpoint of both grammar and meaning, in addition to increasing the scale of knowledge bases containing sample patterns that are used to produce natural language. These efforts are aimed at statistically grasping the intention and meaning of a sentence from the standpoint of the number of word occurrences. As such, these systems provide the foundation for understanding meaning and responding autonomously on the basis of rule base knowledge and common knowledge.

In this paper, the authors propose a method that incorporates mechanisms for handling ambiguity in speech and the ability of humans to create associations, and for formulating replies based on rule base knowledge and common knowledge, that go beyond the level that can be achieved using only conventional natural language processing and vast repositories of sample patterns.

2 Related technologies

2.1 Concept associations

In order to create appropriate associations, a machine needs to be made to understand the strength of the association between two words in terms of meaning. To quantitatively evaluate the strength of the association between two words, we use a calculation of the degree of association [8] using a concept base [7].

A concept base [7] is a knowledge base comprising words (concepts) that has been mechanically auto-constructed from multiple Japanese dictionaries, newspapers and other sources, along with a glossary of words expressing those meanings (attributes) and concepts (in the case of the above reference, approximately 90,000). Sets of concepts and attributes are weighted so as to express the importance of the set. A given concept A is defined as a paired set consisting of an attribute a_i that expresses the meaning characteristic of the concept and a weight w_i that expresses the importance of attribute a_i in expressing concept A. The attribute a_i is called a primary attribute of A, while an attribute for which a_i is the concept is called a secondary attribute of A.

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_N, w_N)\}$$

Calculation of the degree of association [8], which is the strength of the association between two concepts, is evaluated quantitatively using the concept base. The two concepts have been expanded as far as the secondary attributes, and the calculation is carried out using the weight to determine the optimum combination of primary attributes and to evaluate the weight of matching attributes. The degree of association is a real number between of 0 to 1, with a higher value indicating a stronger association. Table 1 shows an example in which calculation of the degree of association was carried out on concept A and concept B.

Table 1. Example of calculation of the degree of association

Concept A	Concept B	Value for degree of association
car	automobile	0.488
car	flower	0.008

2.2 Understanding of common knowledge

When human beings are given ambiguous information, in most cases they are able to decipher it appropriately, proceed to appropriate conversation, and take action. This is because humans accumulate knowledge concerning language over many years of experience and establish what we call "common knowledge" relating to the concepts that form the base of this acquired knowledge. However, in order to obtain human-like conversation from machines, the machines must be provided with the capability to understand matters that are common knowledge. This requires constructing a system that will elicit common-knowledge recollections with strong word associations from multiple common-knowledge standpoints. In this paper, the authors describe a system that focuses on common-knowledge standpoints, particularly on places and feelings.

As a standpoint for common knowledge in terms of place, we used the place judgment system [9]. In this system, the machine recalls a person or object in a place, or an event that happened at the place, from words that call to mind the place (referred to as "place words"). An association is then created between the person, objects, or events at that location, and the place words, which are then registered in a knowledge base containing the place words. In the case of a person or object, the associated place word is called a place subject word, while in the case of an event, it is called a place object word. Then, using the concept base and the NTT Thesaurus [10], the system is able to utilize words that are not registered in the knowledge base. Table 2 shows an example of this system in use.

Table 2. Example of use of the place judgment system

Input	Place word?	Place subject word	Place object word
milk	×	—	—
bookstore	○	customer, store manager, book, magazine, etc.	sell, buy, attend to customers, etc.

The feeling judgment system [11] is used as the standpoint for the common knowledge called "feeling". This system obtains a word pertaining to a feeling or sensation that people are able to recall on the basis of common knowledge, to which characteristics have been assigned, in relation to a noun. "Feeling" refers to a sensation resulting from a stimulation of one of the five senses (sight, hearing, smell, taste, and touch). A total of 98 such "feeling words" that pertain to the five senses were selected from among all adjectives and adjective verbs. Then associations were built in the knowledge base between the nouns and the feelings that characterize them. Additionally, the concept base and NTT Thesaurus are used to recall sensations for words that are not in the knowledge base itself. Table 3 shows an example of the use of the feeling judgment system.

Table 3. Example of use of the feeling judgment system

Concept	Feeling words
apple	red, sweet, round
sunset	dazzling, red, beautiful

2.3   Semantic comprehension system

In order to carry out autonomous conversation, it is necessary to understand the meaning of conversational sentences spoken by the user. Therefore, information frames are created that make distinctions regarding the roles of words in the sentence. In this paper, we used a technique that involves dividing input sentences into frames of "what", "when", "where", "who", "whom" "why", or "how" (6W1H) + verb and storing them accordingly. This is a method of organizing that uses case markers and grammar rules to comprehend sentences, and which we refer to hereinafter as the semantic comprehension system [12].

Input sentences (simple sentences) that do not contain complex concepts and compound sentences are divided into information frames of 6W1H + declinable word (verb) and stored accordingly. Figure 1 shows an example.

Input text

I make a big snowman at the ski area on my winter holiday.

Semantic comprehension system

Information frames

Who	What	When	Where	Why	How	Whom	verb
I	big snowman	winter holiday	ski area				make

Figure 1. Working example of the semantic comprehension system

3   Association responses

Once a sentence has been understood, the system generates a reply based on common knowledge and the ability to create associations. In this paper, the authors propose a method for associated replies elicited from information relating to the place as an example of how the common knowledge and associative ability described earlier are applied.

3.1   Place associations

We propose a method in which the system associates actions of the speaker (what the speaker did) and replies based on words relating to the location (place words) in the sentence spoken by the user. Figure 2 shows an example of this process.

USER: I went to the art museum
SYSTEM: Did you see the painting?

Figure 2. Example of place association

In order to create an association with the actions of the speaker (what the speaker did), a sentence is targeted in which the "what" frame is blank at the time when the sentence is analyzed by the semantic comprehension system. The target sentence also includes a place word or words, and the "verb" frame exists in the place verb knowledge base that will be described later.

When the conversation involves the actions of a user in relation to a place, the input sentence needs to include the information that "the place will be visited", and that information needs to be the primary content. Given that, we stored verbs such as "go" and "visit", which primarily indicate that the speaker went to a location, in the location verb knowledge base. Table 4 shows a place verb knowledge base.

Table 4. Place verb knowledge base (13 words)

go, visit (a place), visit (a person), depart, go out, take (someone), take off, come back, return home, return, approach, enter, come

First, the system obtains the place subject words and place object words for the place word in the input sentence. In the case of "I went to the art museum", "art museum" serves as the place word. Table 5 shows an example of this process.

Table 5. Example of obtaining place subject words and place object words pertaining to "art museum"

Place word	Place subject word	Place object word
art museum	sculpture, painting, curator, handicraft, work of art, print	exhibit, view, appreciate, look

Next, words that pertain to people, such as "curator" and "station personnel", are eliminated from the place subject words. In this step, the NTT Thesaurus is used to eliminate words that are included in the "people" node.

Furthermore, since it is difficult for words that are frequently used in the passive voice to be regarded as representing the actions of a speaker, using a large-scale case frame system automatically constructed from the Internet (hereinafter referred to as the large-scale case frame system), words that are frequently used in passive-voice expressions were eliminated from the place object words.

The abovementioned large-scale case frame system is organized on the basis of declinable words and nouns that are related to those words, according to how the declinable word is used. It was constructed automatically from approximately 500,000,000 Japanese texts on the Internet, and currently comprises approximately 90,000 declinable words. The modification structure and other information about sentences used routinely on the Internet can be looked up from the noun or declinable word that is provided.

Table 6. Frequency of use in terms of grammatical structure

Input		Output	
Noun	Declinable word	Grammatical structure	No. of occurrences
painting	exhibit	*ga*-marked (subjective)	376
		o-marked (objective)	267
		no-marked (possessive)	47
		kara-marked (from)	2

Tables 6 and 7 show examples of a declinable word ("exhibit") and a noun ("painting") that have been looked up using this method. We know that when these two words are used in combination they are most frequently used in the nominative case (*ga*-marked words) and passive voice. Table

6 indicates that they are most used in *ga*-marked words, while Table 7 shows that, in terms of *ga*-marked words, they are most frequently used in the passive voice. Because this combination is often used in the passive voice, it is deleted.

Table 7. Frequency of voice for the phrase "paintings are exhibited"

Input			Output
Noun	Declinable word	Grammatical position	Voice and number of occurrences
painting	exhibit	*ga*-marked	active voice: 28 passive voice: 348 causative: 0

After the above method has been carried out, the degree of association is calculated with respect to place subject words, and only those combinations of place object words with the highest degree of association are retained. Additionally, the degree of association between place words in the input sentence and all place subject words is calculated. The combinations of place subject words and place object words with the highest degree of association to the place words are then selected.

Using the pair of a noun and declinable word that has been selected (in the example provided here, "painting" and "look"), a response sentence to the question sentence is created. With respect to the conversion method used to turn this into a conversational sentence, we use templates that were created on the basis of large-scale case frames and rules formulated especially for the Japanese language, but specific examples have been omitted in this paper.

3.2 Adjective associations

With adjective associations, a reply is created that expresses "agreement" in relation to the "what" frame in the input sentence. Figure 3 shows an example of this process.

> USER: I was viewing the roses in the garden.
> SYSTEM: Roses are really beautiful.

Figure 3. Example of adjective association

In order to carry out a conversation expressing routine agreement, it is necessary to obtain epithets that can be generally associated based on the phrasing used in the "what" frame.

Thus, epithets are obtained using the feeling judgment system. Moreover, the large-scale case frame system is used in a supplementary manner, and the top 30 epithets are obtained. Table 8 shows candidate epithets that could be associated with the example word "rose".

Table 8. Candidate epithets associated with "rose"

	Associations from the feeling judgment system	Associations from the case frame system
rose	beautiful, red	like, pretty, love, etc.

The degree of association between the epithets obtained and the words in the "what" frame is then calculated, and the epithet that has the highest degree of association is determined. In the example presented here, the set of "rose–beautiful" indicated the highest degree of association, and was thus selected. The association response sentence is then created using the noun and the selected epithet set. Specific examples concerning the conversion method used to turn this into conversational sentences have been omitted.

3.3   Topic-changing associations

In the abovementioned association methods, we explained how identifying associated actions that could be predicted from the input sentence could be used to create a response. In this section, we will describe the creation of responses that introduce a different topic, albeit related to the content of the input sentence. Figure 4 shows an example of a topic-changing association response.

USER: I saw a panda in the zoo.
SYSTEM: Did you see the lion in there?

Figure 4. Example of a topic-changing association response

In this example, despite the fact that the user has conveyed the information that he or she saw a panda, the system provides a response relating to a lion. This is not a direct response to what the user said, but it is an associated topic. If the system simply receives information and responds directly to it, as was described above, conversation tends to get bogged down and it is difficult to expand the interaction into new topics. Thus, topic-changing associations are carried out with the aim of adding movement to the conversation and developing it into a new topic.

In this study, input sentences are assumed to contain a place word and a "what" frame. As an example, let us suppose that the user presents the input sentence, "I played poker at a casino" (shown in Figure 5). The word "poker" in the "what" frame is excluded from the list of place subject words for the place word "casino" in the input sentence, which included "roulette, coin, baccarat, poker, slot", from which unnecessary words have already been excluded. This makes it possible to leave as candidates, words and phrases that are not the same as those in the "what" frame of the input sentence, but which still have an association with the place word of the input sentence.

The degree of association between the group of remaining place subject words and the place word "casino" is then calculated, and the word with the highest degree of association is obtained. In the example, the set of "casino–roulette" demonstrated the strongest degree of association. Additionally, the degree of association between the word "roulette" that was obtained and the place object word is also calculated, and the declinable word with the highest degree of association is obtained. In this case, the set of "roulette–play" demonstrated the highest degree of association. As a result, the set of "roulette–play" was selected.

A response sentence is then created using the selected noun and declinable word set. Specific examples concerning the conversion method used to turn this into conversational sentences have been omitted.
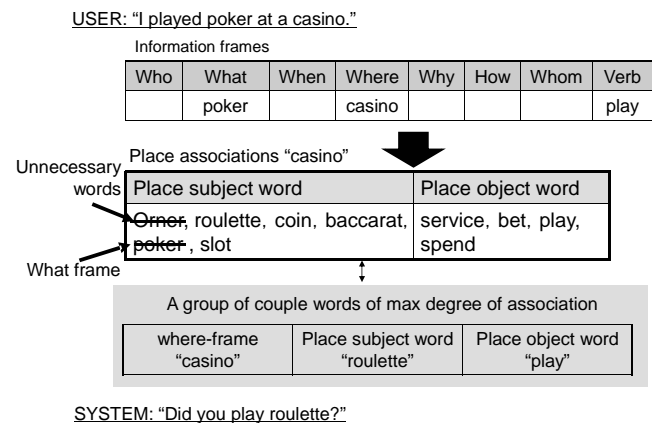
Figure 5. Example of a topic-changing association

4   Evaluation

We evaluated the association-based response method to determine whether the response sentences were truly appropriate. For the assessment data, 100 conversational sentences (that were neither questions nor orders), which included nouns related to place were taken from junior high school English reference books, and the response sentences output by the proposed method were evaluated via visual examination conducted by three people who were not involved in creating the system.

During the evaluation, the response sentences created were classified as either "appropriate" or "not appropriate". The evaluation criterion was that if two or more of the three evaluators considered the respective response to be appropriate, then the response was judged acceptable; if one of three evaluators considered it to be appropriate, then it was judged as moderately acceptable; and if all three evaluators considered it to be inappropriate, then it was judged as unacceptable.

Figure 6 shows the results when sentences were evaluated for each aspect of place association, adjective association and topic-changing association. If we take the combined percentage scores for acceptable and moderately acceptable as our degree of accuracy, then the accuracy for place association was 80%, the accuracy for adjective association was 74%, and the accuracy for topic-changing association was 71%.
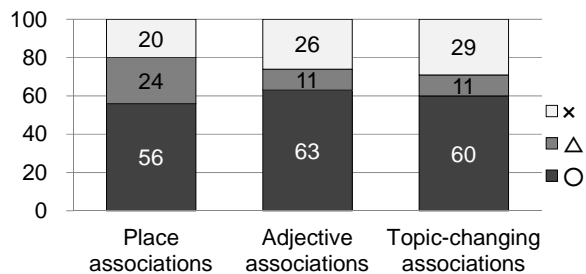
Figure 6. Evaluation results for association responses

5   Conclusion

In this paper, the authors proposed a method of association responses for machine conversation in which a greater number of responses are provided via associations based on the content of the conversation. The results of verification of the system constructed on the basis of the proposed method indicated that it was an effective technique, having attained high scores for accuracy, specifically 80% for place associations, 74% for adjective associations, and 71% for topic-changing associations.

It is believed that creating response sentences using the proposed method can facilitate conversation, and that these responses can provide the machine with common knowledge and the ability to comprehend the content of the conversation. This, in turn, could be used to promote the use of the method.

References

[1]   J.Weizenbaum, : ELIZA ― A Computer Program For the Study of Natural Language Communication Between Man and Machine, *Communications of the Association For Computing Machinery*, Vol. 9, No. 1, pp. 36–45 (1965)

[2]   R.Tokuhisa, R.Terashima,: An Analysis of 'Distinctive' Utterances in Non-task-oriented Conversational Dialogue, *Transactions of the Japanese Society for Artificial Intelligenc*e AI22, pp.425-435(2007)

[3]   M.Inaba, N.Isomura, F.Toriumi, K.Ishii,: Evaluation method of Non-task-oriented dialogue system using Semantic Network, *IPSJ SIG Technical Report*, ICS2009(16), pp.29-34(2009)

[4]   Y.Kimura, K.Araki, Y.Momouchi, K.Tochinai,: Spoken Dialogue Processing Method Using Inductive Learning with Genetic Algorithm, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*s, J84-D-II(9), pp.2079-2091(2001)

[5]   D.Kawahara, S.Kurohashi,: Case Frame Compilation from the Web using High-Performance Computing, *Information Processing Society of Japan Special Interest Group of Natural Language Processing* 2006(1), pp.67-73(2006)

[6]   Maedche, A. and Staab, S.: Ontology learning for the Semantic Web, *Intelligent Systems, IEEE*, Vol. 16, No. 2, pp. 72–79 (2001)

[7]   N.Okumura, S.Tsuchiya, H.Watabe, T.Kawaoka,: A Construction of Large-scale Concept-base for Calculation of Degree of Association between Concepts, *Journal of natural language processing* 14(5), pp.41-64(2007)

[8]   H.Watabe, N.Okumura, T.Kawaoka,: The Method of Measuring the Degree of Association between Concepts using Attributes of the Concepts and Coincidence Information, *Journal of natural language processing* 13(1), pp.53-74(2006)

[9]   J.Sugimoto, H.Watabe, T.Kawaoka,: Constructing a Commonsense Place Judgment System Which Uses a Concept Base, *Information Processing Society of Japan Special Interest Group of Natural Language Processing* 2003(4), pp.81-88(2003)

[10]   S.Ikehara, M.Miyazaki, S. Shirai, A.Yokoo, H.Nakaiwa, K. Ogura, Y.Ooyama, Y.Hayashi,: Goi-Taikei A Japanese Lexicion, *Iwanami Shoten Publisher* (1997)

[11]   S.Tsuchiya, K.Kojima, H.Watabe,: Unknown Word Processing Method for the Common Sense Judgement System, *Transactions of the Japanese Society for Artificial Intelligence: AI 17*, pp.667-675(2002)

[12]   Y.Shinohara, H.Watabe, T.Kawaoka,: A Conversation Semantic Understanding Method Using the Commonsense Judgment System, *Information Processing Society of Japan Special Interest Group of Natural Language Processing* 2003(4), pp.89-96(2003)

Entropy-based Word Sense Disambiguation for Semantic Annotation system

Haifeng Sun, Jianxin Liao, Jingyu Wang, Qi Qi, Tong Xu
State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing, China

Abstract - *To make the text retrieval and index more accurate, several systems were proposed to annotate the semantic of key words using Wikipedia as a catalog. In this paper, we discuss a model by using human senses to express information, and use it to annotate most name entities. For this purpose, an Entropy-based Semantic Annotation (ESA) system is implemented to recognize and annotate name entities in text. A novel Word Sense Disambiguation (WSD) algorithm is designed to disambiguate the ambiguous name entities, and to improve the performance when the relatedness between name entity and context is weak. Moreover, an advanced Viterbi algorithm is employed to optimize the overall entity assignment. In our experiments, two manually-annotated datasets are used to evaluate the ESA system, which is better than Tagme approaches in performance with 79.61% of F-measure. The time complexity of ESA system is O(n), which ensure the ESA system's quickly response.*

Keywords: Wikipedia; Word Sense Disambiguation; Semantic Annotation; Entropy; Viterbi algorithm

1   Introduction

Data types and scale of human society is growing in an amazing speed which is caused by the emerging services such as cloud computing, internet of things and social network, the era of big data has come. Text data is the main presenting forms, how to recognize and retrieve the content of text data better has attracted much attention. In order to analyze unstructured or semi-structure textual data more accurately, many studies in the information retrieval and semantic web areas have proposed to link the name entities of text to an extension of semantic information, which makes it possible for other tools to understand and retrieval the content. This solution is named as semantic annotation. Its key idea is to identify the name entities from textual data and annotate them with un-ambiguous entities drawn from a catalog. There are many catalogs to choose from, but different catalogs are chosen by different approaches. An important catalog is WordNet, which groups English words with structured sets. It

offers a good structure with low coverage. Some system such

as [1] used Wordnet to Disambiguate. As other important catalog, FrameNet database offers large number of semantic frames which is a schematic representation of a situation involving various participants, and other conceptual roles. Recently, Wikipedia has become more and more widely used as a catalog for text semantic annotation. It offers better tradeoff between structure and coverage than WordNet. It consists of 3 million English pages. Each page has the structured natural language content where some key words have been annotated manually. Semantic annotation based information derived from Wikipedia is used in many articles [2-7]. In this paper, we adopt the Wikipedia as the catalog.

Some annotation systems employ knowledge-based Word Sense Disambiguation (WSD) algorithms [2, 3] derived from Wikipedia. For example, Milne and Written's system [2] uses the unambiguous terms to disambiguate the ambiguous terms by calculating the ambiguous terms senses' relatedness to unambiguous. This algorithm needs the text to offer enough unambiguous terms with a similar topic, which limits the scope of application. Recently Ferragina and Scaiella proposes Tagme system [3] which considers improving the performance by using both unambiguous and ambiguous terms through unified formula which an average relatedness score between two terms is gotten through. This algorithm solves the problem of short text with few unambiguous terms. Both the two systems disambiguate terms mainly by comparing each sense of the term to other terms through calculating the relatedness score, it makes the relatedness bias emergence in the process of disambiguation. Relatedness bias means the term will be more likely to choose the sense with higher relatedness score, even though the term is weakly correlate with the main topic of text or the right sense is implied by writer.

In this sense, a WSD algorithm which can solve the relatedness bias is a valuable help in annotation process. Besides this, annotation system should be designed fast, because in those texts data may be retrieved at query and thus cannot be pre-proceeded. For this purpose, we have designed and implemented an Entropy-based Semantic Annotation (ESA) system which annotates name entities of input text to corresponding Wikipedia articles fast and accurately. The system is divided into two parts, name entity recognition and word sense disambiguation. The name entity recognition is

used to recognize the name entities of input text. The word sense disambiguation part employs entropy theory for WSD algorithm to consider the factor that the degree of the effect from the context to current name entity sense choice. According to the result, adjust the parameters of the disambiguate progress to obtain the right sense.

In summary, the main contributions of this paper are as following: A new Entropy-based WSD algorithm is designed to disambiguate the ambiguous name entities, mainly improved the performance when little relatedness between the name entity and context. We observe name entities as a chain and every sense as a state, and use an advanced Viterbi algorithm to improve efficiency.

Experiments show that ESA system's F-measure is up to 79.61%, with the pair of 80.03% precision rate and 79.21% recall rate which is better than [2, 3]. The time complexity of ESA system is linear in the number of processed name entities' senses. Thus verifying the proposed system is appropriate for annotated name entities in plain text.

2   Relate work

Using Wikipedia as a catalog for annotating semantic of the terms develops in a few years. Several systems are designed and implemented to do this task automatically. The first system using Wikipedia in this way is Mihalcea and Csomai's Wikify system [4], which is separated into two steps to work, automatic keyword extraction and word sense disambiguation. The first, Mihalcea and Csomai use several methods to extract keywords, the most accurate approach employs the link probability to do this task. The link probability is obtained by counting the number of documents where the term is already selected as a keyword divided by the total number of documents where the term appeared. Then, disambiguate the terms extracted by prior phase. Wikify employs two algorithm to disambiguate terms, the best approach is to extract features and its surrounding words, and compares this to the training data from Wikipedia. The performance of Wikify can match the test data with a precision of 93% and a recall of 83%. However, This method need to parse the entire Wikipedia, which requires enormous preprocessing effort.

Milne and Witten's system [2] employed a machine-learning approach that yielded considerable improvements than Wikify. A relatedness score *relatedness(a,b)* between two senses *a* and *b* is proposed in this paper. Milne and Witten's system uses unambiguous terms as context to disambiguate ambiguous ones by a classifier using commonness, relatedness and context quality features.

Kulkarni et al. [5] introduces a collective method to score an annotation by two terms. They formulate the trade-off between local term-to-entity compatibility and pairwise

coherence. The formula is NP-hard, hence they propose an approximation to solve the problem.

The Tagme system [3] is designed by Ferragina and Scaiella to annotate terms in short fragment. In order to solve the problem that text with little unambiguous terms, Tagme system uses all possible senses of term to disambiguate other terms by the average relatedness of prior term's senses to the latter one's sense. The performance of Tagme is better than previous systems with an F-measure of about 78%, with the pair of 78.41% precision and 77.48% recall.

Besides these system, some systems employ Wikipedia to disambiguate word sense, all of them achieve a good performance. Turdakov et al. [7] observes the nature language as multiple interleave chain consisted by related terms, they use an expanded Hidden Markov Model to do WSD task. Nadeau et al. [8] uses a learning algorithm to disambiguate word sense, which designs local kernels and global kernels based on the kernel based approach, and uses a linear combination to combine local and globe information. Last year, Pinheiro et al. [9] introduce a new method to disambiguate word sense by a common sense knowledge base and Wikipedia articles. It defines a new measure of evaluate sematic relatedness between concepts. It considers the inference ability of the concepts expressed in the common sense dataset, extends the ability to identify the meaning of the algorithm. ALA$_{TK}$ [10] is proposed a supervised method based on the Topic Kernel function for annotating the schema labels with Wikipedia. In this system, ALA$_{TK}$ uses SVM learning system based on the features of domain knowledge which are trained from Wikipedia to disambiguate word senses.

3   The motivation

As we all know, language is a main tool to spread information. Polysemy is a significant language phenomenon, and it appears widely in every language. To disambiguate the ambiguities terms, much disambiguated systems have been implemented. To our knowledge, previous works mainly used the relatedness on semantic from terms which will be annotated by system to the context to evaluate every sense of terms to be the appropriate sense's probability with a numeric value. Then use the machine learning algorithm or other algorithms to disambiguate terms. This makes relatedness bias happens, where the term may have little relatedness with context and have a little more relatedness value than other senses, the term tends to select the most relatedness sense with previous system even though the term should be annotated with the other sense. For example, in the sentence "*Cristiano Ronaldo hits the headlines when he crashed his Ferrari*" which is mentioned in [7], three terms need to be annotated: *Cristiano Ronaldo*, *headline* and *Ferrari*. The term *Ferrari* is an ambiguous one and has at least two meanings: (a) a brand of car; (b) an Italian football player. Most WSD algorithms tend to use football player to annotate Ferrari instead of car
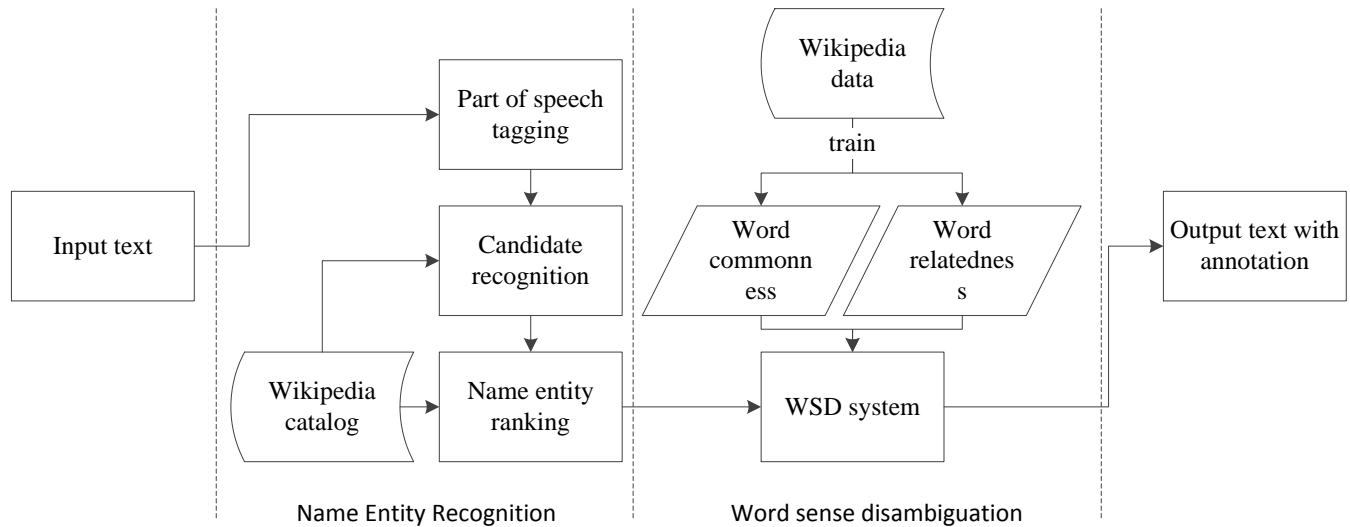
Fig. 1. The architecture of the system to automatically semantic annotate with Wikipedia

brand, because the football player have more relatedness to Cristiano Ronaldo. In fact, the football player Ferrari has little relatedness with Cristiano Ronaldo, people won't talk about them in the same time. So Cristiano Ronaldo has little effect to disambiguate the term Ferrari. In this time, reducing the relatedness effect is important to disambiguate the term Ferrari, and annotating it with most general sense which is a brand of car.

It is a problem how to select the right sense by these features. There may be little relatedness between name entity and surround name entities. In this situation, we often tend to use the word common sense information to decide the name entity's intended sense. While there may be much relationship between name entity and context, we will tend to use the words relatedness of context to decide the name entity's sense. For detecting the degree of relationship between name entity and surround name entities, entropy theory is employs to evaluate it.

4 Text annotation

This section describes a new approach to recognize the name entities occur in plain text, and identify for each name entity with an appropriate link to a Wikipedia article automatically. The requirement is to add links for the name entities in a document, which will allow reader or machine to understand and retrieval the content of the text accurately and conveniently. Automatic Semantic annotation system implies solutions for the two main tasks performed when adding links to an anchor: (1) name entity recognition, and (2) word sense disambiguation.

The first task consists of identifying those words and phrases that are considered appearance in document at hand. These typically include technical terms, named entities, new terminology, as well as other concepts, which are appearance as Wikipedia articles. For instance, in the sentence that

"*Venice's most famous painter comes to Rome*", *Venice*, *painter* and *Roma* are the name entities which could be annotated with corresponding Wikipedia articles. This task is identified with the problem of name entity recognition, which targets in an input natural language text. The second task consists of finding the correct Wikipedia article that should be linked to a candidate anchor. The anchor can be usually linked to more than one Wikipedia article, so we need to face the problem of word sense disambiguation. The correct interpretation of the name entity may depend on the context where it occurs.

A system is developed and implemented to solve the above problem in two steps, as illustrated in Figure 1. To begin, the input text is passed to the part of speech tagging process in order to extract the noun from text by syntactic information. Using the Wikipedia catalog, pick the name entities which are existed in Wikipedia catalog out. The Name entity ranking part is used to assign the candidates which are a set of words or just a phase. The Name entity is annotated by the WSD system with the right Wikipedia article. Wikipedia data is trained to word commonness data and word relatedness data. They are then used to disambiguate texts in WSD system by the WSD algorithm. In the following two sections, we show how the system works to support the process of name entity recognition and disambiguating the word sense.

4.1 Name entity recognition

Since the motivation of annotation is for machine to index, retrieve and do some other nature language process automatically, our objective is to label the most name entities, which is similar to name entity recognition. Consider the previous work on name entity recognition [11], there are three main methods according to the learning method. They are supervised learning, semi-supervised learning and unsupervised learning. Supervised learning methods techniques include Hidden Markov (HMM) [12], Decision

Trees [13], Maximum Entropy Models (ME) [14], Support Vector Machines (SVM) [15], and Conditional Random Fields (CRFs) [16], using features such as syntactic features, syntactic patterns, and others. The main technique for Semi-supervised learning is called "bootstrapping" and involves a small degree of supervision, such as a set of seeds, for starting the learning process [17]. At last, the typical approach in unsupervised learning is clustering. For example, in O. Etzioni et al. (2005) [18], Pointwise Mutual Information Retrieval is used as a feature to assess that a named entity can be classified under a given type.

For ESA system, an important rule is that candidate name entities should be linked to Wikipedia article, so the system must recognize the name entities which are contained in Wikipedia articles. Based on these, ESA uses the supervised learning methods to recognize name entities by utilizing the Wikipedia data and the syntactic features. To recognize most name entities, we use the syntactic information to recognize nouns in text first, and then try to recognize name entities contained in Wikipedia from the recognized nouns. It is apparent that the name entities may be separate words or phrases, which produce a problem to choose separate words as candidates or to choose phrases as candidates. For example, there is a fragment "the machine learning", where the words machine, learning and the phrase machine learning are all contained in Wikipedia, so a ranking algorithm is needed to choose the right candidates. We divided the name entities recognition algorithm into three steps, namely: (1) part of speech tagging; (2) candidate recognition; (3) name entities ranking.

The part of speech tagging step analyzes the part of speech of the sentences from the input texts through the induction process, in this paper, we use the Stanford University tagging process. The candidate recognition step parses the words where is annotated with noun by previous step and extracts all possible n-grams that also present in Wikipedia. The ranking step uses the same method suggested in [4] to choose the right candidates.

4.2 Word sense disambiguation

In what follows, a (text) anchor for a Wikipedia page p is the text used in another Wikipedia page to point to p. We denote by $pg(a)$ the set of all Wikipedia pages linked by a. We consider two features for disambiguation which are used in [4, 6]. These features consist of commonness and relatedness. The commonness is the prior probability. It respects the number of times each concept used for each word. We use $pr(p/a)$ to denote the occurrence probability of an anchor a points to $p \in pg(a)$. Because of polysemy and variant names, we shouldn't only use the commonness, but also use the context information. The relatedness is the surround name entities which influence the concepts selected. Assume that there are two different Wikipedia pages which are p_1 and p_2 point to the anchor a, and the context is more familiar to p_1
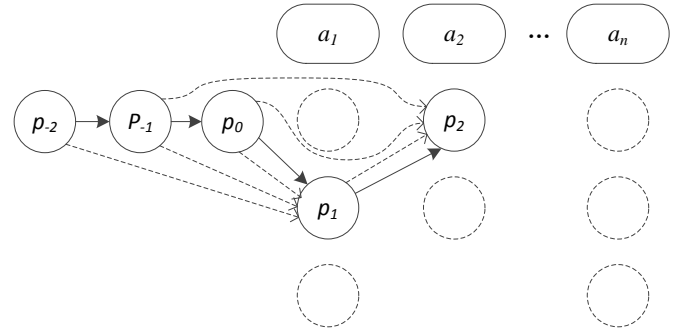
Fig. 2. The model of advanced four-gram Viterbi algorithm with p_{-2}, p_{-1}, p_0 as start node

than p_2. This means the context suggest the reader to use the relatedness information of the context to select the right sense, we will tend to select p_1 to point to the anchor a. As if the relatedness between context and p_1 is almost equally to the relatedness between context and p_2, which means the relatedness information of the context is useless for reader to select the right sense. In this case, we will select the most probably sense by common sense information.

There is a set of Wikipedia pages $p_a \in pg(a)$ which could point to an anchor a. The $rel(p_a, p_b)$, which the relatedness of the surrounding anchor's sense p_b to the Wikipedia page p_a, could be looked as the probability of predicting from the context which is limited as one anchor instead of many anchors in this part to the Wikipedia page p_a. The relatedness between the two Wikipedia pages p_a and p_b is computed as suggested in [2] as:

$$
\begin{aligned}
&rel\left(p_a, p_b\right) \\
&= \frac{\log(\max(|in(p_a), in(p_b)|)) - \log(|\,in(p_a) \cap in(p_b)\,|)}{\log(W) - \log\left(\min\left(|in(p_a), in(p_b)|\right)\right)}
\end{aligned} \quad (1)
$$

Where $in(p)$ is the set of Wikipedia pages pointing to page p and W is the number of pages in Wikipedia. We use the entropy which is the key measure of uncertain in information theory to measure the uncertain of predicting the set of Wikipedia pages pointed to the anchor. In order to make the relatedness to be used as probability, we use $Z_{pb}(a)$ to normalize the score of relatedness.

$$
Z_{p_b}(a) = \sum_{p_i \in pg(a)} rel\left(p_i, p_b\right) \quad (2)
$$

$$
H_{p_b}(a) = - \sum_{p_i \in pg(a)} \frac{rel\left(p_i, p_b\right)}{Z_{p_b}(a)} \log \frac{rel\left(p_i, p_b\right)}{Z_{p_b}(a)} \quad (3)
$$

In entropy theory, the higher score of entropy indicate the higher of uncertain. The higher of uncertain indicate it is useless for context to choose the right page p_i to point to anchor. So, the higher score of entropy indicate the less useful for context to choose the right page p_i to point to anchor. At

this time, we will tend to use the commonsense knowledge to judge the right sense. So, there are different levels to ultra the context in different situation.

Let A_t be the set of all anchors occurring in an input text T, ESA system tries to disambiguate each anchor $a \in A_t$ by computing a score for each possible sense p_a. We will use the surrounding name entities to disambiguate a. Given that $b \in A_t$ may have many senses we compute this vote between each sense p_b of the anchor b and the sense p_a which we wish to associate to the anchor a. Computing the $vote_{pb}(p_a)$ from each sense p_b to vote the sense p_a, we need to calculate $H_{pb}(a)$ with the previously explained rules. Then we use the formula as follow to calculate the $vote_{pb}(p_a)$:

$$vote_{p_b}(p_a) = \frac{\lambda}{H_{p_b}(a)} \frac{rel(p_a, p_b)}{Z_{p_b}(a)} \quad (4)$$

We notice the formula uses the ratio of λ which is a constant obtained by training data and $H_{pb}(a)$ as the weight of $rel(p_a, p_b)/Z_{pb}(a)$. By this way, value of $vote_{pb}(p_a)$ can be significantly increased by lower value of $H_{pb}(a)$. If the value of $H_{pb}(a)$ is closed to 0, the ratio of λ and $H_{pb}(a)$ will be very great, so we set the $H_{pb}(a)=0.01$ when $H_{pb}(a)<0.01$. For calculating a probability $P(p_a/a)$ of the sense p_a to be annotated to anchor a, all of the anchors detected surrounding anchor a and the commonness information need to be considered. Euclidean distance is used to combine these two features, $P(p_a/a)$ is calculated as:

$$p(p_a|a) \propto \sqrt{pr(p_a|a)^2 + \left(\frac{\sum_{i \in A_t} vote_{p_i}(p_a)}{|A_t|-1}\right)^2} ?(i \neq a) \quad (5)$$

Using the value of $P(p_a/a)$ to rank the senses of anchor, the most probability sense will be annotated to anchor. Moreover, since every anchor's selection will affect the choice of other anchor, we need to choose the best combination which has the highest sum of every anchor's score instead of local optima. The objective function is as follow (the follow formulas are the same):

$$\max f = \sum_{i \in A_t} P(p_i | i), (p_i \in pg(i)) \quad (6)$$

$$\max f = \sum_{i \in A_t} \sqrt{pr(p_a|a)^2 + (\frac{\sum_{j \in A_t} vote_{p_j}(p_a)}{|A_t-1|})^2}, (i \neq j) \quad (7)$$

As describe above, time complexity of this algorithm is $O(n^2)$. However there are so many combinations as candidates, it is time-consuming to calculate every combination. For the purpose of simplify calculation, we adopt an advanced Viterbi algorithm to solve this problem, which we used the anchors as
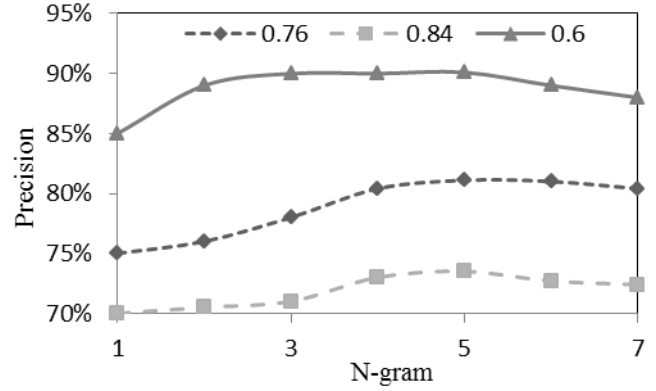
Fig. 3. Performance of ESA system by varying recall and N-gram over the WIKI-ANNOT30

the observation sequence and the senses candidates of each anchor as set of states. Like in the classical Viterbi algorithm, we discover dynamic programming to find the maximum probability path which is a state sequence. The main distinction between the classical model and the one presented in this section is that each anchor's sense depends on the last n-1 anchors' senses. Based on performance considerations, this system uses the set of n = 4, which is obtained based on the experimental. The details are described in the following section.

According to the anchor sequence, make a sense chain. Each anchor is annotated by one sense which is affected by previous anchors' senses. For first anchor to disambiguate, a three length start chain: p_{-2}, p_{-1}, p_0 is set at first, where p_{-2}, p_{-1}, p_0 are setting with no real content. Like figure 2 shows that use anchor sequence of a_1, a_2, … a_n to annotate the sense of each anchor with the candidate senses. In figure 2, a round rectangle stands for an anchor, and the circles below each anchor are the candidate senses for them. The solid line circles form a sense chain. To combine new anchor's sense with current senses chain, previous three sense nodes are used to disambiguate it. With all the previous senses chain of p_{-2}, p_{-1}, p_0, p_1, p_2 ,…,p_{i-1} ($p_{i-1} \in pg(a_{i-1})$), select an appropriate sense from anchor a_i's candidate senses and combine it to the chain. We will get the sense of a_i by:

$$p_i \in pg(a_i)$$
$$= argmax \sqrt{pr(p_i|i)^2 + \left(\frac{\sum_{j=n-3}^{j=n-1} vote_{p_j}(p_i)}{3}\right)^2} \quad (8)$$

Through this optimization, the time complexity is reduced to O(n). It not only gets faster than before but also performs excellently as before.

Annotated senses produced by WSD Phase for each anchor has to be pruned, so as to discard the possible mistake annotated anchors. We use the parameter ρ_{NA} as the threshold

to prune the results. ESA system used each anchor's highest commonness value and highest relatedness value to set the ρ_{NA}:

$$\rho_{NA} = k \sqrt{\left(\max_{p_i \in pg(a)} pr(p_i|a) \right)^2 + \left(\max_{p_j \in pg(a)} \frac{\sum vote(p_j)}{|A_t - 1|} \right)^2} \quad (9)$$

The parameter k is variable from 0 to 1. Using each anchor's ρ_{NA} to compare best senses value, if $P(p_a|a) < \rho_{NA}$, the annotation for a is discarded. The value of k is more close to 1, the ρ_{NA} is greater, this will lead to reduce the recall rate and improve the precision. On the contrary, The k is more close to 0, the recall rate will be improved and the precision will be reduced. The ρ_{NA} can be used to balance recall and precision.

5 Experiments

In order to examine the performance of ESA system, we use two datasets in experiments, which are used in [3]. They are WIKI-DISAMB30 and WIKI-ANNOT30. Since the system needs to query the information on the Wikipedia frequently, it is dally to query it on the internet. We get the May 2010, English Wikipedia dump to the local, and preprocess to be used to query the information and train the process.

For reducing the algorithm time complexity, we use the n-gram model to describe the annotation data structure. A set of experiments is done for setting the appropriate value of N. We use parameter ρ_{NA} to control the recall of the system, and compare the precision in different setting of N from 1 to 7. Figure 3 shows precision floats by different setting of N in three different recall rates. All of three lines grow before N = 4, then the lines float slightly. Based on this, considering the precision and algorithm efficiency, we use the setting of N = 4.

We use standard precision, recall and F-measure to evaluate the performance of the WSD system. In order to evaluate the Word disambiguates system we use the fragment in WIKI-DISAMB30 as input data. Then we use the WSD system to disambiguate the anchors in the fragment and annotate them with the Wikipedia article URI, and compare the annotated result to the result of the anchor given by dataset. We compare the result against to Milne and Witten's disambiguator and Tagme's disambiguator. The results demonstrate among them are showed in Table I.

TABLE I. Performance of three system over WIKI-DISAMB30

	Precision	Recall	F-Measure
Milne & Witten	92.3	84.6	88.3
Tagme	91.7	90.9	91.2
ESA system	88.3	93.4	90.77
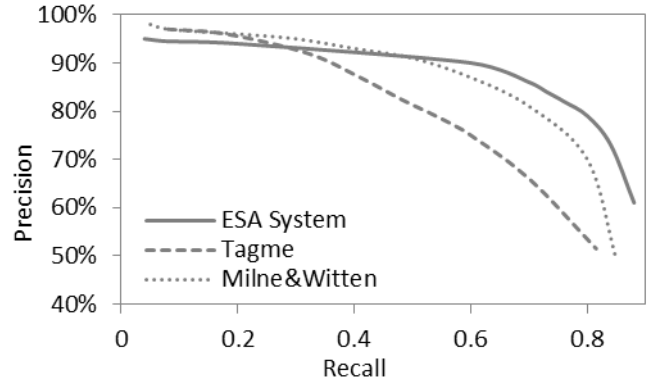
Fig. 4. Performance of three system by varying the value of ρ_{NA}

The precision of ESA system is slightly worse than current systems, but precision is significantly improved, so the overall F-measure of ESA system is better than Milne & Witten's with about 2.4%, and a little worse than Tagme with about 0.5%.

The main difference between ESA system and previous systems is that ESA system evaluates the relatedness between anchor's candidate senses and other anchors senses before using the relatedness and commonness information to vote the anchor's sense. The high score demonstrate the other anchors senses play an important role in affecting the selection of anchor's sense. ESA system will tend to consider using the word relatedness information with the higher weight to disambiguate. On the contrary, ESA system will tend to consider using the word commonness information with the higher weight to disambiguate. This is crucial in our scenario because the name entities in input text may not have the same topic. Some of them may have less relationship with other entities. We plan to dig into the disambiguation's algorithm in order to improve these figures further.

TABLE II. Performance of two disambiguators over WIKI-ANNOT30

	Precision	Recall	F-Measure
Tagme	76.27	76.08	76.17
ESA system	80.03	79.21	79.61

Finally, we evaluated the overall system by the test data set of WIKI-ANNOT30. Results are reported in Table II, where we compare ESA system's annotates precision and recall score with Tagme system. As showed in Table II, ESA system has significantly improves Tagme both in precision and recall. The precision is improved by about 4% and the recall is better than Tagme with about 3%. The reason of improving the recall's score is that ESA system not only annotate the keywords around the topic of input text, but also the other name entities appearance in the text.

Both ESA system and Tagme system need to make a tradeoff of precision and recall. We use parameter ρ_{NA} to control the performance of ESA system. Figure 4 shows the different value of ρ_{NA}, the different performance of the three annotators. As Figure 4 report, ESA system improves the

precision in high recall area; in the area of low recall, ESA system performance is not good. This shows that ESA system has a good performance on improving the annotated recall, because ESA system weakens the relatedness effect to annotate more name entities with less relatedness to the context.

6   Conclusions

This paper introduces a new method based on entropy to disambiguate name entities. It is a better deal of the relationship between name entity senses choices and context. An advanced Viterbi algorithm is employed in ESA system to reduce the complexity of the algorithm. Though the result achieved by ESA system slightly reduces accuracy rate sometimes, it is improved significantly in recall rate than previous systems. This makes ESA system has a wider range of applications, because ESA system annotated most name entities in texts instead of key words in texts.

7   References

[1]   Pandey, Prabhat, and Rahul Arora. "Cross-Lingual Word Sense Disambiguation using Wordnets and Context based Mapping." 2012.

[2]   Milne, David, and Ian H. Witten. "Learning to link with wikipedia." In Proc. of the 17th ACM conference on Information and knowledge management. 2008.

[3]   Ferragina, Paolo, and Ugo Scaiella. "TAGME: on-the-fly annotation of short text fragments (by wikipedia entities)." In Proc. of the 19th ACM international conference on Information and knowledge management. ACM, 2010.

[4]   Mihalcea, Rada, and Andras Csomai. "Wikify!: linking documents to encyclopedic knowledge." In Proc. of the sixteenth ACM conference on Conference on information and knowledge management. ACM, 2007.

[5]   S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti. "Collective Annotation of Wikipedia Entities in Web Text. " In Proc. of ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), pages 457–465, June/July 2009.

[6]   S. Cucerzan. "Large-scale named entity disambiguation based on Wikipedia data. " Proc. of Empirical Methods in NLP, 2007.

[7]   Turdakov, Denis, and Dmitry Lizorkin. "HMM expanded to multiple interleaved chains as a model for word sense disambiguation." Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation.—Hong Kong: City University of Hong Kong. 2009.

[8]   Tonelli, Sara, Claudio Giuliano, and Kateryna Tymoshenko. "Wikipedia-based WSD for multilingual frame annotation." Artificial Intelligence. 2012.

[9]   Pinheiro, Vládia, et al. "Knowledge-intensive word disambiguation via common-sense and wikipedia." Advances in Artificial Intelligence-SBIA 2012. Springer Berlin Heidelberg, 2012. 182-191.

[10]   Sorrentino, Serena, Sonia Bergamaschi, and Elena Parmiggiani. "A supervised method for lexical annotation of schema labels based on wikipedia." Conceptual Modeling. Springer Berlin Heidelberg, 2012. 359-368.

[11]   Nadeau, David, and Satoshi Sekine. "A survey of named entity recognition and classification." Lingvisticae Investigationes 30.1 (2007): 3-26.

[12]   Bikel, Daniel M., et al. "Nymble: a high-performance learning name-finder." Proceedings of the fifth conference on Applied natural language processing. Association for Computational Linguistics, 1997.

[13]   Sekine, Satoshi. "NYU: Description of the Japanese NE system used for MET-2." Proceedings of the Seventh Message Understanding Conference (MUC-7). 1998.

[14]   Borthwick, Andrew, et al. "NYU: Description of the MENE named entity system as used in MUC-7." In Proceedings of the Seventh Message Understanding Conference (MUC-7). 1998.

[15]   Asahara, Masayuki, and Yuji Matsumoto. "Japanese named entity extraction with redundant morphological analysis." Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003.

[16]   McCallum, Andrew, and Wei Li. "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons." Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4. Association for Computational Linguistics, 2003.

[17]   Pasca, Marius, et al. "Organizing and searching the world wide web of facts-step one: the one-million fact extraction challenge." Proceedings of the National Conference on Artificial Intelligence. Vol. 21. No. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

[18]   Etzioni, Oren, et al. "Unsupervised named-entity extraction from the web: An experimental study." Artificial Intelligence 165.1 (2005): 91-134.

Method to add new attributes to concepts by Web and second-order attributes

Misako Imono[1]**, Eriko Yoshimura**[2]**, Seiji Tsuchiya**[2] **and Hirokazu Watabe**[2]

[1]Dept. of Knowledge Engineering & Computer Sciences, Graduate School of Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan

[2]Dept. of Intelligent Information Engineering & Sciences, Faculty of Science and Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract **-** Natural language is composed of various expressions and phraseology, but humans can properly handle these ambiguities by using their knowledge of words. To have such a human mechanism in a machine such as a robot, it is necessary to model the knowledge of words and process their associations. Knowledge of a word that is common sense to humans was modeled on a machine as a Concept Base. A Concept Base is a knowledge base that defines words as concepts. The concept is defined by a set of attributes representing a characteristic of the concept by using other concepts and weights. This paper proposes a method to add new attributes for concepts that are already defined in the Concept Base. Words suitable as attributes are automatically acquired by the Web and by using the attributes ordered by the chain structure of the Concept Base.

Keywords: Concept Base, degree of association, association mechanism

1   Introduction

Humans interpret natural language flexibly based on their knowledge of words and association skills. Natural language is composed of various expressions and phraseology, but humans can properly handle these ambiguities by using their knowledge of words. To have such a human mechanism in a machine such as a robot, it is necessary to model the knowledge of words and process their associations.

Humans understand the knowledge of one word associated with the knowledge of other various words. For example, humans associate "suck" as an action, "luxury goods" as a classification, and "nicotine" as a by-product for the word "tobacco." Humans can perform natural association and understanding of the relation of words by having varied knowledge of the words. Such knowledge of a word that is common sense to humans was modeled on a machine as a Concept Base [1] [2].

A Concept Base is a knowledge base that defines words as concepts. The concept is defined by a set of attributes representing a characteristic of the concept by using other concepts and weights. The weights prioritize the importance of the attributes. Common-sense meaning can be given to the concept by using knowledge of the word as an attribute. So, semantic definitions of the concept having many appropriate attributes are more natural for humans.

This paper proposes a method to add new attributes for concepts that are already defined in the Concept Base. Words suitable as attributes are automatically acquired by the Web and by using the attributes ordered by the chain structure of the Concept Base. Processing to add new attributes is performed by selecting and weighting the possible candidate attributes. It is possible to automatically enrich the meaning of concepts by the proposed method.

2   Association System

The association system based on the Concept Base and the degree of association imitates the knowledge of words and the capability of human association. In the following sections, the Concept Base and the degree of association are described.

2.1   Concept Base

A Concept Base, which is a knowledge base that defines words as concepts, is expressed by the following equation.

$$A = \{(a_1, w_1), (a_2, w_2), ...(a_L, w_L)\} \qquad (1)$$

A is the concept label, a is the attribute, and w is the weight of the attribute. The Concept Base in the Association System builds up to 87242 concepts. Table 1 shows specific examples of some concepts.

Table 1: Specific example of concepts

Concept	(Attribute, Weight)
Summer	(summertime, 0.34) (summer vacation, 0.11) ...
Summertime	(heat, 0.18) (sun, 0.04) ...
...	...

In the Concept Base, words defined as concepts also form the attributes, called first-order attributes. In addition, attributes can be derived from other attributes. Attributes derived from other attributes are called second-order attributes of the original concept. So, the Concept Base is defined by a chain structure of attributes, as shown in Figure 1.

Second-order attributes of
concept "Summer"

sun   **home coming**

heat   **travel**

Summer — **Summertime** — **Summer vacation** →

First-order attributes of
concept "Summer"
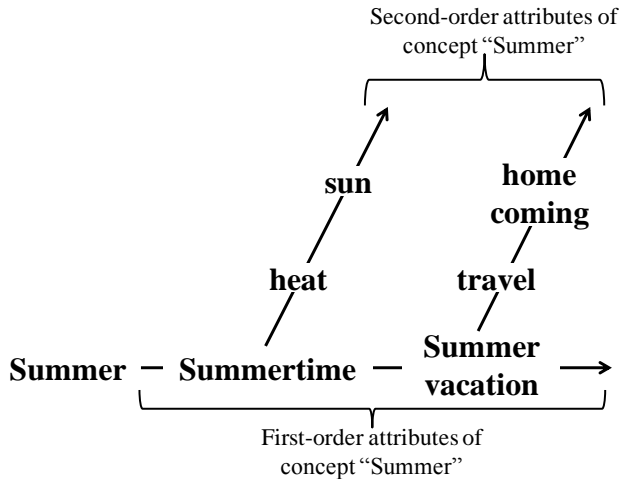
Figure 1: Flowchart of the proposed method

In the figure, the "summertime" attribute of the "summer" concept is defined as a concept. So, the "heat" and "sun" attributes of the "summertime" concept are second-order attributes of the "summer" concept.

2.2   The degree of association

The degree of association (*DoA*) [3] quantifies the relationship between concepts by using attributes that characterize the chain-reaction structure of the Concept Base. The relationship between multiple concepts is expressed quantitatively in this process. The following shows how to calculate the degree of association between concept *A* and concept *B*. This is defined as *DoA (A, B)*.

For concepts *A* and *B* with primary attributes a_i and b_j, weights u_i and v_j, and numbers of attributes *L* and *M*, respectively ($L \leq M$), the concepts can be expressed as follows:

$$A = \{(a_1, u_1), (a_2, u_2), ... (a_L, u_L)\} \qquad (2)$$

$$B = \{(b_1, v_1), (b_2, v_2), ... (b_M, v_M)\} \qquad (3)$$

The degree of match DoM(A,B) between concepts A and B is defined as follows, where the sum of the weights of the various concepts is normalized to 1:

$$DoM'(A, B) = \sum_{a_i = b_j} \min(u_L, v_j) \qquad (4)$$

The degree of association is calculated by calculating the degree of match for all of the targeted primary attribute combinations and then determining the relation between the primary attributes. Specifically, priority is given to the correspondence between matching primary attributes. For primary attributes that do not match, the correspondence between primary attributes is determined to maximize the total degree of matching. By using the degree of matching, it is possible to give consideration to the degree of association, even for primary attributes that do not match perfectly. When the correspondences are thus determined, the degree of association DoM(A,B) between concepts A and B is as follows:

$$DoA(A, B) = \sum_{i=1}^{L} DoM(a_i, b_{xi}) \times \frac{(u_i + v_{xi})}{2} \times \frac{\min(u_i, v_{xi})}{\max(u_i, v_{xi})} \quad (5)$$

In other words, the degree of association is proportional to the degree of identity of the corresponding primary attributes, the average of the weights of those attributes, and the weight ratios.

3   Acquisition of Candidates for attribute

Candidates for an attribute are acquired by two methods: those using the Web and those using second-order attributes derived from the Concept Base structure. The acquisition method of candidates for an attribute is described below.

3.1   Acquisition of candidates for attribute from Web

This process obtains the candidates for an attribute by a Web search. The Web search is performed by using the concepts defined in the Concept Base. Words on the search result pages are considered to be related with the searched concept and therefore are acquired as candidates for the attribute[4]. Figure 2 shows words obtained by the search of the concept "Meniere's disease." The words shown in bold are not registered as attributes of the current concept. Thus, this process can obtain candidates for a new attribute of the concept from the Web.

Concept
  ⇒ Meniere's disease
Obtained words by Web search
  ⇒ **dizziness**, symptoms, tinnitus, **hearing loss**,
     inner ear, disease, **treatment, ear, causing,
     nausea, stress**, seizures, **diagnosis, rotation,
     baskets, medicine, wed tumor, vomiting,
     stage, disease, deterioration, lymph, liquid,
     medicine, lymph, inspection, hospital, sudden**

Figure 2: Words obtained by search of "Meniere's disease."

3.2 Acquisition of candidates for attribute from second-order attribute

The Concept Base is defined as a chain structure of the attribute, as described in Section 2.1. The meaning of concept X is defined by the first-order attributes of X. Similarly, the meaning of each first-order attribute is defined by each of its own attributes, which are the second-order attributes of concept X. It is considered that words directly related to concept X are included in the second-order attributes derived from concept X. So, this process obtains second-order attributes as candidates of the concept. Figure 3 shows how to get the new attributes from the second-order attributes of the concept "winter."
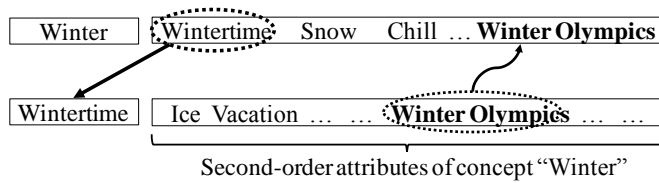
Figure 3: Example of adding attributes to the concept "Winter."

The second-order attributes of "Wintertime," which is a first-order attribute of the concept "Winter," are obtained. These are second-order attributes of the concept "Winter." Second-order attributes obtained are derived from other first-order attributes as well. For example, the second-order attribute "Winter Olympics" is obtained as a first-order attribute of the concept "Winter."

4 Selection of Candidates for attribute

Some candidates for attributes obtained from the Web are not important for the concept because too much noise is present. In addition, a huge number of candidates can be obtained from second-order attributes because the average number of first-order attributes in the Concept Base is 37. Namely, the number of second-order attributes is 1369, which is the square of 37.

The selection of candidates is performed by using the importance of attributes. The importance is calculated in three ways: the inverse document frequency (*IDF*) in the Concept Base, the degree of association (*DoA*), and the weight of the candidates for attributes. The selection is experimentally performed with the importance of each in a variety of thresholds. Each importance is described in the following sections.

4.1 Selection by IDF in Concept Base

IDF in the Concept Base is one of the criteria of the value of each concept in the Concept Base. The idea of term frequency-*IDF* (*TF-IDF*) is often used in document

processing. The IDF in the Concept Base is calculated by viewing the Concept Base as a single document space. As the IDF increases, the importance of the concept increases. The calculation formula of the *IDF* in the Concept Base is defined as follows.

$$IDF\_CON_N(X) = log_2 \frac{V_{all}}{df_N(x)} \qquad (6)$$

$IDF\_CON_N(X)$ is the *IDF* in the Concept Base of concept X in the N-order attribute space, V_{all} is the number of all concepts defined in the Concept Base, $df_N(X)$ is the number of concepts that have concept X as an attribute in the N-order attribute space.

The less frequent the appearance of an attribute is, the higher the value of the IDF in the Concept Base. The concept that has a larger value of the IDF in the Concept Base is important when the concept appears as an attribute. For example, when compared to the number of concepts that have "Color" as attributes, the number of concepts that have "Purple" as attributes in the N-order attribute space is less. The concept that less frequently appears as an attribute strongly characterizes the other concept if the concept appears as the attribute. Thus, the attribute "Purple" is more important for identifying the meaning of concepts than is the attribute "Color." Therefore, the selection of candidates for an attribute by IDF is performed by setting a threshold for the IDF in the Concept Base. If the IDF in the Concept Base of a candidate for an attribute is higher than the threshold, this candidate is chosen as an attribute. In this paper, this selection method uses third-order attribute space because this order was the most accurate in past research [5].

4.2 Selection by degree of association (DoA)

As previously stated, attributes are words that define the meaning of the concept. Therefore, the relevance of the concept and the correct attributes is higher. The selection method is performed by setting a threshold for the degree of association (*DoA*) between a concept and the candidates for attributes.

4.3 Selection by weight of candidates for attribute

It is necessary to give a weight, which indicates the attribute importance for a concept, to all attributes in the Concept Base. In this selection method, candidates for attributes are given a weight and selected by using the weight threshold.

The weight of attribute *B* for concept *A* is defined by the following equation.

$$w(A,B) = DoA(A,B) \times IDF\_CON_3(B) \qquad (7)$$

DoA(A,B) is the degree of association between concept *A* and attribute *B*. The *IDF* in the Concept Base is calculated in third-order attribute space. This selection method is performed by setting a threshold for the weight calculated by this equation.

5   Evaluation of Concept Base

For evaluation of the Concept Base, *X-ABC* evaluation data, which is composed of a set of four concepts, is prepared. In this set, concept *A* is the most relative concept, concept *B* is the relative concept, and concept *C* is not relative with basic concept *X*, which is one of any of the concepts defined in the Concept Base. These data contain 500 sets of concepts. Table 2 shows examples of the *X-ABC* data.

Table 2: *X-ABC* evaluation data

X	A	B	C
Music	Musical	Sound	Train
Sea	Ocean	Salt	Car
...	...	...	...

The evaluation method uses the degree of association between concepts. The degree of association between *X* and *A* is expressed as *DoA(X,A)*. In the same way, the degrees of association between *X* and *B*, *X* and *C* are expressed as *DoA(X,B)* and *DoA(X,C)*, respectively. At this time, if the following equations are satisfied, it is assumed that the correct answers are given.

$$DoA(X,A) - DoA(X,B) > AveDoA(X,C) \qquad (8)$$

$$DoA(X,B) - DoA(X,C) > AveDoA(X,C) \qquad (9)$$

$$AveDoA(X,C) = \frac{\sum_{i=1}^{500} DoA(X_i,C_i)}{500} \qquad (10)$$

DoA(X,C) must be 0.0 ideally because no relative exists between concept *X* and concept *C*. But, as a characteristic of the calculation of the degree of association, the case that only one attribute pair exists between *X* and *C*, that is, no relative concept *C* is associated with *X*, is not calculated to be 0.0. Given this result, the calculation of the degree of association has some error. Consequently, when the degree of association between *DoA(X,A)* and *DoA(X,B)* and between *DoA(X,B)* and *DoA(X,C)* has a difference that is more than the error margin, the evaluation is correct. The correct ratio of this evaluation method is called the *C-Average Rank Rate*.

6   Accuracy Evaluation

We made the selection described in Section 4 for each candidate for an attribute. Table 3 shows the highest accuracy obtained in each selection. Integrated acquisition acquires candidates for an attribute by using both methods of acquisition (from the Web and from second-order attributes).

As a final result, the proposed method for the integrated acquisition of attributes obtained a maximum accuracy improvement of 2.0% compared with the acquisition either from the Web or from the second-order attributes.

Table 3: *X-ABC* evaluation result

	IDF in Concept Base	Degree of association	Weight
Before acquisition	83.6%		
Acquisition from Web	84.2%	83.3%	82.0%
Acquisition from second-order attributes	84.2%	84.0%	84.6%
Integrated acquisition	85.6%		

Table 4 shows the change in the average number of attributes and the difference before acquisition and after each method.

Table 4: Average number of attributes and difference

	Average number of attributes	Difference before and after
Before acquisition	38	-
Acquisition from Web	41	3
Acquisition from second-order attributes	44	6
Integrated acquisition	58	20

The number of attributes increases in all methods by the acquisition of attributes for the concepts. The integrated acquisition method is able to acquire the most attributes. Table 5 shows examples of the success of the attributes acquired by both methods.

Table 5: Average number of attributes and difference

Concept	Attributes from second-order attributes	Attributes from Web
Rain	Rainy weather, Heavy rain	Umbrella, Forecast, Observation
Paper	Kent paper	Recycling, Pulp
Happy	Cried for joy	Cheering, Souvenir, Service

The attributes specific to the concept were obtained from the second-order attributes. However, the attributes associated with the concept but not identical to the concept are obtained by using the Web. For example, the attribute of the concept "Paper" from the second-order attributes is one specific type of paper, "Kent paper." In contrast, the attribute "Recycling", which is associated with "Paper" and the attribute "Pulp" as the raw material for "Paper," are obtained.

7   Conclusion

This paper proposes a method to acquire new attributes for concepts by integrating methods using the Web and second-order attributes. As a result of the proposed method, the average number of attributes increased from 38 to 58. The obtained Concept Base accuracy was improved by 2.0% compared to that before adding the attributes. This result shows that Concept Base can expand a concept's meaning by adding attributes. Thus, the proposed method automatically enriches the meanings of concepts.

Acknowledgment

References

[1]   K. Kojima, H. Watabe, and T. Kawaoka, "A Method of a Concept-base Construction for an Association System: Deciding AttributeWeights Based on the Degree of Attribute Reliability", Journal of Natural Language Processing, Vol.9, No.5, pp.93–110 , 2002.

[2]   N. Okumura, E. Yoshimura, H. Watabe, and T. Kawaoka: "An Association Method Using Concept-Base", KES 2007/WIRN2007, Part I, LNAI4692, pp.604–611, 2007.

[3]   H. Watabe and T. Kawaoka: "The Degree of Association between Concepts using the Chain of Concepts", Proc. of SMC2001, pp.877-881, 2001.

[4]   Y. Tsuzi, H. Watabe and T. Kawaoka: "The Method of Acquisition of The New Concept and Its Attribute Using The World Wide Web", The 18th Annual Conference of the Japanese Society for Artificial Intelligence, 2D1-01, pp.1-4, 2004.

[5]   T. Araki, N. Okumura, H. Watabe and T. Kawaoka: "Dynamic Calculation Method of Degree of Association Considering the Common Attributes of Target Concepts", The Science and Engineering Review of Doshisha University, Vol.48, No.3, pp.14-24, 2007.

DEVELOPMENT OF FILIPINO PHONETICALLY-BALANCED WORDS AND TEST USING HIDDEN MARKOV MODEL

Arnel C. Fajardo[1] and Yoon-joong Kim[2]

Department of Computer Engineering, Hanbat National University, Daejeon City, Republic of Korea

Abstract- *In this paper, two sets of phonetically balanced words (PBW) in Filipino were developed; namely the 2-syllable, and 3-syllable PBW list. These are tested as a speech corpus in a word-level recognizer using the Hidden Markov Model (HMM) as a framework and Mel-Frequency Cepstral Coefficient (MFCC) as a feature extraction technique. Thus, this study is a preparation for a Large-corpus Filipino Language ASR using HMM.*

For the testing of the PBW sets, fifty speakers were trained (25 male and 25 female speakers). For the recognition of the 2-syllable word list, an average accuracy rate of 93.25% and 88.67% were achieved for the speaker dependent and speaker independent tests, respectively. For the recognition of the 3-syllable word list, the recognizer achieved an accuracy rate of 99.53% and 96.30% for the speaker dependent and speaker independent tests, respectively.

Keywords: Filipino phonetically balanced words, Filipino word corpus, Hidden Markov Model

1   Introduction

Automatic speech recognition (ASR) is the process of decoding speech into its corresponding word sequence. In the recent years, Filipino researchers aimed to provide an accurate speech recognizer [1][2]. However, an efficient solution for the Filipino Language has not been successful.

The Hidden Markov Model (HMM) is a doubly stochastic process with one that is not directly observable [3]. This hidden process can be observed only through another set of stochastic process that can produce the observation sequence.

HMMs are widely used as a basis for pattern-based acoustic model for speech recognition [4]. This model was used in the previous studies relating to an Automatic Speech recognizer for the Filipino Language [1][2].

In 2003, an ASR for Filipino phonemes was developed [1]. This study reported to have achieved recognition accuracy of 85.5%. However, this recognizer was used to recognize phoneme utterances of the Filipino alphabets using discrete HMM limiting itself for small vocabulary speech corpus.

A Filipino Speech Corpus was developed by Guevara R., et. al (2002) to be used for continuous Speech recognition. Dela Roca G., et.al (2003) tested the developed Filipino corpus to recognize continuous speech and achieve 32% recognition accuracy. In an attempt to increase this accuracy, a study in 2010 was conducted wherein an Indonesian speech corpus was used for the recognizer as training sets to recognize Filipino utterances [5]. The Indonesian speech corpus contains 80 hours of recording compared to the developed Filipino speech corpus in 2003 that contains 4 hours of recording. This cross-lingual approach achieved 79.50% recognition accuracy. However, none of these previous researches used a phonetically balanced set of words for the development of its speech corpus.

Thus, the researchers' objectives are to (1) present the development of Filipino PBW and (2) test the recognition accuracy of the developed Filipino PBW using HMM.

This paper will provide some information of Filipino language, the development of phonetically balanced-word lists and the recognition accuracy test for the phonetically balanced speech.

2   The Specifics of Fipino Language

Filipino is the language used largely in the Philippines with 22 million native speakers [6].

Between the 1930s and mid-1970's, a system of syllabication for the alphabet called *abakada* was developed by Lope K. Santos to represent the native sounds [7]:

> *a ba ka da e ga ha i la ma na nga o*
> *pa ra sa ta u wa ya*

to represent the Filipino alphabet, consisting of 5 (a , e, i, o, u) vowels and 15 (b, k, d, g, h, l, m, n, ŋ , p, r, s, t, w, y) consonants. The Filipino alphabet, though in a sense, considered as phonetic, does not reflect exactly the correct sound in written form [8]. There are words present in the Filipino Language that are spelled the same but are pronounced with a slight difference, which produces difference in meaning.

> bata /b:a - ta/ - "a child"
> bata /ba - ta/ - "to bear or endure"

The word *bata* with the phonetic representation of /b:a – ta/ denotes a long sound which is produced by a short pause after the affected syllable, while the other phonetic representation /ba-ta/ is produced continuously without breaks.

Thus, the Filipino phonemes can be broken down into the following phonemes:

Vowels
/a/ /e/ /i/ /o/ /u/

Consonants
/b/ /k/ /d/ /g/ /h/ /l/ /m/ /n/ /ŋ/ /p/ /r/ /s/ /t/ /w/ /y/

The Filipino vowel phonemes can be described as /a/ low central unrounded, /e/ mid front unrounded, /i/ high front unrounded, /o/ mid back rounded, and /u/ high back rounded. According to tongue height, we have two front vowel phonemes /i e/ , and two back vowel phoneme /o u/, and one central vowel phoneme /a/.

The Filipino consonants are produced through the help of the lips (labial), teeth (dental), alveolar ridge (alveolar), palate (palatal), velum (velar), and glottis (glottal).

3   Phonetically Balanced Words

In the development of an automatic speaker-independent continuous Filipino speech recognizer, a set of properly selected word list is required for the development of a recorded speech corpus. The set of words must be phonetically balanced in nature, and must contain all phonemes present that are characterized by the Filipino language.

In the construction of large-vocabulary word recognition, a set of recording must be obtained from a spoken corpus gained from a written corpora or a phonetically balanced word list. A Filipino Speech corpus was developed by Guevara, et. al [2] that includes both open-ended and close-ended spoken words. This methodology in training data is not phonetically balanced. A phonetically balanced speech text used for English, German, Swedish, Danish, Hebrew, Italian, Finish, French, and Portuguese often taken into the following criteria [9][10]: Syllable structure, equal phonetic structure, phonetic balance, equal average difficulty and equal range of difficulty, common words, and speaker intelligibility.

In [13], a total set of 257 words was produced for the 2-syllable word list, and 212 words for the 3-syllable word list. These words were then used for the recording of speech corpora that are used for testing via word recognition.

3.1   Developed Phonetically Balanced Word Lists

The Filipino phonetically balanced words were evaluated from 16 articles found from a Filipino based textbook written for senior public school students, "Bagwis". This textbook is approved by *the Department of Education,* an executive department of the Philippine Government which governs the Philippine system of basic education, thus be considered reliable with a minimal chance of error. All the articles extracted from the textbook are written in Filipino, which consists of a total of 9768 words. The 9768 extracted words were inspected and a list of unique words was gathered. The list of unique words was manually transcribed phonetically based on the UP Diksyonaryong Filipino, a monolingual dictionary maintained by the University of the Philippines Center for Languages [11]. Phonemes such as /p:/ /b:/ /m:/ /t:/ /d:/ /n:/ /s:/ /l:/ /k:/ /g:/ were included to denote a longer duration of phoneme pronunciations as compared to its shorter counterparts. The diphthongs /iw/ /ay/ /aw/ /oy/ /ey/ /uy/ were also included as part of the vowel phoneme list.

The 2938 unique words were gathered where the frequency of words, phonetic structure, syllabications, and number of syllables was included from the 9768 word included in the articles.

Table 1: Frequency of Syllable Counts from the Extracted Unique Words

Syllable Count	Frequency
1-syllable	101
2-syllable	780
3-syllable	912
4-syllable	740
5-syllable	299
6-syllable	72
7-syllable	23
8-syllable	7
9-syllable	2
10-syllable	1
13-syllable	1
Total	**2938**

2-syllable and 3-syllable word lists were selected based on the frequency of its occurrence from the articles. These word lists were then filtered by the number of times it is used in the

article, thus a total of 323 2-syllable and 249 3-syllable words were extracted of which have more than 1 frequency of occurrence in the list to ensure commonality of words.

Table 2: 2-syllable words and 3-syllable words and its frequency of occurrences.

Syllable Count	1 Occurrence	>1 Occurrences	Total
2-syllable	457	323	780
3-syllable	663	249	912

These words are grouped according to their phonetic structure, taking phonetic structure constituting at least 80% of the total numbers of the 3-syllable and 2-syllable words.

Two hundred fourteen (214) of the 3-syllable words are represented with the following phonetic structures: *cv-cv-cvc, cv-cv-cv, cv-cvc-cvc, cvc-cv-cv, cvc-cv-cvc, v-cv-cvc, cv-cv-vc, cv-cvc-cv,* and *v-cv-cv* with the frequencies of 0.416, 0.1526, 0.0602, 0.0441, 0.0562, 0.0441, 0.0361, 0.0321, and 0.0321, respectively; while, 261 of the 2-syllable words are represented with the following phonetic structures: *cv-cvc, cv-cv, v-cvc,* and *cvc-cv* with the frequencies of 0.4024, 0.1889, 0.1331, and 0.0836 respectively.

Table 3: Phonetic Structure of 2-Syllable Unique Word List

Phonetic Structure	Frequency	Vowel	Consonants
cv-cvc	130	260	390
cv-cv	61	122	122
v-cvc	43	86	86
cvc-cv	27	54	81
cvc-cvc	25	50	100
cv-vc	13	26	26
v-vc	12	24	24
vc-cvc	3	6	9
ccv-cv	3	6	9
vc-cv	2	4	4
cv-vc	1	1	2
v-v	1	2	0
cvc-ccv	1	2	4
cvc-ccvc	1	2	5
cv-v	1	2	1
Total	**323**	**646**	**849**

Table 4: Phonetic Structure of 3-Syllable Unique Word List

Phonetic Structure	Frequency	Vowel	Consonant
cv-cv-cvc	100	300	400

cv-cv-cv	38	114	114
cv-cvc-cvc	15	45	75
cvc-cv-cvc	14	42	70
v-cv-cvc	11	33	33
cvc-cv-cv	11	33	44
cv-cv-vc	9	27	27
v-cv-cv	8	24	16
cv-cvc-cv	8	24	32
cv-v-cvc	6	18	18
cv-cv-v	3	9	6
cv-vc-cvc	3	9	12
cv-v-cv	3	9	6
cvc-cvc-cv	3	9	15
cvc-cvc-cvc	2	6	12
cvc-cv-vc	2	6	8
cvc-v-cvc	2	6	8
vc-ccv-cvc	1	3	5
vc-cv-cvc	1	3	4
vc-cv-ccvc	1	3	5
v-v-cvc	1	3	2
v-cvc-cvc	1	3	4
cvc-v-cv	1	3	3
cvc-vc-cvc	1	3	5
cvc-cv-v	1	3	3
cv-cv-ccv	1	3	4
ccvc-cv-cv	1	3	5
ccv-cv-cv	1	3	4
Total	**249**	**747**	**940**

The total frequency of the phonemes represented (*F) is the* summation of all frequency of phonemes in a word (*pfw)* representing the number of a specific phoneme in a word [e.g., *pfw* (a) = 2 in the word *bata*] multiplied by the frequency of word occurrence implies the number of times the specific word occurred from the whole text corpus, divided by the total of phoneme frequencies in the word list.

A frequency of each phoneme is calculated with the formula:

$$F = \frac{\sum(pfw*wf)}{n} \qquad (1)$$

Where:
F   frequency of phonemes represented in the word list
pfw   frequency of phoneme in a word
wf   frequency of word occurrence
n   total number of phonemes in the word list

This value is compared to the acceptance value (*threshold value)* with the formula:

$$aV = \frac{1}{(x*m)} \qquad (2)$$

Where:
aV   acceptance value/threshold value
x   average of the vowels/consonants in a phonetic structure
m   total number of words

Table 5: Acceptance Values of Phonemes for the 2-Syllable and 3-Syllable Words

	Vowel	**Consonants**
2-syllable words	0.0023	0.0018
3-syllable words	0.0015	0.0012

The acceptance value for the frequency of vowels from the 2-syllable word list is 0.0023 and 0.0018 for the consonants based on the 261 2-syllable words while the acceptance value for the vowels from the 3-syllable word list is 0.0015 and 0.0012 for consonants based on 214 words in list. These values are compared from the frequency of each phoneme in the list to validate if the phoneme is well represented. Phonemes lower than the acceptance values would not be represented, thus the words including the phonemes will be removed from the accumulated list while frequencies higher or equal to the acceptance values well represented by the list.

From the 20 basic phonemes of the Filipino language, 16 phonemes were added (10 long consonants, and 6 diphthongs). 5 phonemes were not represented in the 2-syllable word list (/m:/, /g:/, /iw/, /ey/, /uy/) since the frequencies of these phonemes are less than the acceptable value of 0.0023 for vowels, and 0.0018 for consonants. 4 phonemes were not represented in the 3-syllable word list (/m:/, /g:/, /iw/, /ey/, /uy/) which are less than the acceptable values of 0.0015 for vowels and 0.0012 for consonants. The total number of phonemes represented by the 3-syllable word list is 214, and 261 for the 2-syllable word list. Thus, the final list for the phonetically balanced word list for the Filipino Language will be 257 for 2-syllable words, and 212 for 3-syllable words.

Table 6: Word Count For The 3-Syllable and 2-Syllable List with the Number of Represented Phonemes

	List 1 3-Syllable Word List	**List 2 2-Syllable Word List**
Phonemes represented	32	31
Number of words per list	212	257

3.2   Phonetically Balanced Speech Corpus

The speech corpus is a requirement for the development of automatic speech recognition systems. The speech corpora used in this study were gathered from the 212 2-syllable word list (PBW2) and 257 3-syllable word list (PBW3). These words were recorded for the training and testing of words using an ASR system developed with HMM. The speech data were recorded from 50 fluent Filipino respondents (25 male, and 25 female). All members of the group have Filipino as their primary and native language, with no speaking ailments, and were at their proper dispositions.

4   Test of Phonetically Balanced Speech

Hidden Markov Models (HMM) are probabilistic models [12] used for modeling stochastic sequence with underlying finite state structure. HMM is a widely used tool to model a speech utterance for contemporary speech recognition development.

In this study, HTK toolkit was used to develop the speech recognizer using the Hidden Markov Model as the framework, and MFCC as the feature extraction method.

4.1   Speech Data

The recordings were done in an isolated room using a uni-directional microphone connected to a computer with input speech sampled at 16 kHz at mono using a linear PCM and saved as waveform file format (*wav). A distance of approximately 5-10 centimeters is used between the mouth of the speakers and the microphone used.

The speakers were asked to utter the words from the PBW2 and PBW3 word lists clearly. The speeches collected were used as training data, and test data.

The training data are used from 20 female and 20 male speakers, which recorded 2 sets of word utterances. The test data were grouped as 'speaker dependent' and 'speaker independent'. The speaker dependent speech were taken from the same speakers from the training data (20 female and 20 male), which recorded another set of word utterances while the speaker independent speech were taken from 5 female and 5 male speakers not included in the training data which recorded a set of word utterances.

4.2   Feature Specifications

The feature extraction stages the use of Mel-Frequency Cepstral Coefficients (MFCC) to estimate each of the speech data using the HTK tool, HCopy. The main feature extraction parameters used in the study consists of 39 dimensional feature vectors of 13 MFCCs, the first and second derivatives. The window size is set to 25ms with a target rate of 10ms using a hamming windowing. The pre-emphasis coefficient value is set to 0.97.

4.3 HMM Acoustic Model Specifications

The speech data are trained into a 7-state HMM for the PBW2, and 9-state HMM for PBW3. The training is performed through multiple iterations to re-estimate the parameters using the Baum-Welch re-estimation algorithm. In this study, a total of 20 re-estimation iterations were conducted. The model is for the state machines that consist of 7 different states, and 9 different states for the 7-state and 9-state HMM respectively, with the first and the last state characterized as the non-emitting entry and exit states representing null or silence in the speech data.

5   Results

The performance of the ASR is tested against two types of speakers: one of which was involved in the training (dependent speakers) and the other of which was only involved in the testing (independent speakers). The evaluations for the testing are shown in Table 7. For the testing of the PBW2 list, a recognition rate of 93.25% is gathered for the dependent data, and 88.67% for the independent data. For the PBW3, a recognition rate of 99.53% is gathered for the dependent data, and 96.30% for the independent data.

Table 7: Performance Evaluation Results

	Dependent Speaker	Independent Speaker
PBW2	93.25 %	88.67 %
PBW3	99.53 %	96.30 %

6   Conclusion

In this study, a speech recognition system was developed to test the recognition of the developed Filipino phonetically balanced word list (PBW2 and PBW3) using continuous density HMM. The implemented system for the PBW2 achieved a recognition rate of 93.25% and 88.67% for the dependent and independent testing data, respectively; while a recognition rate for PBW3 of 99.53% and 96.30% for the dependent and independent data.

The developed PBW lists can be improved by increasing the size of text corpora, which will also increase the number of words in the PBW list. With this, a new PBW list can be examined for the development of a speech corpus.

Future works will involve of increasing the Filipino text corpora in a phoneme level recognition to further improve the quality and the efficiency of the speech recognizer. This will be used to train a larger set of vocabulary for the improvement of the performance and recognition accuracy of the system.

7   References

[1] Navaro, R. D.*, Recognition of Tagalog Alphabets Using The Hidden Markov Model*, (2007).

[2] Guevara, R., Co, M., Espina, E., Gracia, I., Tan, E., Ensomo, R., and Sagum, R., *Development of a Filipino speech corpus*, (2002).

[3] Rabiner, L. R. and Juang, B. H., *Fundamentals of Speech Recognition*, Englewood Cliffs, NJ, Prentice Hall, (1993).

[4] Kumar, K., *Hindi Speech Recognition System using HTK*, International Journal of Computing and Business Research (Vol 2, Issue 2), (2011).

[5] Sakti S., Isotani, R., Kawai H., and Nakamura, S., *The Use of Indonesian Speech Corpora for Developing a Filipino Continuous Speech Recognition System,* (2010).

[6] http://wika.pbworks.com/w/page/8021671/Kasaysayan, "*Ebolusyong ng Alpabetong Filipino*", (Retrieved 2012).

[7] Lewis, P. M., *Languages of Philippines. Ethnologue: Languages of the World* (16th ed.), (2009).

[8] Santiago, A., Tiangco, N.*, Makabagong Balarilang Pilipino*, (1985).

[9] Ferrer O., *Speech audiometry: a discrimination test for Spanish language*, (1960).

[10] Rosas y De Mendizabal B., *Speech Audiometry in English, Portuguese and Spanis*h, (1958).

[11] Travero, K.*, UP Diksyonaryong Filipino*, 1st Edition, UP Publishing Press, Quezon City, (2001).

[12] Hachkar, Z., Farchi A., Mounir. B., El Abbadi J., *Building HMM recognizer for Arabic Digits using HTK,* (2007)

[13] Fajardo A., Kim Y.J., *Test of Vowels in Speech Recognition using Continuous Hidden Markov Model and Development of Phonetically Balanced-Words in the Filipino Language,*5th ICEBE (2012)

How to Make NL "Understanding" Systems
Run Orders of Magnitude Faster

by Steve Richfield, IEEE 41344714

CTO of *our thing*

5498-124th Avenue East; Edgewood WA 98372

Steve.Richfield@gmail.com   505-934-5200

Abstract - *A new method of greatly speeding up natural language (NL) "understanding" is presented here. This speedup is achieved by nearly eliminating the overhead of failed tests during parsing and subsequent processing. Failed tests account for more than 99% of time taken by present natural language understanding systems. This speedup is independent of the particular NL processing algorithms used.*

This method triggers queuing of rules based on the appearance of their least frequently used elements, so rules whose least frequently element is absent incur no overhead. Lower-level rules are placed in higher-priority queues. High-level rules create output.

This new capability should revolutionize the Internet by making practical the real time analysis of postings for particular semantic content to trigger individually crafted advertisements or AI replies. This method will also facilitate developing a "drop in" natural language understanding module suitable for many disparate applications.

Keywords: Natural language processing, Software performance, Detection algorithms, Dynamic compiler.

1 Introduction

A subtle phenomenon has doomed natural language (NL) projects for the last 40 years. Current literature is almost devoid of recognition of this problem, let alone providing any discussion of potential solutions. Researchers have built countless NL systems only to abandon them without public explanation. I will explain this phenomenon and present a practical solution.

2 NLP's Dirty Little Secret

Natural language processing (NLP) has long been concerned with NL parsing, disambiguation, and semantic representation, together collectively referred to as "understanding". However, 40 years of effort have failed to produce a good method for understanding NL. Examination of past NL projects, including my own DrEliza.com, has uncovered a hidden problem that doomed these projects right from their start – as the rules and relationships grow in numbers, complexity, and depth, the number of tests that fail undergo a combinatorial explosion that quickly limits the number of rules that can be honored within an acceptable processing time. This continues to happen despite the orders of magnitude improvement in processing speed that has become available in recent years. To illustrate, to achieve a 3-second response time to short open-ended NL queries falling far short of "commercially viable production performance" IBM's WATSON utilizes 2,880 POWER7 processors, making it the 94th fastest supercomputer in the world.

NL projects seem to go through a common development cycle. Researchers implement a demonstration, then the program slows nearly to a stop as they start adding rules on the way to making it useful enough to sell. Soon it becomes apparent that adding more rules is counterproductive because the program is already running too slowly to demonstrate in real time. This occurs before entering thousands of idioms, adding automatic spelling correction, or including other important pieces of a commercial quality system. Then, careful analysis of the programming usually uncovers clever ways of making the program run an order of magnitude faster, e.g. by moving language rules from a database into arrays. However, by this time it has become clear that an order of magnitude is not nearly enough additional speed to achieve the desired performance, so yet another NL project is "shelved".

Researchers have attempted to address this situation by making their rules smarter (and in some cases dumber; to do less, but faster), but this doesn't address the fundamental issue that failed tests produce no output, yet consume >99% of the processing time. Some projects have produced fast parsing by separating the semantic analysis into a separate module. However, the semantic analysis that follows is subjected not only to the same combinatorial explosion, but the explosion is made worse by having to

reassemble components of semantic units separated during parsing.

A parallel problem existed in the early days of computers, when programs had to wait for their own I/O. Buffered I/O eliminated this problem, and programs ran much faster.

3 The Concept

This article explores a new method which nearly eliminates the time costs of failed tests, thereby speeding up the parsing of NL by several orders of magnitude. This is accomplished by only performing rules whose least likely elements are present, and doing this in a way that incurs zero overhead for rules whose least likely elements are absent. This presents a new problem – of coordinating the complex process of parsing once most of the "structure" has been selectively eliminated. This problem is overcome by creating a new structure to queue the evaluation of surviving rules having least likely elements present, and to coordinate the order or evaluation, linking of rules, scope issues, etc.

4 Compared to Other Methods

Since I first proposed this method, various people have come forward with statements like "I thought that _____ (fill in the blank with their favorite parsing method) was fastest." Methods that involve parsing character strings, rather than first converting words to ordinals, have a built-in opportunity to run more than an order of magnitude faster, by simply converting words to ordinals and performing the same analysis using integer operations on ordinals that represent entire words. Some methods, like recursive ascent-descent parsing, can be directly emulated on this platform at speeds that are orders of magnitude faster. Some methods, like those using left associative grammars, discard semantic unit information to run fast. Semantic units must be subsequently reconstructed before parsing can be useful. This reconstruction requires performing the same tests as required by other methods, >99% of which can be eliminated using the method described below.

5 The Method

During initialization the first few thousand most commonly used words are processed into the lexicon, so that later, when rules are being compiled, it will be easy to compare words in rules to identify which are least frequently used.

All input is first broken into tokens representing a word, number, or punctuation. The word tokens are then hashed as part of being converted to ordinals representing their frequency of use, e.g. the most common word in English is "the", which would be represented as 1.

Stored in the lexicon is the hash for the words (used to resolve collisions), the word strings (used to create output), and a list of pointers to the rules for which that word is the least frequently used (LFU) word in the rule.

Rules contain a compiled representation of their operation, and information regarding their depth (to place them into the correct queue) and scope (to restrict analysis to the appropriate syntactical unit, e.g. a sentence).

During execution, words are converted to ordinals. In the process, pointers to the rules for which those words were those rules' LFU words are placed into appropriately prioritized queues that also hold the locations of the associated words. Pointers to lower-level rules are placed in higher-priority queues, to perform the lower-level rules first, before performing higher-level rules that were placed in lower-priority queues.

String operations, e.g. as used to parse common German run-on words, are performed on the strings in the lexicon during initialization to add new rules to existing lexicon entries to process the substrings. String operations can be performed during execution, which can be useful for handling invented words. However, substrings cannot be used as LFU triggers, although they can be used in rules that have been triggered by other words or rules.

The vast majority of low-level rules will never be performed in any given passage because their LFU words will not be present. Only the higher-level rules that were referenced by successfully-performing lower-level rules will be performed. Those few higher-level rules that are performed will often reference lower-level rules that haven't been performed. Rules that have not been performed can safely be presumed to be FALSE, because they must lack their least likely elements to not have been performed.

It is hard to guesstimate the speedup that this will provide, partly because there are numerous other methods with which to compare it, and partly because no one has seriously attempted to enter the rules to fully understand any language. However, the range seems to be somewhere around 3-4 orders of magnitude improvement in speed.

6 Path Toward Universal Acceptance

The next step is to propose a broadly usable and easily extensible representation standard for parsing, disambiguating, and other rules, which will be needed to construct a software platform on which to build many products that involve understanding NL.

NLP Data Structure
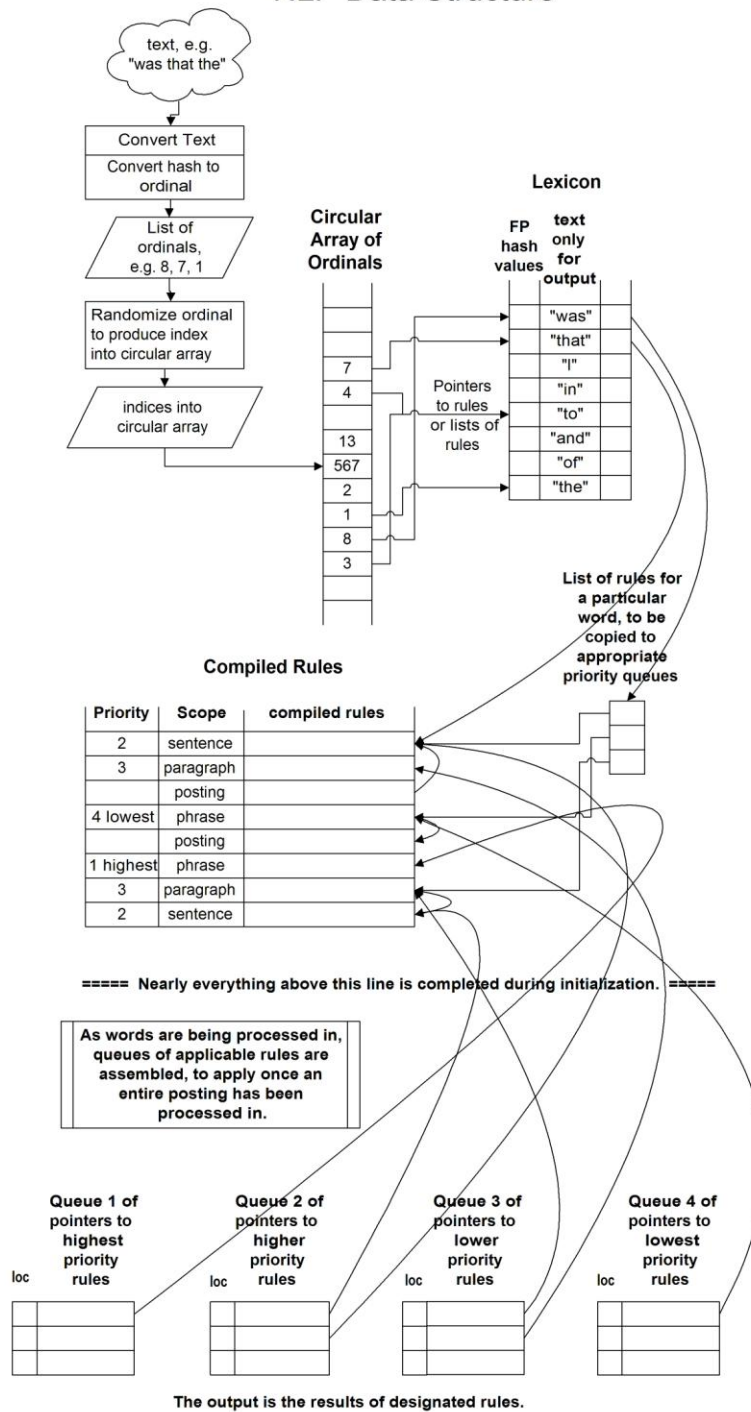
Figure 1

However, by far the most labor intensive step lies beyond; to invest the large amount of linguistic work needed to precisely define the parsing of various world languages as they are written and spoken "in the wild".

6.1 Representation beyond BNF

Most of the thousands of rules would be written by linguists, not programmers. It would seem that a dual form of representation should be supported, where rules could be entered in either of two forms.

1. Grammatically correct sentences in any of several supported natural languages, inspired by the NL statement syntax used in COBOL.

2. Algebraic representation, which would be a major extension of Backus-Naur Form (BNF).

Syntax specification has previously been all about input, but also needed is a canonical way of specifying output to disparate applications, perhaps akin to that used in some meta-compilers.

6.2 Need for Coordination

There are several VERY different uses for NL parsing, each with its own special problems. For example DrEliza.com (which I wrote) has no provision for being able to output any of its own input, as there is no need for this in its application.

I have had experience writing commercial production compilers, source language optimizers, a linker, and worked in teams on computer-language related products, including the vectorizer and optimizer for CDC's supercomputer compiler. However, methods used to process computer languages are broadly inapplicable to efficiently processing NL.

The goal of automated language translation programs is to output (with suitable translation) EVERYTHING that is input, which is somewhat analogous to a compiler that targets a computer having a hyper-complex instruction set, but lacks operations that perform the precise functions of the operators in its input syntax.

Computers faced a similar challenge during the 1970s and 1980s. Thousands of computer programs in various computer languages running on 36-bit and longer CISC architectures, like IBM-7094 and Burroughs 6000 processors, were being converted to different computer languages running on simpler 32-bit architectures like IBM-360 and Intel processors that cost far less because they were made with far fewer transistors. Now, transistors are essentially free, so long word length CISC architectures should be revisited. That is another story for another paper. A number of automated translation tools were developed that operated similarly to tools Microsoft later offered to assist in converting programs to their simplified .NET platform. These translation programs left a lot of residual problems, not unlike those left by modern day automated language translation programs. Computers are MUCH less flexible in interpreting their programs than people are in reading text, so a higher state of perfection was needed in translating computer programs compared to translating NL.

To succeed, experienced automated language translation personnel would of necessity be participating in any effort to design a universal NL platform.

A working group is needed to define the syntax and operation of parsing rules for both input and output that will work for everyone, populated by people with sufficiently disparate backgrounds. The goal would be to produce a "drop in" module that will work for any NL application, ranging from problem solving and question answering, to automated language translation.

This module would support application-specific rules, backed up by a vast library of rules to take care of the myriad details of parsing NL, e.g. how to handle thousands of idioms, each with their own peculiarities. Using the method described above, including multiple languages in the library would have a negligible affect on performance, e.g. only where there are multiple identical words in different languages that have different meanings but which may be used together. Language-related disambiguation would be necessary only in these rare cases. This is so rare that I know of no such example to cite here, so the cost in time would be negligible.

6.3 Placement/Payload Theory

It is my theory that computerized speech and written understanding has eluded developers for the past ~40 years in part because of a lack of a fundamental understanding of the task, which turns out to be very similar to patent classification.

When classifying a patent, successive layers of sub-classification are established, until only unique details distinguish one patent from another in the bottom-level subclass. When reviewing the sub-classifications that a particular patent is filed within, combined with the patent's title, the subject matter usually becomes apparent to anyone skilled in the art.

However, when a patent is filed into a different patent filing system, e.g. filed in a different country where the sub-classifications may be quite different, it may be possible that the claims overlap the claims of other patents; and/or unclaimed disclosure would be patentable in a different country.

Similarly, when you speak or write, in your own mind most of your words are there to place a particular "payload" of information into its proper context, much as patent disclosures place claims into the state of an art. However, your listeners or readers may have a very different context in which to file your words. They must pick and choose from your words in an effort to place some of your words into their own context. What they see as the "payload" may not even be the payload you intended, but may be words you only meant for placement. Where no placement seems possible, they might simply ignore your words and file **you** as being ignorant or deranged.

The expressed relationship between various placement and payload words carries the author's point of view. For some applications, like automated language translation, this may be important to extract and preserve, while it is best ignored when solving problems, except when containing statements of ignorance. For example, take the statement "*I have a headache because I got drunk last night.*" The facts are "*I have a headache*" and "*I got drunk last night*". There is also a suspicious "*because*" relationship between those two facts. Most headaches are secondary to dehydration, especially those related to prior alcohol consumption. Hence, the primary reason the author of this statement has a headache is probably because he has not consumed enough water, and **not** because of the lesser contributing factor of having gotten drunk last night. In this context, "*because*" becomes a statement of ignorance and hence potential payload.

Many teachers have recorded a classroom presentation and transcribed the recording, only to be surprised to learn that what they actually said was sometimes the opposite of what they meant to say. Somehow the class understood what they meant to say, even though their statement was quite flawed. When you look at these situations, the placement words were adequate, though imperfect, but the payload was okay. Indeed, if another person's world model is nearly identical to yours, very few placement words are needed, and so these words are often omitted in casual speech which complicates translation.

These omitted words fracture the structure of about half of all sentences "in the wild", often rendering computerized parsing impossible. If a computer program first identifies

prospective payloads, and then looks for nearby placement information while ignoring things it can't deal with, then fractured sentences only cause difficulty when the fractures are critically located.

As people speak or write to a computer, the computer must necessarily have a **very** different point of view to even be useful. The computer must be able to address issues that you cannot successfully address yourself, so its knowledge must necessarily exceed your own in its subject domain. This leads to some curious conclusions:

1. Some of your placement words will probably be interpreted as "statements of ignorance" by the computer, and so be processed as valuable payload, to trigger an appropriate response to teach you something you clearly do not know.
2. Some of your placement words will probably refer to things outside of the computer's domain knowledge, and so must be ignored, other than being recognized as non-understandable restrictions on the payload, that may itself be impossible to utilize.
3. Some of your intended "payload" words will serve as placement.

DrEliza's application seeks to intercept words written to someone who presumably has substantial common domain knowledge. Further, the computer seeks to compose human-appearing responses, despite its necessarily different point of view and lack of original domain knowledge. While this is simply not possible for the vast majority of writings, DrEliza can simply ignore everything that it is unable to usefully respond to.

If you speak a foreign language, especially if you don't speak it well, you will immediately recognize this situation as being all too common when listening to others with greater language skills than your own speaking among themselves. The best you can do is to quietly listen until some point in the conversation when you understand enough of what is being said and you are able to add something useful to the conversation.

Note the similarity to advertising within present (2013) Google Mail, where advertisements are selected based upon the content of email. If Google's computers were to perform a deeper analysis they could probably eliminate ~99% of the ads as not relating to users' needs and greatly improve users' experience, and customize the remaining 1% of the ads to precisely target users' interests.

That is very much the goal in my application. The computer knows about certain products and solutions to common

problems, etc., and scans the vastness of the Internet to find people whose words have stated or implied a need for things in the computer's knowledge base, and have done so in terms that the computer can "understand".

6.4 Representation Implications

When advanced compilers, especially optimizing and vectorizing compilers for supercomputers, compile computer programs to executable code, they make no attempt to translate program statements one-at-a-time. Instead, they build a diagram of the entire program, simplify and otherwise improve the diagram, and then translate the diagram to executable code. The same could be done with human speech and writing, in which case the output order would often be rearranged from the input order, in ways where statements would be clear and direct. Outputting in the same language, such a program might make a good automated editing program.

Several complex NL understanding project proposals have incorporated some sort of an a priori world model, onto which they plan to hang information gained from their input. Often the world model is simply built into the ontological information about individual words. It is unclear whether such a priori structures are a help or a hindrance, especially if there is some efficient way to find related statements without committing to file facts into a particular knowledge structure.

Foregoing an a priori world model requires a simple representation in which to state and store all of human knowledge. Presuming the placement/payload theory is correct, statements would be represented as groups of information fragments, some placement, and some payload, originally depending on what the author already knows. The computer would have the job of processing and/or outputting these fragments of information, without knowing which was placement and which was payload.

Most applications, from query to translation, require some way of accessing statements relating to particular collections of information fragments. One can envision contorted table structures and recursive descent searching algorithms to find the statements that best address particular subjects, but it would be preferable to simply have an efficient database tool like SQL do the job for us.

6.5 An Issue with SQL

SQL is my favorite AI programming language, because I can do really complex information accessing with single statements. However, it is powerless to access records that have particular contents, where those contents could appear in any field of a record.

Present SQL products are unable to "wildcard" subfields in an index, both in the records themselves and in the **SELECT** statements that access records. Complex kludged workarounds, e.g. using string operators in **SELECT** statements, indirectly force the SQL engines to perform sequential searching. These kludges slow things down so much that they are only practical for small demos. Other kludges in effect overlay wildcard accessing over conventional fully-specified keyed accessing, resulting in slow programs that are cluttered with the code needed to make this work.

Some SQL products have sophisticated string search capabilities, but are slowed down by their string operation, when AI programs work better with ordinals instead of strings.

Some future advanced NL understanding projects may have to wait for an SQL-like product that supports some sort of new capability to index through groups of information fragments.

This facility will never appear in SQL until someone specifies a particular new SQL capability that would facilitate these applications. Then, someone can add the new capability to one of the shareware SQL products, so that NL understanding can proceed without this challenge hanging over it.

7 Conclusion

Now that you understand the "logic" that has misguided so many NL understanding projects onto the scrap heap, you can quickly recognize it when you see it again in the future, and explain the pitfall that awaits such efforts. The answers to simple questions like "How are you going to represent words?" and "What triggers the evaluation of a rule?" will usually tell you if the projects are on the wrong track.

This new method promises several orders of magnitude improvement in the speed of NL understanding, regardless of which model of language is being used. This method has its own characteristic strengths and weaknesses around which a robust rules compiler could compile rules suitable for just about any imaginable approach to NL understanding.

However, it is one thing to "understand" NL, and quite another to usefully manipulate it, e.g. mine it for knowledge or translate it to a foreign language. Some sort of new database capability appears to be needed to replace

present ad hoc methods with high-level query-driven database solutions.

If this technology and associated descriptions of common languages are to be shared then some standards are necessary. If you would like to participate in developing a robust representation and interface to support your own needs for NL understanding, or if you just want some tables (like the 10,000 most commonly used English words in order of frequency of use), then please contact me at:

Steve.Richfield@gmail.com

8 The Future

It is hoped that this methodology will affect the world in three important ways:

1. People will stop writing NL understanding code that has no real possibility of ever scaling up to a useful level of functionality.

2. The Internet in general, and Google in particular, will shed its dependence on isolated word and n-gram recognition for web searching and advertisement triggering, and shift to looking for statements having specific meanings.

3. Future advertising engines that will watch the Internet for problem statements to trigger advertisements; will also be able to watch the Internet for problems statements relating to health, maintenance, politics, and other interesting domains. Then, precisely targeted responses can be produced to convey key knowledge, to finally achieve the goal of having an *Intelligent Internet*.

9 References

[1] U.S. Patent Application 13/836,678.

Meaning Judgment Method for Alphabet Abbreviation Using Wikipedia and Earth Mover's Distance

Seiji Tsuchiya[1]**, Misako Imono**[2]**, Eriko Yoshimura**[1]**, and Hirokazu Watabe**[1]
[1]Dept. of Intelligent Information Engineering and Science, Faculty of Science and Engineering,
Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2]Dept. of Information and Computer Science, Graduate School of Engineering,
Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *Recently, not only the person and things, but also words are imported by internationalization and informationization, and the scene using the loanword is increasing. However, these expressions are hard to understand for a child and the elderly person. Therefore, when such an expression is used for sentences, it might be hindered to understand entire sentences. Moreover, because an original word is omitted, it is likely to become the same expression as other words. Therefore, the alphabet abbreviation often has the polysemy. In this paper, a method of extracts an alphabet abbreviation from a sentence and judges the meaning of the expression is proposed. This method selects a correct meaning from two or more meanings of the alphabet abbreviation that suit for sentences, judging by using Wikipedia and Earth Mover's Distance. Moreover, a correct meaning is judged by the association mechanism, using an original knowledge base that defines the concept of the word. The accuracy of the proposed method was 74%.*

Keywords: alphabet abbreviation, Wikipedia, Concept Base, Degree of Association, Earth Mover's Distance (EMD)

1   Introduction

Recently, not only the person and things, but also words are imported by internationalization and informationization, and the scene using the loanword is increasing. The loanword is often used as the alphabet expression and the katakana expression, etc. in Japan. However, these expressions are hard to understand for a child and the elderly person. Therefore, when such an expression is used for sentences, it might be hindered to understand entire sentences. Especially, the alphabet abbreviation is the classic example. For instance, there is an expression by "IC". It is used when the word is omitted, and composed by initial of a certain word. If original word of the alphabet abbreviation is not understood, the expression is not understood though such an alphabet abbreviation is convenient. Moreover, because an original word is omitted, it is likely to become the same expression as other words. Therefore, the alphabet abbreviation often has the polysemy. Previous example, "IC" has two or more

meanings such as "Integrated circuit" and "Interchange in the expressway", etc.

In this paper, a method of extracts an alphabet abbreviation from a sentence and judges the meaning of the expression is proposed. This method selects a correct meaning from two or more meanings of the alphabet abbreviation that suit for sentences, judging by using Wikipedia and Earth Mover's Distance. Moreover, a correct meaning is judged by the association mechanism, using an original knowledge base that defines the concept of the word.

2   Proposed Method and   Elemental Technique
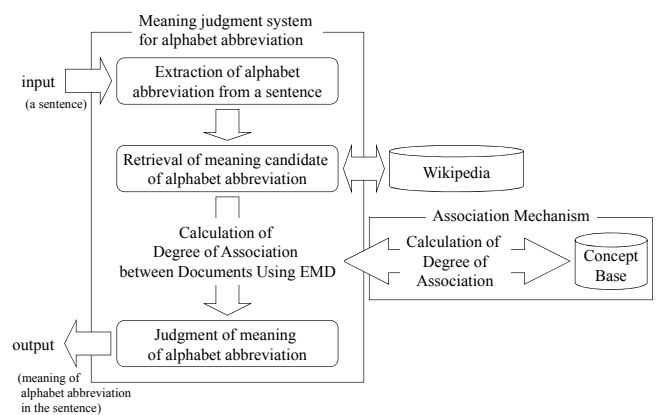
Fig. 1 Outline of the proposed meaning judgment method for alphabet abbreviation

Figure 1 shows the outline of the proposed meaning judgment method for alphabet abbreviation. When a sentence is inputted an alphabet abbreviation is extract which included in the sentence. The extracted alphabet abbreviation is retrieved with Wikipedia [1], so the original word of the alphabet abbreviation is obtained. The original word of the alphabet abbreviation judges the word that suit for sentences by evaluating the relativity of input sentences and Wikipedia's explanation sentences of those words in two or more cases. The relations between the sentences are evaluated by technique of Earth Mover's Distance. And, the relations

between the words are evaluated by the idea of the Concept Base [2][3] that defines the concept of the word and the idea of the Degree of Association [4][5] that calculates the association between words. A lot of words like the proper noun etc. are not existed in the Concept Base because the Concept Base is constructed by using the national language dictionary. However, it is necessary to conceptualize the word to calculate the Degree of Association, so the word that doesn't exist in the Concept Base is automatically conceptualized by using information on Web.

2.1 Concept Base

A Concept Base is a large-scale database that is constructed both manually and automatically using words from multiple electronic dictionaries as concepts and independent words in the explanations under the entry words as concept attributes. In the present research, a Concept Base containing approximately 90,000 concepts was used, in which auto-refining processing was carried out after the base had been manually constructed. In this processing, attributes considered inappropriate from the standpoint of human sensibility were deleted and necessary attributes were added.

In the Concept Base, concept A is expressed by attributes a_i indicating the features and meaning of the concept in relation to a weight w_i denoting how important an attribute a_i is in expressing the meaning of concept A. Assuming that the number of attributes of concept A is N, concept A is expressed as indicated below. Here, the attributes a_i are called primary attributes of concept A.

$$A = \{(a_1, w_1), (a_2, w_2), ..., (a_N, w_N)\} \qquad (1)$$

Because the primary attributes a_i of concept A are taken as the concepts defined in the Concept Base, attributes can be similarly elucidated from a_i. The attributes a_{ij} of a_i are called the secondary attributes of concept A. Figure 2 shows the elements of the concept "train" expanded as far as the secondary attributes.

Fig. 2 Example demonstrating the concept "train" expanded as far as secondary attributes

2.2 Methods of Automatically Expanding the Concept Base

If terms that are not in the Concept Base (undefined terms) are not given attributes, it will not be possible to seek the Degree of Association between undefined terms and other terms. For this reason, we propose a method of

conceptualizing undefined terms based on Web data, which currently the largest collection of linguistic data, and then adding these to the Concept Base.

2.2.1 Making Concepts of Undefined Terms

In order to conceptualize an undefined term, the attributes and weighting of the term are acquired from the Web using the procedure described below:

(1) A search engine is used to search for the entered but undefined term as a key word and obtain the content of the "top 100 search results" page.
(2) A morphological analysis is then applied to the document collection, and unnecessary data, such as HTML tags, will be removed and independent terms extracted.
(3) From among the independent terms obtained, only those that exist in the Concept Base are extracted as the attributes of undefined terms.
(4) The attribute frequency is multiplied by the SWeb-idf value, a statistically investigated idf of terms on the Web, and the value obtained is set as the attribute weighting. These are rearranged in order of the weighting. SWeb-idf will be explained in the following section. Attributes that do not exist in the SWeb-idf database are considered as terms that do not exist much on the Web, so they were multiplied using the maximum value of SWeb-idf.

2.2.2 SWeb-idf

SWeb-idf (Statics Web-Inverse document frequency) is an idf value that statistically examines the idf of terms on the Web. First, it generates 1,000 proper nouns that are randomly chosen. A search was carried out for each of the 1,000 terms created and the content of the top ten search result pages was obtained for each single term. As a consequence, the number of search result pages amounted to 10,000. Because we were able to obtain a number of terms that amounted to about the same 90,000 terms contained in the Concept Base from these 10,000 pages, which is a knowledge base that extracted the concepts (terms) from sources such as multiple Japanese language dictionaries and newspapers, we considered the 10,000 pages to be the information space for all data on the Web. SWeb-idf, which expresses the idf value of the terms within those pages, can be found using equation 2 below.

$$SWeb-idf(t) = \log \frac{N}{df(t)}, \quad (N=10000) \quad (2)$$

The terms and idf values obtained from this were registered in the database. The $df(t)$ part of the equation is the number of concept t pages that appear within all of the document spaces (10,000 pages).

2.2.3 Weighting Method Using Frequency of Appearance within the Attribute

Although the weighting of an attribute of an undefined term can be found using SWeb-idf, a distortion of the Concept Base frequency data will occur if the Web data weighting is used as is and added to the Concept Base. This is because the frequency data for terms differ in the Web data and the Concept Base and their weighting values change. As a result, it is only used when SWeb-idf obtains an undefined term's attribute candidate and not used for the attribute a weighting of the undefined term. Thus, when an undefined term is added to the Concept Base the frequency data of the Concept Base must be used to assign a weighting. This is why we propose a weighting method that takes the Concept Base attribute space into consideration as a means of assigning a weighting to the attribute of an undefined term. Since the attribute assigned to a concept is a term that expresses characteristics, it can be understood as being the explanatory text of that concept. The frequency of appearance of an attribute in this document space is considered the probability of the attribute relative to the concept.

It is possible to see the n order attribute space for the concept as a set of explanatory text for the concept. The frequency of appearance calculated from this n order attribute is called the frequency of appearance within the n order attribute. In this paper, the secondary attribute space is used. Based on the thinking behind the weighting of the tf-idf, if the frequency of appearance of the secondary attribute of undefined term attribute A is $freq(A)$, the total number of undefined term primary attributes is R, the idf value of the Concept Base space for undefined terms is $cidf(A)$, then the weighting $wc(A)$ can be expressed as shown in the following equation:

$$wc(A) = \frac{\log(freq(A))}{\log(R)} cidf(A) \qquad (3)$$

2.3 Calculating of the Degree of Association

For concepts A and B with primary attributes a_i and b_i and weights u_i and v_j, if the numbers of attributes are L and M, respectively ($L \leq M$), the concepts can be expressed as follows:

$$A = \{(a_1, u_1), (a_2, u_2), ..., (a_L, u_L)\} \qquad (4)$$

$$B = \{(b_1, v_1), (b_2, v_2), ..., (b_M, b_M)\} \qquad (5)$$

The Degree of Identity $I(A, B)$ between concepts A and B is defined as follows (the sum of the weights of the various concepts is normalized to 1):

$$I(A,B) = \sum_{a_i=b_i} \min(u_i, v_j) \qquad (6)$$

The Degree of Association is calculated by calculating the Degree of Identity for all of the targeted primary attribute combinations and then determining the correspondence between primary attributes. Specifically, priority is given to determining the correspondence between matching primary attributes. For primary attributes that do not match, the correspondence between primary attributes is determined so as to maximize the total degree of matching. Using the degree of matching, it is possible to give consideration to the Degree of Association even for primary attributes that do not match perfectly.

When the correspondences are thus determined, the Degree of Association $R(A, B)$ between concepts A and B is as follows:

$$R(A,B) = \sum_{i=1}^{L} I(a_i, b_{xi})(u_i + v_{xi}) \times \{\min(u_i, v_{xi}) / \max(u_i, v_{xi})\} / 2$$

In other words, the Degree of Association is proportional to the Degree of Identity of the corresponding primary attributes, and the average of the weights of those attributes and the weight ratios.

2.4 Degree of Association between Documents Using EMD

When seeking the degree of similarity between a search request and a search target, no matter how accurately the relevance between terms can be defined, if the calculation cannot take place based on the values, it will be impossible to find the precise degree of similarity between the documents. A variety of methods can be used for the calculation. For instance, one method would be to perform the calculation by correlating the terms in order beginning from the highest degree of relevance between the terms. A method that involves a one-to-one correlation can only correlate to the smallest number of terms between the search request and the search target. For example, if the search request has three terms and the search target has 100 terms, 97 of the search target terms will not be subjected to calculation. Furthermore, it is believed that when performing the actual search, users will not enter many terms in the search request, so the assumption is that there will be a large difference in the number of terms in the search request and search target. Therefore, it is necessary to consider the importance of terms in the text and the relevance between them and to be flexible in handling M relative to N.

For this reason, the Earth Mover's Distance (EMD) [6], which has been drawing attention in the field of similar imagery searching, has been employed in this study as a method that calculates the degree of similarity between documents. The EMD is an algorithm that seeks the optimal solution for transportation costs in a transportation problem. As a result, if the weighting between the demand point and the supply point and the distance between these points are

defined, it can be used to solve any type of problem. By employing the EMD and taking the weighting of terms and the relevance between terms into consideration, correlation can be flexible and the degree of similarity between sentences can be found.

2.4.1   What is the EMD

The EMD is a distance scale that calculates by means of the Hitchcock transportation problem, which is one type of linear programming problem. Given two discrete distributions, it is defined as the minimum cost of converting one distribution to the other distribution. The transportation problem is the problem of solving transportation from the supply point to the demand point in order to satisfy the demand at the demand point at minimum cost.

When seeking the EMD, the two distributions are expressed as sets that have been assigned element weightings. If one of the distributions P is expressed as a set, the expression becomes, $P = \{(p_1, w_{p1}), \ldots, (p_m, w_{pm})\}$. Distribution P is currently expressed as having m number of characteristics. p_i represents the characteristics, while w_{pi} represents the weighting of the characteristics. In like manner, if the other distribution Q is expressed as a set, the expression becomes, $Q = \{(q_1, w_{q1}), \ldots, (q_n, w_{qn})\}$. As for the EMD calculation, even if the number of characteristics for both distributions differs, it has a characteristic that allows the calculation to take place. Let us assume that the distance between p_i and q_j is d_{ij} and the distance between all features is $D = [d_{ij}]$. If we assume the amount of transportation from p_i to q_j to be f_{ij}, the total amount of transportation becomes $F = [f_{ij}]$. Here, we will find the amount of transportation F, which creates the minimum cost function shown in equation 7, and calculate the EMD.

$$WORK(P,Q,F) = \sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij} f_{ij} \qquad (7)$$

However, when minimizing the above cost function, the following restrictions must be satisfied.

$$f_{ij} \geq 0, \ 1 \leq i \leq m, \ 1 \leq j \leq n \qquad (8)$$

$$\sum_{j=1}^{n} f_{ij} \leq w_{p_i}, \ 1 \leq i \leq m \qquad (9)$$

$$\sum_{i=1}^{m} f_{ij} \leq w_{q_j}, \ 1 \leq j \leq n \qquad (10)$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{p_i}, \sum_{j=1}^{n} w_{q_j}\right) \qquad (11)$$

In this case, we know that the amount of transportation in equation 8 is positive and we also know that transportation goes one way, from p_i to q_j. Equation 9 indicates that transportation cannot take place above the weighting of the transportation source p_i. Equation 10 indicates that acceptance cannot take place above the weighting of the transportation destination q_j. Finally, equation 11 indicates the upper limit of the total amount of transportation and is limited by the smaller of the sum total of either the transportation destination or transportation source. The EMD between distributions P and Q can be found as indicated below by using the optimal total amount of transportation F found under the limitations indicated above.

$$EMD(P,Q) = \frac{\displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} f_{ij}}{\displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}} \qquad (12)$$

The reason the optimal cost function $WORK(P, Q, F)$ is used as is here as the EMD is that the cost function depends on the sum total of the weighting of either the transportation source or the transportation destination. So, that influence will be eliminated by normalization.

2.4.2   Applying EMD to Document Search
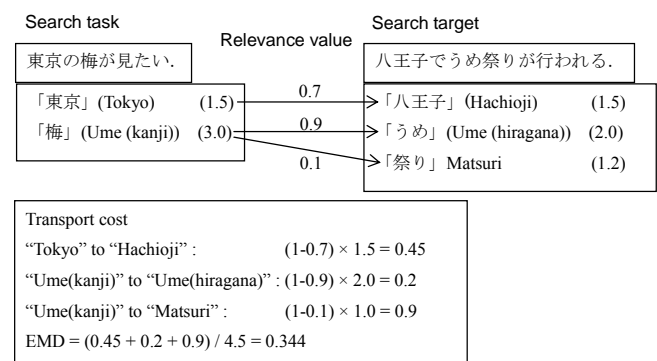
Fig. 3 Examples of Applying EMD to Document Search

Figure 3 shows examples of applying EMD to document search. To apply EMD to document search, the demand point and supply point, demand volume and supply volume, and the distance between each demand point and supply point must be defined. For the demand point, the index term for the search task is assigned, while for the supply point, the index term for the search result is assigned. The demand volume and supply volume each use the tf-idf weighting [7], which concerns index terms. The distance between the demand point

and supply point can be considered to be the relevance between index terms and, thus, can be found in the proposed methodology by the Degree of Association that uses the Concept Base. Since the value of the Degree of Association will be larger as the relevance increases, it will be converted into a value in which the Degree of Association value will be subtracted from 1. The calculation of EMD is located at the bottom of Figure 3. The reason that the amount of transportation between "ume" and "matsuri" is 1 is because a weighting of 2 was transported from "ume (kanji)" to "ume (hiragana)" and the excess weighting of "ume (kanji)," 1, was transported to "matsuri. " The weighting is transported in this manner to terms with a high degree of relevance and the transportation will take place until the supply volume disappears or the demand volume is satisfied. In this way, a flexible M versus N that considers relevance and weighting between index terms is possible. As a characteristic of the EMD, if the value of the distance between index terms is from 0 to 1, then EMD also becomes a value from 0 to 1. Additionally, if there is similarity between documents, the value falls, and if there is a lack of similarity, the value rises. Thus, document retrieval is realized by presenting documents to the user in sequence beginning with documents with low values.

3   Meaning Judgment Method of Alphabet Abbrebiation

The proposed meaning judgment method is composed of three processing as shown in Figure 1: extraction of alphabet abbreviation from a sentence, retrieval of candidate meaning of alphabet abbreviation and judgment of meaning of alphabet abbreviation.

3.1   Extraction   of   alphabet abbreviation from a sentence

The rule of extracting the alphabet abbreviation from sentence is three of the following:

(1)   The string of alphabet character composed of the capital letter and the small letter of one character or more, and it is extracted from sentence.

(2)   The string of character that there is a figure after the alphabet character string is extracted. For instance, correspond to "CO2" etc.

(3)   The string of character which including the sign is not extracting to the string of alphabet character. For instance, correspond to "http://www" etc.

3.2   Retrieval of meaning candidate of alphabet abbreviation

The alphabet abbreviation extracted by the rules of section 3.1 is retrieving with Wikipedia, and the original word is obtained from candidate of the word (meaning).

When the candidate meaning is one, the word and the detailed explanation that becomes the original word of the alphabet abbreviation are described in Wikipedia. Accordingly, the word that becomes the original word is a meaning of the alphabet abbreviation in this case.

On the other hand, the words that become two or more original words are enumerated in Wikipedia when there are two or more meaning candidates. Therefore, all enumerated words are acquired as a meaning candidate in this case.

3.3   Judgment of meaning of alphabet abbreviation

When two or more meaning candidates are acquired by processing of section 3.2, it is necessary to judge a correct meaning from several meaning candidate. A correct meaning is judged from the evaluation of meaning association between an input sentence and the meaning candidates. The meaning candidates are expressed as explanation sentences in Wikipedia. The Concept Base and the method of calculation of Degree of Association explains in Chapter 2 are used for the evaluation of meaning association.

The words included in the explanation sentences of the meaning candidates are conceptualized by using the attributes which exist and defined in the Concept Base. It is explained in section 2.1. When the words included in the explanation sentences of the meaning candidates do not exist in the Concept Base, they are conceptualized by the explained method in section 2.2. Moreover, the words included in the input sentence are similarly conceptualized. In addition, the weights of words are given by tf-idf technique [7].

The association between the words included in the explanation sentences of the meaning candidates and the words included in the input sentence are calculated by the method of the calculating of the Degree of Association explained in section 2.3. And, the association between the explanation sentences of the meaning candidates and the input sentence is calculated by the method of the EMD explained in section 2.4 using results of the association between words. As a result, a correct meaning of the alphabet abbreviation is decided by strongest relation between the explanation sentences of the meaning candidate and the input sentence.

4 Performance Evaluation of the Proposed Meaning Judgment Method for Alphabet Abbreviation

100 newspaper articles that contained the string of alphabet character were selected at random and used for evaluation data. A correct meaning of the alphabet abbreviation was judged by three subjects. When the processing explained by section 3 was done to the 100 newspaper articles, the alphabet abbreviations of 54 words with two or more meanings have been extracted.

Figure 4 shows the comparative results of proposed method using EMD and method no using EMD. The accuracy of the proposed method was 74%. The accuracy of proposed method using EMD was improved 12% to compare with the method no using EMD. By these results, the proposed method is able to say effective.
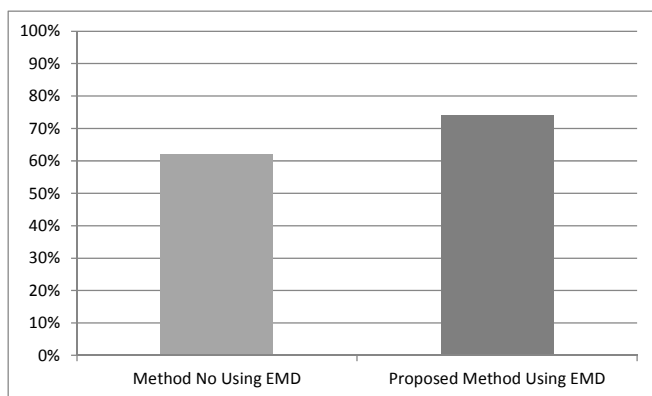
Fig. 4 Result of the proposed meaning judgment method for alphabet abbreviation

5 Conclusions

A method which extracts an alphabet abbreviation from a sentence and judges the meaning of the expression was proposed in this paper. This method selects a correct meaning from two or more meanings of the alphabet abbreviation that suit for sentences, judging by using Wikipedia and Earth Mover's Distance (EMD). Moreover, a correct meaning was judged by the association mechanism using an original knowledge base that defined the concept of the word.

100 newspaper articles that contained the string of alphabet character were selected at random and used for evaluation data. The alphabet abbreviation of 54 words was extracted from 100 newspaper articles. A correct meaning of the alphabet abbreviation was judged by three subjects, and the accuracy of this method was 74%. By the result, we believe that the proposed method is effective.

References

[1] Wikipedia, http://ja.wikipedia.org/wiki

[2] Kojima, K., Watabe, H. and Kawaoka, T.: A Method of a Concept-base Construction for an Association System: Deciding AttributeWeights Based on the Degree of Attribute Reliability. Journal of Natural Language Processing. 9(5), pp.93–110, 2002.

[3] N. Okumura, E. Yoshimura, H. Watabe, and T. Kawaoka, "An Association Method Using Concept-Base", KES 2007/WIRN2007, Part I, LNAI4692, pp.604–611, 2007.

[4] H. Watabe and T. Kawaoka: "The Degree of Association between Concepts using the Chain of Concepts", Proc. of SMC2001, pp.877-881, 2001.

[5] Hirose, T., Watabe, H. and Kawaoka, T.: Automatic Refinement Method of Concept-base Considering the Rule between Concepts and Frequency of Appearance as an Attribute. Technical Report of the Institute of Electronics, Information and Communication Engineers. NLC2001-93, pp.109–116, 2002.

[6] Y. Rubner, C. Tomasi, and L. Guibas: "The earth mover's distance as a metric for image retrieval", Int. J. Comput. Vision, Vol. 40, pp. 99–121, 2000.

[7] Salton, G. and Buckley, C. , "Term-weighting approaches in automatic text retrieval", *Information Processing & Management*, Vol.41, No.4, pp.513-523, 1988.

Implementation of Intelligence Robotic Engines Using Ontologies and Process Natural Language

Rafael Luiz de Macedo, Elvis Fusco, and Fábio Dacencio Pereira
UNIVEM, University Center Eurípides of Marília, Marília, SP, Brazil

Abstract - *The AIML (Artificial Intelligence Markup Language) is a set of tags XML (eXtensible Markup Language) able to represent and relate expressions in natural language allowing the creation of robotic engines capable of maintaining a simple dialog, however, robotic engines are limited in the amount of questions they may respond, thus, failing to keep a simple dialogue for a long time. This article is intended to extend the AIML in incorporation of HTML5 elements and custom patterns using a Java interpreter. Through these proposed elements and with the aid of a source code Analyzer software developed semantic searches can be performed on the web using the sources of pages and, from them, the elements are parsed and extracted to a semantic treatment in extracted information, generating new information handled in the form of questions and answers for the robotic engines developed in AIML markup language.*

Keywords: AIML; Artificial Intelligence; Engines Robotic; Semantic Web.

1 Introduction

Current technologies have allowed more and more, the perception that it is possible to make a machine be intelligent enough to answer questions made through robotic engines built with artificial intelligence.

The language is intended to be used for communication on the world. When you learn the language, the more the world can test theories about the world to know that they give support to attempt to understand the language. And, if we can create a computational model of language, can be a powerful tool for communication on the world [8].

One of the languages that enables the creation and development of these robotic smart Motors is the markup language AIML (Artificial Intelligence Markup Languages). But, the AIML language has a limitation on the development of these engines, making it limited the number of questions that can be answered by these robotic engines.

With the emergence of the Internet, you can create new AIML tags interpreted with HTML5 standards and customized tags for semantic searches are carried out on the web. These searches will be made by means of the questions

that the robotic engines are not possible answer them. And, with a source code Analyzer software that makes extracting information from treatment Web sites found on the Internet and turning these new information on new questions and answers to enhance and extend the knowledge of these robotic intelligent engines.

The purpose of this article is to extend markup language AIML AIML, with new elements from the HTML5 tags and embedding of parser software development able to make the treatment of the information extracted from the source code of websites, thus generating new AIML language information.

To search and interpretation of information on the WEB, it is necessary to use concepts of search and natural language interpretations. In particular, ontology and Natural language processing impose these rules.

2 AIML

The AIML language enables the creation of robotic conversational engines called Chatterbots (Chat = Chat and bot = robot), a robot with Artificial Intelligence that interacts with users, through conversations, causing a human thinks is "talking" with another human.

One of the best-known software in natural language processing is the A.L.I.C.E. (Artificial Linguistic Internet Computer Entity). With the development of the A.L.I.C.E. language, resulting in the AIML markup language. AIML is license under the GNU GPL and there are AIML interpreters available in Java, Ruby, Python, C++, c #, Pascal, and other languages.

The language is a set of AIML tags XML (eXtensible Markup Language) capable of representing and linking natural language expressions, allowing the creation of engines able to keep a simple dialog. Each AIML tag set has one or more tags category calls, and the categories are developed upon a context; a category is formed by all the tags and template pattern, which are the tags responsible for interpreting the message entered by the user and send a reply message back to the user.

The AIML is divided into two modules, language and motor. The language is where all knowledge of natural

language and information developed in AIML language, and the motor makes communication between natural language and markup language AIML, however there is a limitation in your modules. The development of AIML is related to a context, making it limited its application. So, in a natural language processing system using the AIML, the conversation does not remain for a long time in a dialogue because it is lost in a question out of context that was developed.

Alan Turing (1912-1954), in his famous essay "Computing Machinery and Intelligence" (Turing 1950), suggested that, instead of asking whether machines can think, we must ask if machines can pass a test of behavioral intelligence, which came to be called Turing test [8].

The markup language AIML is an open source language, thus enabling the use of language in the search for improvement and creation of intelligent robotic conversation engines. Alan Turing proposed that, if it is not possible to distinguish a machine answers those provided by a person, the machine could be considered smart and, to the present day, there are no machines capable of going for this Turing test.

To create robotic engines, can be used a specific platform and, in a markup language, there are many platforms and languages available for the creation of these robotic engines.

In this article, is being used an interpreter in the Java programming language and platform ProgramD, which is a Open Source platform.

The ProgramD platform is the most widely used platform in the development of robotic intelligent engines in AIML markup language and the most complete in AIML language feature, in addition to being the best ever tested implementation of AIML.

3   HTML

With the advancement of Internet and browsers, HTML5 (Hypertext Markup Language) is a valid attempt to standardize how browsers interpret, classify and present the information received.

While the World Wide Web Consortium (W3C) focused their attention to the creation of the second version of XHTML, a group called the Web Hypertext Application Technology Working Group or WHATWG working on a version of HTML that would bring more flexibility to the production of websites and Web-based systems.

HTML5 is a markup language that facilitates the development of sites, because, with a few lines, it is possible to build a simple website and in addition enable plugins that facilitate the development of interfaces.

Taking advantage of this new standard of organization of information on the Web it is possible to increase the scope and ability of languages like the AIML. Currently, most Web sites is developed on top of HTML5 and the new elements created in the AIML language will make the interpretation of these HTML5 tags, performing semantic search on the Web to find sites in HTML5 developments.

4   Ontology

The ontology about the knowledge of be, studies the understanding of be. it creates various types of different domains, which each domain, is renowned on a particular subject, such as in a semantic search on the Web.

According to [4], ontology is a specification of a conceptualization, that is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents.

On the Web, ontology is used to organize a knowledge domain, to be able to model a knowledge. Example of use of the ontology is a semantic search on the Web, because it gets a more accurate search, knowing of knowledge related to the subject or the object being sought and thereby getting more accurate results.

The integration of the Semantic Web with the AIML language has the main goal of making the Web usage to enhance and embrace the AIML language and robotic engines that are created on top of that language, so that increases the number of questions that can be made to these robotic engines.

For that language AIML can accomplish these semantic searches, new elements will be developed AIML interpreted with HTML5 tags and custom patterns. To carry out these searches, however, it is necessary to obtain clear and precise information about the context of the questions that are made to these robotic intelligent engines. To this will be made to human questions and thus information about the context in which the question belongs.

Intelligent robotic engines developed in AIML markup language, to fail to answer any questions by not contain in its natural language information Bank, will hold two important questions so that it can be carried out searches fast and accurate web semantics. The two questions are: First-what is the name of the subject? Second-in a nutshell describes, with which it relates?

With the information obtained by these two questions in a nutshell semantic search can be performed on the web more quickly and accurately with the application of OpenCyc, which makes use of the concepts of the ontology.

5 OpenCyc

The OpenCyc platform is your gateway to the total power of Cyc, the world's largest and most complete general knowledge base and commonsense reasoning engine. OpenCyc contains hundreds of thousands of terms of Cyc, arranged in a carefully designed ontology. Cycorp offers this ontology at no cost and encourages you to use and extend, this ontology, instead of starting your own from scratch [5].

Cycorp is a leading provider of semantic technologies that bring a new level of intelligence and common sense reasoning to a wide variety of software applications. The software combines an incomparable sense Cyc ontology and knowledge base with a powerful motor of reasoning and natural language interfaces to enable the development of new applications of knowledge [5].

The Cyc was founded by Dr. Douglas Lenat, in the year 1984, your goal was as a lead project at microelectronics and computer technology.

The OpenCyc makes use of numerous fields of ontology and is an open source platform that helps and helps in mapping and information extraction.

Now it's even easier to use the rich and diverse collection of real-world concepts in OpenCyc to bring meaning to their semantic web applications! The entire contents of OpenCyc is now available both as downloadable OWL ontologies and semantic web endpoints (that is, permanent URIs). These URIs return RDF representations of each Cyc concept as well as a human-readable version when accessed through a Web browser [6].

The use of the OpenCyc assists in better absorption in the information found on web sites and in generating new questions and answers to intelligent robotic engines developed in AIML markup language.

6 Processing Natural Language

Natural language processing is the study of language, being spoken language or text written language. The study of language enables us to a better understanding of the language and, if the machines understand language, you can create intelligent robotic engines.

The written language makes use of lexical, syntactic and semantic knowledge of the language and also all the necessary information about the real world [7].

Natural language processing is divided into several components, each component has its function in natural language recognition. The components, Morphological Analysis, syntactic analysis, Semantic Analysis, integration of discourse and Pragmatic Analysis, are the Group of components that make up the body of natural language processing.

6.1 Morphplogical Analysis

Every word is analyzed separately and is made the recognition of same; the words are analyzed in terms of its components, and the signs, such as punctuation, are separated from the words [7].

6.2 Syntatic Analysis

Syntactic analysis is the recognition of a set of words to see if they are correctly linked. Some sequences of words may be rejected if you violate the rules of the language about how words can be combined. For example, a parser of Portuguese would reject the phrase: "the Boy goes to the store" [7].

6.3 Semantic Analysis

Is done a mapping in the words to make the recognition and see if the meaning of the set is correct. Each set of words has a semantic meaning, belongs to some specific context.

Semantic analysis is the recognition that context to which the set of words belongs and does a semantic mapping to see if is correct its meaning. A set of words that cannot be done the semantic mapping is rejected by the semantic Analyzer.

6.4 Integration of Speech

Is the meaning of a sentence depends on the previous phrases isolated for its meaning and influence in subsequent sentences. For example, the word "it", in the phrase "John wanted it" depends on the context of the previous speech; While the word "John" may influence the meaning of phrases, such as: "He always wanted to" [7].

6.5 Pragmatic Analysis

A pragmatic analysis makes the interpretation of the phrase, recognizing what it meant. The phrase may represent a request, an affirmation, a warning and other representations contained in natural language.

The phrase "you know what time is it?" should be interpreted as a request to which the hours are informed [7].

7 Robotic Motors Combined With Process Natural Language

Robotic Motors make the recognition of a set of characters of a natural language, that is, recognize words and phrases typed by users; However these robotic engines cannot understand the meaning of words or phrases typed, more clear and precise, cannot understand the semantics.

A robotic engine created in markup language AIML, contains a limitation on the recognition of these questions; You can only perform this engine have questions in your natural language information bank.

With the rules of natural language processing techniques and a specific domain of ontology, it is possible to develop a software source code Analyzer that makes a better understanding in natural language, generating new questions and answers to the robotic engines.

When a robotic motor can't answer a question, it will make use of the new elements of AIML markup language for semantic searches on the web, finding sites that contains information related to this question was made and with the integration of the software source code Analyzer applies a mapping and treatment information that is contained in the source code of the web site that was obtained in the search result to generate new questions and responses, in order to be able to answer the question that is not contained in your natural language information bank.

8   Related Work

The implementation of information mapping requires a great aid of the domain ontology concepts, in order to get a better absorption of the context in which this information belong. But, in related jobs realizes that it is too big the difficulty in obtaining a total absorption and in many cases it is not clear what this information is.

The concepts of the ontology are applied in several works of information mapping, one of those jobs that make use of these concepts of the ontology is the "Extracting Ontological Structure of the OpenCyc for Reuse and Portability of Cognitive Models" [4], which makes the mapping and information extraction using OpenCyc.

General-purpose knowledge ontologies, such as OpenCyc, have been suggested as a means of increasing the portability and reuse of cognitive models through a mapping onto domain-independent language. Previous efforts have revealed that this mapping process is difficult to perform due to several factors including the difficulty of understanding the underlying structure of the ontology and mismatches in representation between the target cognitive modeling architecture and the source ontology [4].

The approach of "Extracting Ontological Structure of the OpenCyc for Reuse and Portability of Cognitive Models" was divided into three stages, determining a suitable mapping, an extracted hierarchy and show the results.

The first step is the hardest step is the step which determines an appropriate mapping of specific terms, because a search is normally performed in a simple way, containing little information related to what is being sought and performing a search without much precision.

To perform a more accurate search, one would use the simple lookup as a starting point and dig for more specific constants. It is important to mention here that the full meaning of an OpenCyc term is best understood as a combination of 1) the name, 2) the related (more general/specific) terms, and 3) the "comment" tag associated with the term. [4]. After the implementation of the first stage, is passed to the second stage, extracted or pruning hierarchy hierarchy.

In the second stage is a pruning on information obtained on the web. Once the web of terms has been extracted from OpenCyc, some amount of pruning can be done; the level of pruning (or possibly expansion) depends highly on the intended use of the web. For instance, a web pruned from the root down to the most specific parent term (Lowest Common Genl or LCG) is a useful way to get an overall sense of the complexity and structure of OpenCyc [4].

With the exclusion of information that are not totally related to the subject that was sought, the amount of information that was obtained in the search, decrease and end up getting more accurate and thus can make a better mapping and information extraction. After the implementation of the second stage, will be held the third and last step, you will see the hierarchy.

In the third and final step is presented the results obtained after being performed a search on the web and mapping information for the exclusion of information unrelated to the search context. Hierarchically, is presented as the search result before you run a mapping and deleted some information and after the mapping is presented in the same hierarchical way the result.

The end result is obtained after the application of these three steps, but, with the implementation of the first two steps it is possible to obtain a more accurate result. In the first two steps, determining a suitable mapping and a hierarchy extracted, made use of the OpenCyc for application of the concepts of some specific fields of the ontology.

In this article, the form in which it is made, treatment and extraction mapping in the information found on the web sites used which were used for the development of source code Analyzer software, makes the use of the concepts of ontology, and also, the rules of natural language processing for better software development and recognition of natural language that the site belongs.

The form in which the work "Extracting Ontological Structure of the OpenCyc for Reuse and Portability of Cognitive Models" [4], applies the concepts of the ontology and the mapping in the information with the help of the OpenCyc is very rich in functionality, quality, get a quick and precise result.

Without the help of OpenCyc in software development source code Analyzer in this article, you can get an extraction of the information contained in the web sites.

9   Result

The software source code analyzer developed in this article, performs some analysis and certain treatments on top of the scenarios that were developed for this article.

The operation of the software source code Analyzer is divided by steps. In the first step, is passed around the site's source code found on a file already created by the parser, so that software, so it is possible to analyze and give the right treatment to the source code found. In Fig. 1, is shown as a file, after being passed the whole source code of the site found for the file.

```
<!DOCTYPE html>
<html>
<link      href="css.css"      rel="stylesheet"
type="text/css"/>
<body>                              I
<div id="total">
    <img    src="banner.gif"    alt="GaitoBot"
width="800"
height="160"/>
   <div id="conteudo">
      <h2><b>Abstract</b></h2>
      <div id="paragrafo">|
             <p class="intro">What is AIML? The
AIML (Artificial Intelligence Markup Language)
is  a  set  of  tags  XML  (Extensible  Markup
Language)  able  to  represent  and  relate
expressions  in  natural  language  allowing  the
creation  of  engines  capable  of  maintaining  a
simple dialog. This research project is intended
to  enhance  the  AIML  incorporating  standard
HTML5  TAGs  and  customized  using  a  java
interpreter.</p>
         </div>
      </div>
   </div>
</div>
</body>
</html>
```

Figure 1. Uploaded file with the source code of the site.

In the second step, the software analyzes all the file containing the source code of the Web site and apply the treatment; the parser makes the recognition of sets of words that do not belong to any programming language of Web sites and separating these sets of words into a new file in the order in which the words were found. All tags and commands that belong to any Web programming language sites are dropped by the software. In Fig. 2, is shown the result of the second stage, which is the file generated with the information found on the site.

Abstract

What is AIML? The AIML (Artificial Intelligence Markup Language) is a set of tags XML(Extensible Markup Language) able to represent and relate expressions in natural language allowing the creation of engines capable of maintaining a simple dialog. This research project is intended to enhance the AIML incorporating standard HTML5 TAGs and customized using a java interpreter.|

Figure 2. File with the information that has been extracted from the source code of the site.

In the third and final step, the parser software loads the file that contains the information taken from the Web site analyzed, that, with this file, applies a treatment of the questions and answers. The software analyzes the entire file looking for questions that may be directly or indirectly writes this information.

The questions and answers found by parser software are written to a new file along with AIML tags, so that the engine developed robotic AIML can make use of this file and, therefore, answer the questions that pertain to this file generated by the source code parser software. In Fig. 3 shows how an AIML file generated by the source code parser software.

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<aiml version='1.0.1' xmlns='http://alicebot.org/
2001/AIML-.0.1'xmlns:html='http://www.w3.org
/1999/xhtml'xmlns:xsi='http://www.w3.org/2001
/XMLSchema
instance'xsi:schemaLocation='http://alicebot.org/
2001/AIML-1.0.1http://aitools.
org/aiml/schema/AIML.xsd'>
<category>
    <pattern>What is AIML *</pattern>
    <template> O AIML (Artificial Intelligence
(Artificial Intelligence Markup Language) is a
set of tags XML (Extensible Markup
Language) able to represent and relate
expressions in natural language allowing the
creation of engines capable of maintaining a
simple dialog. This research project is intended
to enhance the AIML incorporating standard
HTML5 TAGs and customized using a java
interpreter.</template>
</category>
</aiml>
```

Figure 3. AIML file generated by the source code parser software.

The AIML file that was generated by the software source code Analyzer will be used by the robotic engine built in AIML markup language, to cover his knowledge with new questions and answers that contains this new file. ProgramD platform is being used, which is a free platform to test the AIML files generated by the Analyzer software.

10 Conclusion

AIML markup language has a limitation in creating robotic talk engines, making it limited the number of questions that may be made to these robotic engines. Due to this limitation, the robotic engines fail to keep a long conversation with a human, because questions made outside the context of knowledge in which these engines were developed cannot be answered.

For such a problem, it is proposed to extend the AIML with the incorporation of elements interpreted with standard and customized, HTML5 tag for semantic searches on the Web, using concepts of a specific domain ontology for a better understanding on the information containing in question held these robotic engines. With the use of the concepts of a domain of ontology, it will be possible to make a more specific search and get more accurate results in the context of the question.

The integration of these new AIML tags with the software source code Analyzer will make the use of natural language processing techniques, for the treatment of this information extracted from sites analyzed, transforming them into new

questions and answers to the markup language AIML knowledge and increase the number of questions that can be carried out for these robotic intelligent engines.

With the use of domain specific ontology concepts by the OpenCyc and technical rules for natural language processing, it is possible to obtain a better result in the generation of new questions and answers to these robotic intelligent engines.

11 References

[1] ALICE AI Foundation. "AIML: The Artificial Intelligence Markup Language" http://www.alicebot.org/aiml.html, May 2012.

[2] ALVES, Rachel C. V. (2005) "Web Semântica: Uma Análise focada no uso de metadados", Dissertação (Mestrado em Ciência da Informação) – Faculdade de Filosofia e Ciências – Universidade Estadual Paulista - UNESP, Marilia/SP.

[3] BUSH, Noel. "ProgramD" http://aitools.org/Programd, November 2012.

[4] BEST, Bradley J., Gerhart, Nathan, & Lebiere, Christian "Extracting the Ontological Structure of OpenCyc for Reuse and Portability of Cognitive Models", Proceedings of the 19th Conference on Behavior Representation in Modeling and Simulation, Charleston, SC, March 2010.

[5] Cycorp Inc. (2013), OpenCyc http://www.cyc.com/platform/opencyc.

[6] Cycorp Inc. (2013), OpenCyc for the Semantic Web, http://www.cyc.com/platform/opencyc/sw.

[7] Gruber, T.R. (1993) "Toward Principles for the Design of Ontologies Used for Knowledge Sharing", International Workshop on Formal Ontology, Padova, Italy.

[8] MENEZES, Paulo, B. (1998), "Linguagens Formais e Autômatos" Second Edition. Porto Alegre.

[9] O'Brien. P, Zhan C. (2000) "Domain Ontology Management Environment", Proceedings of 33rd Hawaii International Conference on Systems Sciences, January, Havaí.

[10] RICH, Elaine e KNIGHT, Kevin. (1993) "Inteligência Artificial" – Second Edition. São Paulo/SP.

[11] RUSSELL, Stuart e NORVIG, Peter. (2004) "Inteligência Artificial" – Translation of Second Edition. Rio de Janeiro/RJ.

[12] School of Mathematics and Statistics University of St Andrews. "Historia do Matemático Alan Turing", http://www.apprendre-math.info/portugal/historyDetail.htm?id=Turing, Escócia, October.

[13] SILVA, Maurício Samy. (2010) "HTML5 - A linguagem de marcação que revolucionou a web" - Publisher Novatec, São Paulo/SP.

[14] WALLACE, Dr. Richard S. (2003) "The Element of AIML Style", © ALICE A. I. Foundation, Inc., Octuber.

[1] WALLACE, Dr. Richard; TAYLOR, Anthony. "AIML Reference Manual", http://www.alicebot.org/documentation/aiml-reference.html, Octuber 2012.

SESSION

KNOWLEDGE DISCOVERY AND LEARNING

Chair(s)

Dr. Raymond A. Liuzzi
Dr. Peter M. LaMonica
Dr. Todd Waskiewicz

A Framework for Multi-source Semantic Information Extraction & Fusion for Collaborative Threat Assessment (SIFT)

Perakath Benjamin, Karthic Madanagopal, Kalyan Vadakkeveedu, Satheesh Ramachandran, and Paul Koola
Knowledge Based Systems, Inc., College Station, TX, U.S.A.

Abstract: *This paper describes the motivations, methods, and automation architecture of a framework for multi-source Semantic Information extraction & Fusion for collaborative Threat assessment (SIFT). First, the technical and pragmatic challenges that motivate the research ideas are summarized. Next, a characterization of the activities for generating decision enabling information from multi-source data is provided. This characterization, called the 'SIFT Method,' specifies the SIFT automation support requirements. The SIFT architecture is described next. Finally, the paper summarizes the significance and benefits of the SIFT solution and outlines key areas that would benefit from additional research and development. The application of SIFT is expected to significantly reduce 'data-to-decision' time through the use of semantic and collaborative visual analytics techniques.*

Keywords:

1   Motivations

This section summarizes the technical challenges that motivate the ideas described in this paper.

1.1   Information overload challenges

Defense and security analysts are drowning in a flood of information. Modern sensors and reporting systems produce unprecedented amounts of data; modern processing capability is transforming, churning, and translating that data at an ever-increasing rate. Yesterday's analysts had to be experts at interpreting the sparse data they could obtain; tomorrow's analyst will need different expertise: finding, corroborating, and interpreting actionable information from the rising flood of data.

Evolving and emerging information technology provides the opportunity to equip tomorrow's analyst for this challenge. Improvements in communications capabilities make collaboration in near real time and at continental distances a reality today. Semantic information processing provides the analyst with access to deeper understanding of data— bringing data closer to information even before the analyst

sees it. And modern visualization techniques allow for rapid assessment of large amounts of data and enhance the analyst's ability to communicate observations, hypotheses, and findings. A key challenge is to provide significant reductions in 'data-to-decision' time through the use of advanced methods and tools.

1.2   Asymmetric threat assessment challenges

Today our enemies can strike anywhere, anytime, and with a wide variety of weapons. As a consequence, a goal for homeland security should not be tied to a specific threat, but should rather address the underlying vulnerability of our society and the inherent unpredictability of hostile conspiracies. The Internet is a reflection of our social fabric, and we can leverage this open source information repository to measure indicators of threat around the globe. Most of this information is in unstructured textual form; consequently, automating the process of information extraction is particularly difficult. In spite of the significant advances made in the fields of machine learning, data/text mining, and information fusion, there are several technical voids that limit the utility of these techniques and tools for facilitating indications and warnings that support automated asymmetric threat detection.

An effective threat assessment scheme must support threat detection at the strategic, operational, and tactical levels of asymmetric warfare and must use information from a universe of data sources. These sources should enable the extraction of multiple measures supporting threat indications at the strategic, operational, and tactical levels. For example, threat dimensions should detect and track a potential adversary's wide variety of preparations such as an asymmetric attack, policy shifts, and/or advances in the capability or the acquisition of new technology. These requirements dictate the need for computational approaches to adapt to real time data feeds.

Emerging threats are fundamentally different in several ways. First, the threat is usually physically small (perhaps from a single operative) and is not easily observed, especially using existing reconnaissance systems. Second, the costs to carry out such asymmetric threats are very low.

For example, it no longer requires the resources of a nation state to develop a weapon of mass destruction. In today's world, analysts need to analyze vast amounts of information from multiple sensors and sources and piece together evidence necessary to understand, assess, and meaningfully interpret threat activities.

The next section describes the Multi-Source **S**emantic **I**nformation Extraction & **F**usion for Collaborative **T**hreat Assessment (SIFT) method.

2 SIFT method

This section describes the SIFT method: a characterization of the activities and their inter-relationships for generating decision enabling information from multi-source data. The SIFT method is comprised of four inter-related activities, shown in Figure 1.

As specified in Figure 1, SIFT provides four broad categories of functions: (i) Semantic Tagging; (ii) Information Fusion; (iii) Discover Knowledge for Sense Making; and (iv) Provide Collaborative Visual Analytics. These activities are summarized in the following paragraphs.

2.1 Perform semantic tagging

Semantic tagging refers to the activity of 'labeling' data with 'tags' that provide 'meaning' in the context of an application about the information contained in the data.

Two types of data are tagged (within the scope of this project): text and images. Text data tagging is done using an ontology-based natural language processing (NLP) capability. This includes the capability to generate Resource Description Format (RDF) semantic labels from multi-format raw text data. The SIFT image data tagging (a) leverages previously image-processed data and (b) exploits the information provided within the image 'metadata' tags. The image processing includes object detection, tracking, and background/foreground segmentation.

A unique aspect of our semantic tagging approach is to use an ontology to increase the semantic quality (depth of meaning represented) of the resulting text and image tags. To illustrate, the semantic tagging activity for an example sentence is shown pictorially in Figure 2.

The 'heavy lifting' that is necessary to convert the raw text into meaningful information is provided by the KBSI NLP 'pipeline,' shown in Figure 3.

The process shown in the figure uses a combination of NLP techniques and is capable of scaling to handle large text data collections in multiple formats. A unique innovation is the ability to use an ontology to increase the richness of the semantic tags. Each block in the pipeline is labeled according to the set of tags that are added to the input text after the input text has passed through the block.
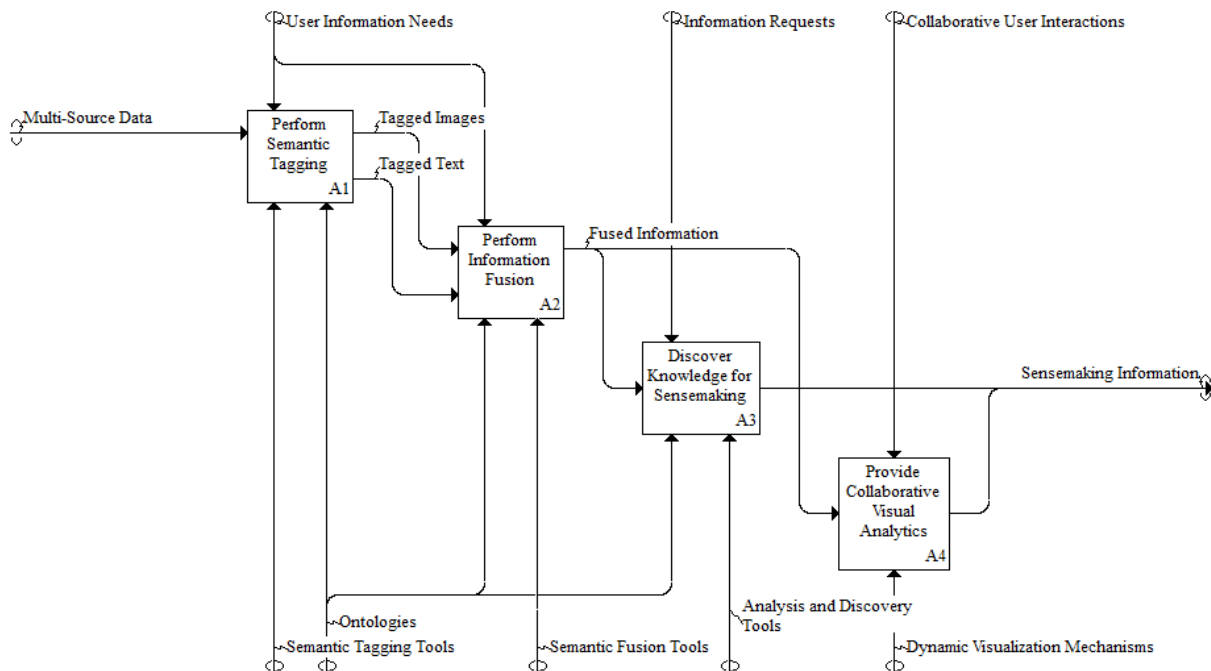
Figure 1. The SIFT Method
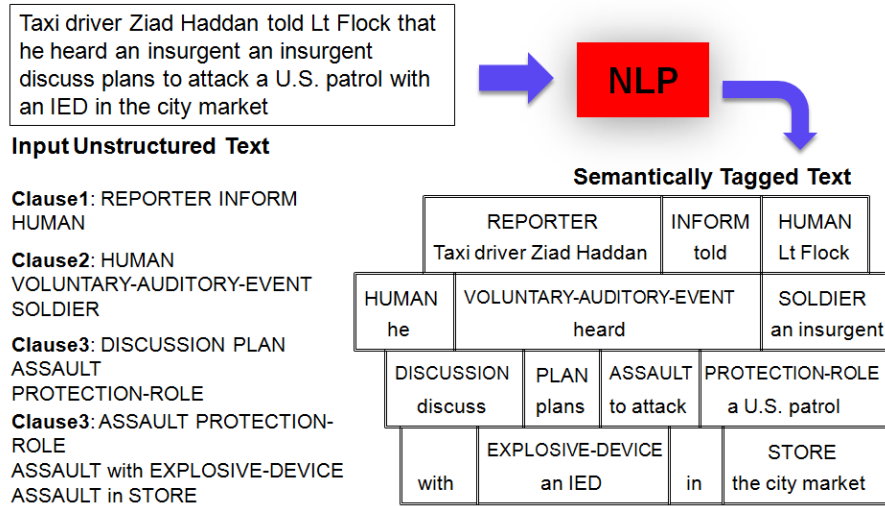
Figure 2. Semantic Tagging Induces 'Semantic Labels' to the Clauses in Every Sentence

- Sentence Boundary Detection (SBD)
- Tokenization (TKN)
- Named Entity Recognition Level 1 (NER1)
- Part of Speech Tagging (POS)
- Phrase Chunking (CNK)
- Named Entity Recognition Level 2 (NER2)
- Subject-Verb-Object, Clause Identification (SVO)
- Semantic Analyzer (SEM)
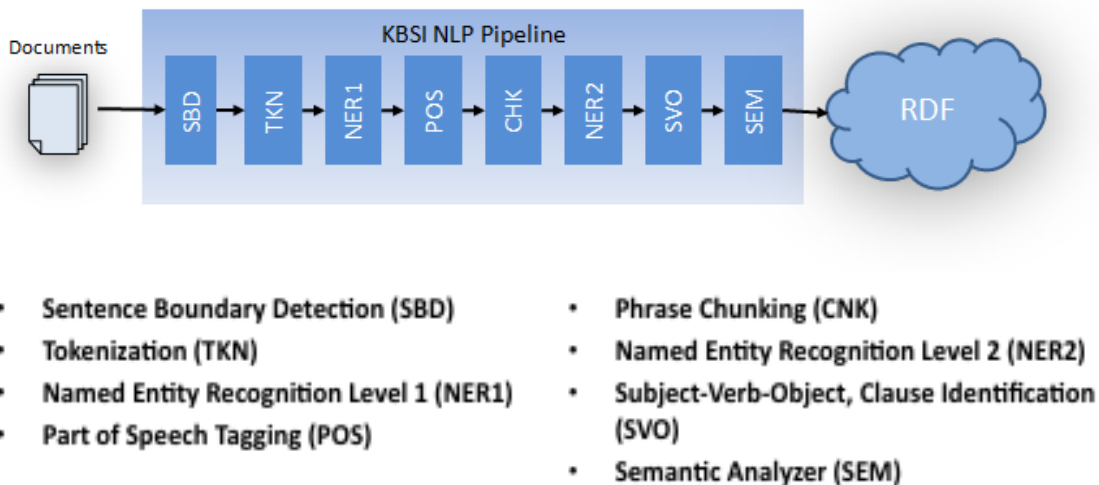
Figure 3. Dataflow Diagram of the KBSI NLP Pipeline

2.2   Perform information fusion

The semantic tagging activity produces an 'integrated tagged data set' that combines the semantic information contained in the tagged text, tagged images, and geospatial databases. The resulting integrated, semantically tagged dataset provides an expressively rich 'knowledge base' that will be useful for exploratory search and discovery and collaborative visualization and sense making. A key idea underlying our approach is the use of a 'reference ontology' to support the fusion of information derived from text and other types of intelligence-seeking sensors.   The current implementation of SIFT focuses attention on fusing information from text and image data

sources.   The approach may be generalized to fuse information from multiple types of data sources.

The multi-source fusion is at a higher level (level 2 and above) in the context of the JDL multi-sensor fusion terminology described in [1] and [2].   There have been research studies to apply ontological semantic modeling for knowledge representation and fusion for image and video data [3]. These methods address some of the challenges in dealing with multi-sensor fusion and image data fusion. Our research focused on information fusion approaches incorporating semantic ontological information extracted from text data sources (human sources and open sources, such as social media) along with image data.

Combining the JDL characterization of multi-sensor fusion with widely used models of situational awareness (SA)

(e.g., [4]), information fusion may be summarized as follows:

- Object Level (Level 1 Fusion). The first level of information fusion relates to SA on the objects that participate within the environment (the status, attributes, and dynamics of relevant elements in the environment). Within the context of threat assessment applications, this refers to fusion that enables generating a good SA of the actors, their attributes and roles, spatio-temporal location within the physical and cyber-world, etc.

- Situation Level (Level 2): The second level of information fusion relates to SA on the intents and motives and the relationships between the objects that participate in the environment. Relative to a threat assessment application context, this would refer to fusion that allows for generating a good SA of the current situation: the state of the world in terms of the structural relationships between the actors, the intent and motives of individuals and the group, the level of achievements towards the intents and motives, etc.

- Impact or Future State Level (Level 3): The third level of information fusion relates to SA on anticipated future courses of action of the participating entities, or the future situation. Within the context of threat assessment applications, this would refer to fusion that allows for generating a good SA of future evolution – the state of the world that could be – in terms of the evolution of the structural relationships between the

actors, the evolution of intent and motives of individuals and the group, the projected level of achievement towards the intents and motives at a future point in time, etc.

The SIFT information fusion approach uses an application domain ontology to intelligently guide the focused contextualization of the fusion activity. The key idea here is that the ontology acts as a 'bridge' that helps determine the mapping between the information items in the different tagged data forms (text, images, and geospatial data). The fusion technique involves performing the following process tasks: (i) the ontology, (ii) the tagged image data, (iii) the tagged text data, and (iv) the tagged geospatial data. The results of (i) through (iv) are then used to compute the 'semantic distance' between the concepts in the ontology and the concepts in (i) the tagged data, (ii) the tagged text data, and (iii) the tagged geospatial data.

The next step is to use a set of heuristics (e.g., find the intersection of 'low semantic distance' mapped ontology concepts for the various mappings) in order to determine the relative strengths of the semantic connections between the concepts of the three different tagged sets (images, text, and geospatial). The semantic connections determined through the application of this 'bridging' heuristic will be used to create fused tags that combine the three tagged data sets.

The example shown in Figure 4 illustrates the above ideas of using an ontology for fusing text and image data.
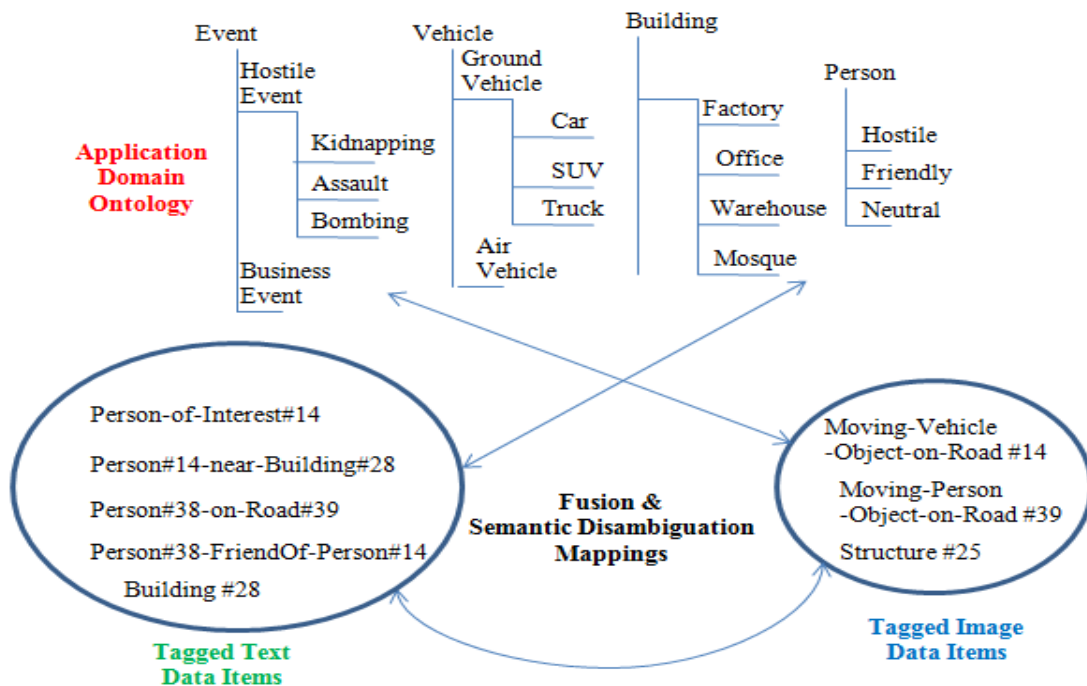
Figure 4. Example that Illustrates the Ontology-Driven Text and Image Fusion Solution Approach

In the above example, suppose that the instances of Moving-Vehicle-Object-on-Road, Moving-Person-Object-on-Road and the Structure (from the tagged image data) were determined to be 'semantically close' to the ontology concepts Vehicle, Person, and Building. In a similar manner, suppose that Person#28-on_Road#39 and Person#14-near-Building#28 (from the tagged text data) were determined to be 'semantically close' to the ontology concepts Hostile Person (subclass of Person). Further, it is apparent that Building#28 is an instance of the Building ontology concept. Assuming that the tagged data had time stamps that were nearly identical, the above semantic distance determinations will be used to induce a 'fusion mapping' between Structure#25 and Building#28, and also between Moving-Person-on-Road#39 and Person#38-on-Road#39. The latter mapping might help make the determination that Moving-Person-on-Road#39 was in fact Person#39 (or increased the probability of the belief in this assertion). The above example, though simple, illustrates the power and utility of an application domain ontology in the multi-source fusion and semantic disambiguation process.

2.3 Discover knowledge for enhanced situational awareness

The results of the semantic tagging and the information fusion activities produce a rich repository of information that may be exploited to support enhanced SA, sense making, and improved decision making. SIFT provides multiple mechanisms for collaboratively 'discovering' action-enabling knowledge from this information, including the following: (a) semantic search, (b) social network extraction and analysis, and (c) event extraction and analysis. The SIFT semantic search capabilities employ an ontology-driven approach that produce results with higher precision and recall; the social network extraction and event extraction are accomplished using ontology-directed semantic querying over the semantically tagged information. Lastly, the 'search collaboration support' in SIFT provides the following distinguishing features: (i) automatic detection of similar tasks between different users with notifications, (ii) the ability to browse the results of a different user who is working on a similar task, and (iii) the ability to modify one's own query builder with the user feedback provided by a different user.

2.4 Perform collaborative visual analytics

This activity provides dynamic visualization mechanisms to enable collaborating end users to better understand information contained within multi-source data leading to sense-making and enhanced SA [5]. This activity uses a semantic tag-based approach that maintains and traces through hierarchies of different scales of space and time granularity for asynchronous collaborative visual analytics. The semantic tags, carrying scaled spatio-temporal information, helps users navigate between text visualization, space visualization, temporal visualization, and spatio-temporal visualization.

3 SIFT architecture

An automation support toolkit has been developed and tested for the SIFT method described in the last section. The SIFT conceptual architecture is shown in Figure 5.

As shown in Figure 5, data streaming from multiple and distributed sources is first pre-processed and then semantically tagged using information extraction and ontology-based NLP methods. The semantically tagged information from multiple text and imagery sources is then fused using information fusion methods to collaboratively derive useful information for decision making. Collaborative and interactive visualization mechanisms are used to facilitate enhanced understanding and SA based on the information contained in the multi-source data. The results of the tagging, analytic processing, and collaborative visualization provide directed insight and enhanced SA to collaborating end users. The SIFT architecture solution provides the capacity to answer questions such as what are the objects or who are the persons of interest in the emerging situation, who are the key leaders and what are their social networks, where are the events taking place, how are the events being carried out, why are the operations of interest, etc.? Answering these questions involves augmenting human subject matter expertise with fused threat assessments derived from multiple sources and sensors while taking into consideration subject matter expert knowledge.

Text Processing Tools: This set of tools enables automated extraction of semantically tagged information from multi-source text data. The text data includes human observation-based data and social media data. Ontology-based NLP methods are used to perform the semantic information extraction and semantic tagging as described earlier. The text processing activities produce semantically tagged text data that is persistently stored and managed.

Dynamic User Interfaces
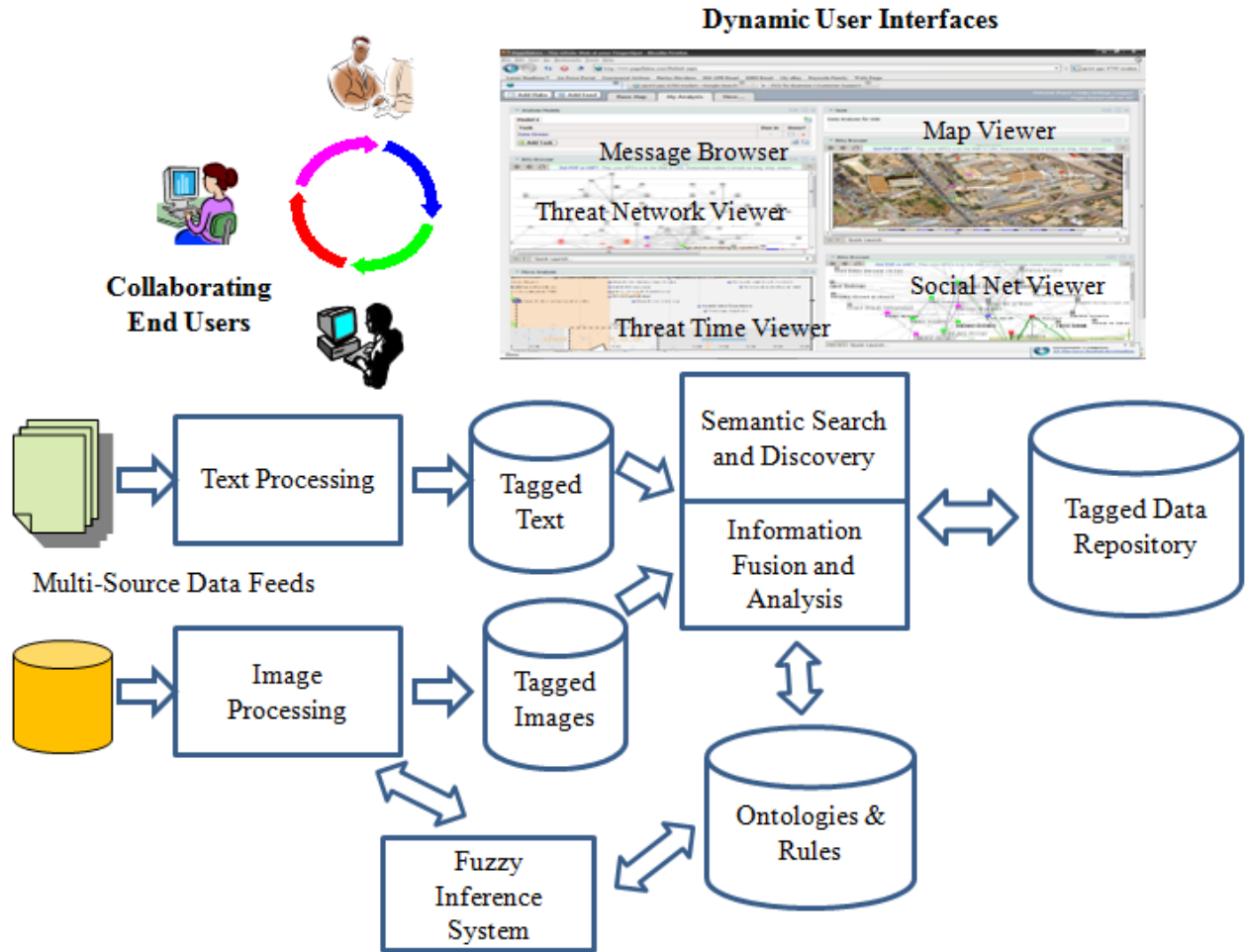
Figure 5. Automated Semantic Tagging, Information Fusion, Discovery, and Collaborative Analytics

Image Processing Tools: This set of tools enables automated processing of surveillance imagery data. The image/video processing algorithms perform lower and mid-level processing tasks such as moving target detection, tracking, object detection, background-foreground detection, learning classification model parameters, image segmentation and registration, etc. Human-in-the-loop 'anomaly detection' is performed using fuzzy inferencing techniques based on an analysts' assessment of imagery data. The image processing steps produce semantically tagged imagery data that is persistently stored and managed.

Fuzzy Inference System: This set of tools enables the capture of surveillance imagery analyst inputs and generates a characterization of 'anomalous objects' in the imagery based on the intuition of the human analysts. The intuition input classification for anomaly detection uses a fuzzy logic-based approach.

Semantic Search and Discovery Tools: This set of tools provides for intelligent search and collaborative discovery of information that will support threat assessment and decision making.   Ontology-directed semantic querying

methods are used to support intuitive discovery and information sharing between collaborating end users.

Information Fusion and Analysis Tools: This set of tools enables fusion of information generated from multi-source and multi-modal data. As described earlier, the types of data include surveillance imagery, human observation-based data and reports, and social media-based text and image data. The fusion-enabled collaborative analysis is a semi-automated process with a human-in-the-loop aiding the analysis and incorporating the reasoning results as part of new analysis hypotheses and explorations. There are tasks that humans can perform better such as *connecting the dots* and *intuition* to generate potential hypotheses about an event or persons of interest. A computer is better at validating or refuting these hypotheses by processing, retrieving, and rank ordering various sources of information and facts in the presence of various levels of uncertainty. By combining the strengths of human analysts with that of the computers, SIFT supports faster and more accurate threat analysis.

The results of the fusion are used by a group of analysts working together collaboratively with the help of the automatic and semi-automatic analysis software tools. The

collaboration support provided by the SIFT architecture includes 'intra-group' (e.g., within a single analysis center) and 'inter-group.' The iterative and incremental understanding of the overall picture of the situation emerges based on such a collaborative effort, with the help of multi-source fusion techniques. Collaboration plays a crucial role in the collective ability of threat assessment teams to effectively process and understand the various individual pieces of discovered "knowledge-bits," each with its own level of inherent uncertainty, piecing them into context, and drawing conclusions. A key aspect of collaborative knowledge discovery is information integration and reasoning with uncertain information in order to derive the optimal collective SA.

In addition to fusion, the deep semantic content of the tagged data is exploited for automated social network extraction and analysis and automated event extraction and analysis. The example SIFT user interface displayed in Figure 6 illustrates how Bayesian fusion may be used to resolve uncertainties in a social network.

Collaboration & Visualization Tools: this toolset enables collaboration and dynamic visualization support using collaborative visual analytics techniques to support end user interpretation and enhanced decision making. Specifically, SIFT provides dynamic and interactive visualization aids to help collaborating end users visualize information to support sense-making as shown in Figure 7.

SIFT supports the technique of 'brushing and linking,' which enables a change in one view to automatically propagate to all other views of related information [6]. The SIFT automation architecture provides brushing and linking for multiple user interface views including social networks, events (including a timeline view), and geospatial map-based visualizations as shown in the figure.

4 Benefits and opportunities for further research and development

4.1 Summary of SIFT benefits

This paper described the motivations, methods, and automation architecture of a framework for multi-source Semantic Information extraction & Fusion for collaborative Threat assessment (SIFT). The main benefits of the SIFT solution described in this paper are (i) significant reductions in 'data-to-decision' time through the use of semantic and collaborative visual analytics techniques; (ii) substantial gains in the quality of shared SA through fusion of information gained from multiple sources; (iii) significant increases in the ability to exploit information and knowledge embedded within data through the use of ontology-driven semantic methods; and (iv) superior abilities to leverage and better use increasingly scarce and time-constrained human cognitive and decision making skills through the use of collaborative sense-making techniques.
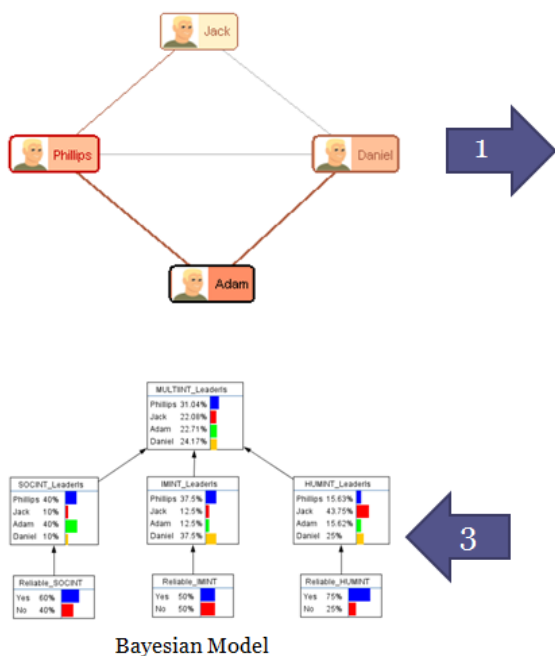
Degree grouped by Source Type:

Name	HUMINT	SOCINT	IMINT
Jack	2	0	0
Phillips	1	1	1
Adam	1	1	0
Daniel	2	0	1

Probabilities calculated from above table:

Name	HUMINT	SOCINT	IMINT
Jack	43.7%	10%	12.5%
Phillips	15.6%	40%	37.5%
Adam	15.6%	40%	12.5%
Daniel	25%	10%	37.5%

Figure 6. Multi-Source Bayesian Fusion to Support Enhanced Situational Awareness
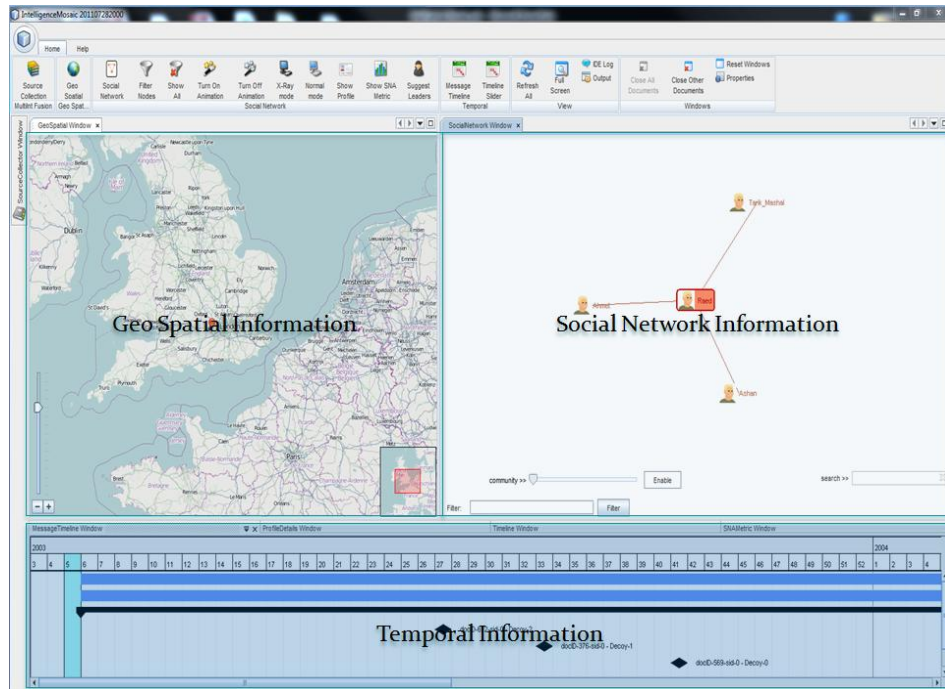
Figure 7. Dynamic, Multi-view, Intelligence Analysis SIFT User Interface Example

4.2   Opportunities for future research and development

Areas that would benefit from further research and development activity are summarized in the following subsections.

4.2.1   Ontology management capabilities

The SIFT method uses an ontology-based approach for semantic information extraction and semantic tagging. Needed are better tools for discovering, eliciting, and maintaining ontologies.

4.2.2   Addressing uncertainty and errors in data

The SIFT method processes raw data of multiple types and modalities and in different formats. Real world data sources contain data that is incomplete, corrupted, and include different types of errors and inconsistencies. There is a need for methods and tools to address and manage errors and uncertainties inherent in data.

4.2.3   Multi-source fusion

Much of the research in data fusion is focused on generating useful information from multiple types of (structured) sensor-based data. There is a need for research on fusion methods that focus on unstructured data types and fusing information derived from structured data with information derived from unstructured data (e.g., fusing information derived from a sensor with information derived from unstructured text).

4.2.4   Truth maintenance

The SIFT methods include the use of automated reasoning for inferring new information. The problem of maintaining and evolving a complex knowledge repository derived from the application of multi-source fusion and reasoning techniques has not been addressed in the current implementation of the SIFT solution. There is a need for better methods and tools that support truth maintenance in knowledge repositories.

4.2.5   Collaboration support

The SIFT solution provides collaboration support for synchronous collaboration among distributed end users. Better methods for collaboration are needed that support synchronous and asynchronous collaboration among users that are distributed in time and space.

5   References

[1]   A. Steinberg et al., "Revisions to JDL Datafusion models," Technical report ERIM International, Inc., 1999.

[2]   D.L. Hall and J. Llanas, "An Introduction to multisensory data fusion," Proc. of the IEEE, Vol. 85, No. 1, Jan 1997, pp-6-23.

[3]   Christopher Town, "Ontological inference for image and video analysis," J. of Machine Vision and Applications (2006) 17(2): 94–115,Springer-Verlag.

[4]   Endsley, M. R. Theoretical Underpinnings of Situation Awareness: A Critical Review. In Endsley, M. R., and Garland D. J. (Editors). 2000. *Situation Awareness Analysis and Measurement*. Lawrence Erlbaum Associates.

[5]   Heer, J., Viegas, F., Wattenberg, M., Voyagers and Voyeurs: Supporting   Asynchronous   Collaborative Information Visualization, ACM Human Factors in Computing Systems (CHI), 1029–1038, 2007.

[6]   Heer, J and Shneiderman, B., Interactive Dynamics for Visual Analysis: a Taxonomy of Tools That Support the Fluent and Flexible Use of Visualizations.   ACM Queue, Feb. 1, 2012.

Cognitive RF Systems and EM Fratricide – Part II

Gerard T. Capraro

Capraro Technologies, Inc., 401 Herkimer Road, Utica, NY 13502 USA

Abstract

The United States Department of Defense and researchers throughout the world have been addressing the overcrowding of the radio frequency (RF) spectrum. When the frequency spectrum is measured over time, technologists have shown that the spectrum is underutilized. This has led to numerous studies concerning cognitive radios, networks, and radar systems to intelligently choose frequencies, waveform parameters, antenna beam patterns, etc. to operate with conventional receivers without causing electromagnetic (EM) fratricide. In many of these studies there is an inherent assumption that the cognitive system knows when and where the fratricide occurs. In a previous paper [11] we presented two approaches for determining if a cognitive solution is causing EMI in nearby military conventional receivers. In this paper we extend the solution to include commercial transceivers along with military and describe a solution of how a radar and communication transceivers can cooperatively share the spectrum and reduce EM interference.

1.0 Introduction

The radio frequency (RF) spectrum is crowded and more space is needed for wireless internet access, cell phone communications, and for military and civilian usage. The US Congress passed a bill to open up more spectra [1] to auction off RF frequencies belonging to the television broadcast industries. To slow down the need for more frequencies and to reduce the number of cell towers to accommodate the growing number of mobile phones, the industry is deploying microcell, picocell and femto cell technologies. However, these initiatives alone will not solve spectrum crowding.

When the frequency spectrum is measured over time, technologists have shown that the spectrum is underutilized. Recognizing this, there have been numerous research projects funded by the US Department of Defense (DOD). The Defense Agency Research Project Agency (DARPA) has probably funded the most projects in this area. Through this research, we now have two distinct users defined as the primary user (PU) (i.e. those who own the license for the frequency range) and the cognitive user (CU)

(i.e. those users trying to share the spectra either by using broadband signals or sampling the spectra in time and transmitting when the PU is not transmitting). Most significant projects in this area include the DARPA XG program and the Wireless Network after Next (WNaN) program. In addition to these efforts, there has been a move to apply Cognitive Radio (CR) technologies to the radar domain (Cognitive Radar efforts) and radio networks. Some of these systems sample the spectrum and transmit if no one else is transmitting at any given frequency. This approach can cause electromagnetic interference (EMI) in nearby receivers. Many people have recognized this problem and have addressed it in many different ways [2 - 6]. Many of their solutions inherently assume they know information about the victim receiver, but do not address how this information is obtained. Our last paper [11] addressed this issue. We presented two approaches for determining if a cognitive solution is causing EMI in nearby military receivers. Since the publication [11] DARPA released a request for proposal (RFP) [16] asking for approaches to solving the EMI problem through techniques that would cooperatively share the spectrum between military radar systems and military and civilian communication systems, e.g. mobile cell communications. This paper will briefly review the results of the previous paper and present an approach of how radar and communication systems can cooperatively share the spectrum and reduce EMI.

Section 2 briefly reviews cognitive radio and radar technologies. Section 3 defines the problem we are addressing. Section 4 reviews two documented solutions [11] and presents a new solution based on cooperatively sharing of the RF spectrum. Section 5 provides a summary and conclusions.

2.0 Some Cognitive Efforts

Next Generation (XG) Program

The XG (neXt Generation Communications) program is developing an architecture that will open up the spectrum for more use by first sensing and then using unused portions of the spectrum.

Figure 1 from [7] is a logical functional diagram of the concept of operations of XG's policy-agile spectrum user, which uses a computer understandable spectrum policy capability. The major components are the Sensor (which senses the environment for determining its availability), Radio (the communications device that can dynamically change its emission and reception characteristics), Policy Reasoner (manages spectrum policy information), and System Strategy Reasoner (manages the multiple radios on a platform).
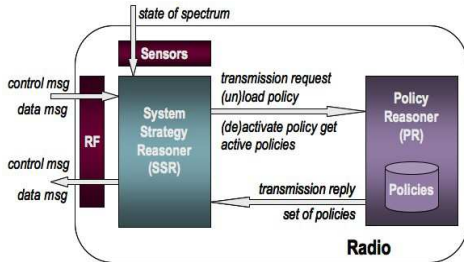
Figure 1.0 Policy-Agile Operation of XG Spectrum-Agile Radio

The last two components are of particular interest in that they utilize Semantic Web technologies. Operating a radio in different parts of the world requires that radios abide by the policies in the area where they are located. The XG program has developed its own XG policy language (XGPL) which uses OWL as its standard representation and will be implemented within the Policy Reasoner.

The Wireless after Next (WNaN)

The WNaN being performed by Raytheon BBN Technologies and funded by DARPA [8] is developing a scalable, adaptive, ad hoc network capability that will provide reliable communications to the military. The basic ingredients of their design are composed of a Dynamic Spectrum Address capability based upon the XG program. It also has 4 multiple transceivers and a disruptive tolerant networking (DTN) capability. The four transceivers provide fault tolerance and allows the system to pick the best channel for communications. The DTN capability allows the nodes to store packets temporarily during link outages. The WNaN also has content based access that allows users to query the network to find information and allow the system to store critical data at locations to minimize time and bandwidth.

Cognitive Radio

Another effort related to communications, and having

similar goals to the XG program, is the Cognitive Radio [9]. Its objectives are to efficiently utilize the radio frequency (RF) spectrum and to provide reliable communications at all times. A basic cognitive cycle view of the radio is illustrated in Figure 2. A general overview and projections of the Cognitive Radio in our society can be found in [10].
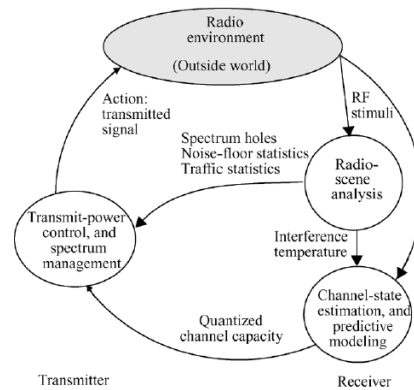
Figure 2.0 Basic Cognitive Cycle

Cognitive Radar

Interest in cognitive radar is growing in the radar community. Figure 3 describes a cognitive radar that is primarily concerned with the tracking stages of a radar [12]. In Figure 4 a cognitive radar architecture is shown from the first textbook written on this subject [13]. The commonality of these designs are the feedback loop between the transmitter and receiver, use of outside sources of information, and the implementation of a learning process.
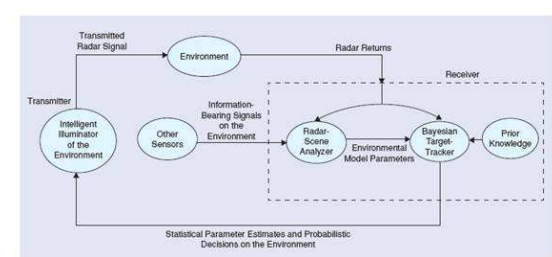
Figure 3.0 A Cognitive Tracking Architecture
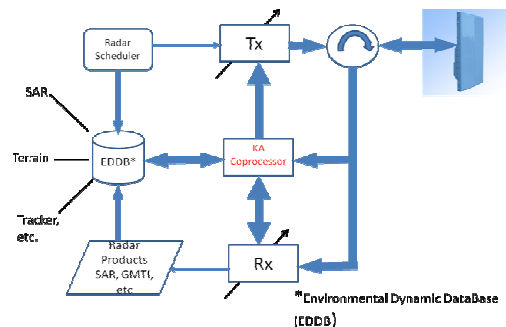
Figure 4.0 Another Cognitive Radar Architecture

3.0 Problem Definition

In all of the above programs and many others, the CU chooses which frequency to transmit using frequency policy rules based upon location and whether someone is currently transmitting within the range of interest. There are at least three issues with this approach. The first is related to the sensing of the environment. What happens if a nearby receiver is not transmitting but is waiting to receive a signal at frequency f1, for example a bistatic radar receiver or an electronic warfare receiver? They don't transmit, they just receive. The second issue relates to the following scenario. Let us assume that one decides to transmit broadband signals below the sensitivity levels of any nearby receivers. As the number of CR increases, the signals within a nearby receiver's passband may exceed the noise floor and interfere with the performance of the receiver [14]. The third issue occurs when a CR decides to transmit at a particular frequency because there are no signals present. The chosen frequency is based upon a linear relationship between the frequency chosen and the sensed environment. The decision policy does not take into effect the nonlinearities between the chosen frequency and other nearby frequencies which can mix nonlinearly and cause receiver intermodulation or mix within the receiver's frontend and cause spurious responses. Most electromagnetic interference (EMI) situations are nonlinear.

EM Compatibility Paradigm Shift

EM fratricide is the situation where we degrade the performance of our own system(s) with our own system(s), e.g. an onboard radar's energy is received by an onboard communication receiver and that degrades the receiver's performance. This is a serious problem, since there are multiple sensor and communication systems onboard platforms. Military weapon systems are engineered to prevent such phenomena between hardware located in close proximity. The military has standards for describing how to build and test hardware for EMC, and how to test weapon system platforms for EMC, e.g. Military Standards 461E and 464. In the 1970's and 1980's the DOD developed EMC prediction tools to assess the EMC of its weapon systems. Using these software tools to perform EM measurements in the 1970s was a major paradigm shift for the EMC community.

Just as we needed a change by using software tools to assess a system's EMC in the 1970s, we now need to rethink how to build complex systems that employ waveform diversity and some of the proposed XG and cognitive radio and radar spectrum management concepts. Whereas in the 1970s we required software tools to predict where to hone our measurements, we now need to use software to help determine when EMI may occur in real time, and manage the EM spectrum while the platform or radio/radar increases its total performance.

4.0 Potential Solutions

To solve the EM fratricide issues discussed above some people are looking to change the beam pattern of the transmitter so that the power coupled to a victim receiver is reduced, some wish to change the transmitted signal's polarization, and of course, there is the attenuation gained by employing orthogonal waveforms. All these solutions help reduce the amount of degradation caused to a friendly receiver. However, these techniques inherently are assuming that one knows that the receiver is being degraded. How would a cognitive radar, radio or a WNaN know about the receiver?

There are currently three scenarios where one can implement a capability to solve the fratricide issue. One is on a single platform such as an aircraft, ship, or a complex weapon system where multiple conventional and cognitive EM equipment reside. The second scenario is concerned with WNaN where we propose to extend its capability and add a gateway to communicate with non cognitive radios as developed under another DARPA program. The EMC paradigm shift for both scenarios requires that the equipment report to a node that is managing the EMC of the platform or the total network. A discussion of these two scenarios can be found in [11]. The third approach is the cooperatively sharing of the RF spectrum which may require modifications of all RF transceivers.

The third approach is concerned with the growing demand for complimentary coexistence between radar and communications devices that concurrently and cooperatively operate in shared spectrum. Over the past several years, DARPA and other DOD agencies have initiated numerous research projects in dynamic spectrum sharing among various military and commercial systems. Some of these approaches were discussed above. However, these spectrum sharing approaches, which relied upon spectrum sensing, did not map the presence of recently non-radiating nodes nearby. They do not implement cooperative techniques amongst those sharing the spectrum, nor have they taken into consideration the benefits possible through anticipating the near-term actions of other users. The DARPA RFP [16] addresses these issues, and requests solutions having radar and

communications devices share information on EMI and how to best remedy the situation in near real time.

Advances in wireless networking and information management technology have resulted in the proliferation of communications devices (e.g., smart-phones) and mobile ad hoc networks (MANET). While waveform diversity and advanced processing has led to the deployment of multifunction wideband radars and sophisticated RF sensor systems on unattended aerial vehicles (UAV), employing embedded computing and geographical/global positioning systems (GPS). Each of these developments requires additional spectrum to achieve mission goals. This RFP addressed three real world scenarios, the interaction of radar with 1.) small cell broadband (SCB), 2) 802.11 Wi-Fi hot spots for the commercial world and, 3.) MANET systems for the military world. Operations in an electronic countermeasures environment must also be considered. Figure 5 illustrates the problem. The number of cell phones is growing at a faster rate than predicted just ten years ago. In order for the industry to provide good coverage in dense environments, including urban areas, while minimizing the cost of additional cell towers, SCB technology [17] is being pursued to exploit the Internet. On the military side, the development of MANET allows communications devices to be mobile and yet be able to communicate long distances without substantially increasing power and without building expensive communications infrastructure.

Figure 5 presents a scenario with radar having the higher priority (primary) and communications as secondary. The first question the radar cognitive processor needs to answer is whether the radar is being interfered with by a MANET, SCB, heterogeneous transmitters/receivers (T/R) or Wi-Fi, and thus requires receivers sensing these bands. In addition, it must quantify the impact of this interference. This cannot be accomplished via "noise" power measurements alone, and must be signal structure sensitive. The radar requires cognitive embedded calibration such as the insertion of known signals (non-interfering false targets) in the receiver front-end. In doing so, the radar will correlate how well it detects these false targets against other signals that it receives from nearby emitters. If a particular communications system is identified as causing interference, then the radar can embed an encrypted message in its next transmission alerting the communications system to the impact, and the two systems can negotiate how to eliminate the EMI. Alternatively, the communications system will communicate to the radar, its position, power output, location, modulation, frequency,

antenna pattern, etc. This will help the radar cognitive processor determine which system is causing the EMI. In addition the radar will also transmit similar information about its transmissions so that the communications systems can use this information when they have a higher priority and the radar may be causing EMI.

When communications devices are primary, they will broadcast while receiving specifics about the radar transmissions. The broadband routers and other heterogeneous T/R devices will require a cognitive capability and receiver that can detect encrypted radar messages, or, alternatively, receive notification messages from other T/Rs. In this manner they can assess whether they are being interfered with by the radar systems. This may be accomplished by correlating the radar transmissions with the packet errors detected during communications. Correlations of excessive packet errors along with distance, timing, antenna characteristics, terrain, and power rules will allow the communication systems to determine how the radar impacts performance. The T/R nodes can communicate to the radar, via distributed control protocols, describing its EMI and jointly, the two cognitive processors will alleviate the EMI.
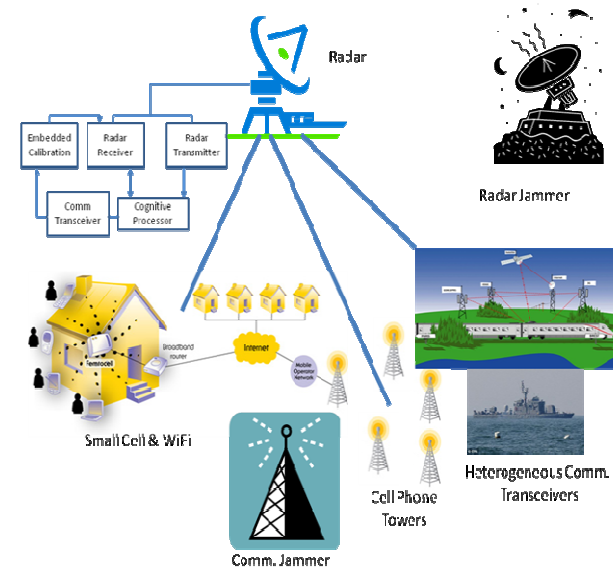
Figure 5 Cooperative Spectrum Sharing

Each T/R node will have a software architecture similar to that shown in Figure 6. Each node should have mission goals and be able to set priorities. We have to establish performance metrics to know how well each node is functioning. We will use databases and knowledge bases to assess and maintain a node's status, and ontologies to describe the rules of spectrum operation in different theaters. Classification is

necessary for radar in defining clutter statistics and in communication nodes to help define multipath and propagation loss. To assess how well a node is performing, we implement a series of tests. Background analysis is a required capability for computing propagation losses, multipath, clutter statistics, etc. and they will use, for example, Land Use Land Cover (LULC) and Digital Elevation Map (DEM) databases. Auxiliary information is required to obtain information such as jammer locations.
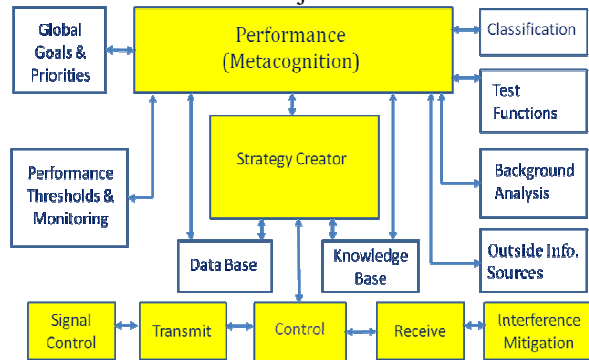
Figure 6 Cognitive Node Software Architecture

5.0 Summary/Conclusions

We have briefly reviewed two approaches for solving EM fratricide where cognitive systems are functioning near conventional T/R [11]. A new approach is being investigated by DARPA where communications and radar systems will cooperatively share the spectrum. We have described the problem and an approach where each T/R node will have a cognitive software architecture as shown in Figure 6. This capability will be embedded in military/commercial radar and radios e.g. MANET, cell towers, WiFi routers, and SCB routers. There is a lot of work that needs to be done in developing these cognitive nodes and how they will cooperatively work together to solve EMI in near real time, learn from their discoveries and pass this knowledge onto other T/R nodes so they can learn from each other.

References

[1]http://blog.broadcastengineering.com/blog-opinions/2012/02/17/congress-paves-way-for-broadcast-spectrum-auctions/
[2] P. Popovski, Y. Hiroyuki, K. Nishimori, R. Di Taranto, and R. Prasad, "Opportunistic Interference Cancellation in Cognitive Radio Systems", 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007.
[3] O. Ozdemir, E. Masazade, C. Mohan, P. Varshney, I. Kasperovich, R. Loe, A. Drozd, and S. Reichhart, "Spectrum Shaping Challenges in Dynamic Spectrum Access Networks with Transmission Hyperspace",

Waveform Diversity and Design Conference, January 2012
[4] Y. Fei and Z. Wu, "An Interference Cancellation Scheme for Cognitive Radio Network", 6th International Conference on Wireless Communications Networking and Mobile Computing, Sept. 2010
[5] V. Chakravarthy, Z. Wu, A. Shaw, M. Temple, R. Kannan, and F. Garber, "A general Overlay/Underlay analytic Expression Representing Cognitive Radio Waveform", Waveform Diversity and Design Conference, June 2007
[6] E. Beadle, A. Micheals and J. Schroeder, "New Alternatives for Interference Tolerant Waveforms Hosted on a Software Programmable Multi-Mission Platform", Waveform Diversity and Design Conference, January 2012
[7] D. Elenius, G. Denker, andD. Wilkins, "XG policy Architecture", ICS-16763-TR-07-108, SRI Project No. 16763, Contract No. FA8750-05-C-0230, April 2007
[8] J. Redi and R. Ramanathan, "The DARPA WNaN Network Architecture", The 2011 Military Communications Conference, November 2011
[9] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications", IEEE Journal on Selected Areas in Communications, vol. 23, no. 2, pp. 201 – 220, February 2005
[10] Ashley, S. "Cognitive Radio", Scientific American, http://www.sciam.com/article.cfm?chanID=sa006&colID=1&articleID=000C7B72-2374-13F6-A37483414B7F0000, February 20, 2006.
[11] G. Capraro and I. Bradaric, "Cognitive RF Systems and EM Fratricide" Proceedings of the 2012 International Conference on Artificial Intelligence, Volume I, pp 110-114, July 16-19, 2012
[12] S. Haykin, "Cognitive Radar", IEEE Signal Processing Magazine, pp. 30 – 40, January 2006
[13] J. Guerci, "Cognitive Radar", Artech House, 2010
[14] I. Bradaric, G. Capraro, and D. Weiner, "Ultra Wide Band (UWB) Iterference – Assessment And Mitigation Studies", AFRL-SN-RS-TR-2006-42, February 2006, http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA446049
[15] H. Keyton, "Networks: Adapting To Uncertainty", DARPA 50Years of Bridging the Gap, http://www.darpa.mil/WorkArea/DownloadAsset.aspx?id=2570
[16]DARPA-BAA-13-24
https://www.fbo.gov/index?s=opportunity&mode=form&id=8e85f738e53747b502b4b9c3732c2e54&tab=core&_cview=1
[17] http://www.smallcellforum.org/

Sensor Planning with Cloud-Based Predictive Analytics

Aaron Wheeler, Jim Dike, and Michael Winburn

3 Sigma Research, Indialantic, Florida, USA

Abstract – *Large-scale disasters come from many sources, ranging from natural events like hurricanes, tornadoes, earthquakes, fires, famines, volcanic eruptions, or tsunamis; to human-made crises like nuclear accidents or political unrest. Successful mission planning and replanning requires exceptional situational awareness and this means having the right information at the right time. Mission planning for disaster recovery requires reliable access to massive amounts of data. Cloud storage offers an ideal solution for a repository of collected sensor data. Sensors could include aerial reconnaissance by both piloted aircraft and UAV, people on the ground collecting information, and social mediate filters that identify and extract important events and trends. Logistics, schedules, and costs impose limitations on data collection for a small number of airborne sensors over a large geographic area. Poorly managed sensors could add redundant or unneeded data to the cloud, or miss better or higher priority collection opportunities.*

The benefits of cloud-based predictive analytics for sensor planning include 1) predictive algorithms for real-time distributed discovery of sensor data needs and sensor data products, 2) routing of sensor data needs to sensors able to collect the data, 3) real-time sensor management to prioritize unfulfilled information requirements and optimally task sensors to collect this information, and 4) learning and anticipation of future sensor data collection needs.

We report on the status of our current research efforts to address the problem of sensor planning with cloud-based predictive analytics and discuss our future research and development objectives.

Keywords: sensor planning, predictive analytics, cloud computing, ontology

1   Introduction

Large-scale disasters come from many sources, ranging from natural events like hurricanes, tornadoes, earthquakes, fires, famines, volcanic eruptions, or tsunamis; to human-made crises like nuclear accidents or political unrest.

Successful mission planning and replanning requires exceptional situational awareness and this means having the right information at the right time. Sensors could include aerial reconnaissance by both piloted aircraft and UAV, people on the ground collecting information with GPS-enabled devices (GPS linked text, audio, still pictures, and video), and social mediate filters that identify and extract important events and trends. Logistics, schedules, and costs impose limitations on data collection for a small number of airborne sensors over a large geographic area. Web-based sensors can operate continuously, but might produce large amounts of unnecessary data. Mission planners avoid these problems by performing sensor planning prior to actual collection.

Disaster response and recovery mission planners face many challenges in managing sensor data collection. Sensor planning often occurs before collection and adaptive replanning of sensor data collection does not happen easily if at all. This results in sensors adding unneeded data to the cloud or missing better or higher priority collection opportunities, as well as unexpected information requirements not getting met. A sensor may get tasked to collect information that already exists. Two or more sensors may get tasked to collect the same information (piloted aircraft and UAV both taking imagery of the same area). A poorly informed sensor may collect sub-optimal information, such as an aircraft taking imagery over only part of the needed area.

Mission planning for disaster response and recovery requires reliable access to massive amounts of data. Cloud storage offers an ideal solution for a repository of collected sensor data. But this solution requires having some way to find information already collect and to identify or anticipate unmet information needs. Otherwise, sensor planners cannot efficiently prioritize and schedule sensor collection tasks.

Missions planners can improve sensor planning in dynamic situations by predicting possible information needs in advance. The benefits of cloud-based predictive analytics for sensor planning include 1) predictive algorithms for real-time distributed discovery of sensor data needs and sensor data products, 2) routing of sensor data needs to sensors able to collect the data, 3) real-time sensor management to prioritize unfulfilled information requirements and optimally task sensors to collect this information, and 4) learning and anticipation of future sensor data collection needs.

This paper reports on the status of our current research efforts to address the problem of sensor planning through cloud-based predictive analytics that match information needs to collected information and identification of unsatisfied information needs.

Section 2 of this paper provides background information about our proposed solution and its components. Section 3

describes our methods and approaches for handling a number of key issues. Section 4 discusses our ideas for implementing the approaches presented in section 3, and section 5 concludes with a discussion of the current status of our research and some future research directions.

2   Background

Sensor planners cannot always plan in advance all sensor data collection tasks needed for a mission because they do not have all the required information before the mission begins and because unexpected events occur during the course of the mission that change information requirements. During an ongoing mission, sensor planners may not have time and resources to effectively manage information needs. Sensor planners must first find the data or information if it exists and, if it does not, schedule a sensor collection task. Sensor planners should also know when data and information no longer has relevance so that sensor planner can stop sensors from streaming unwanted data into the cloud.

2.1   Data and Information

In this research we make a distinction between data and information. Data provides a representation of something. Information describes a useful subset of this raw data [1].

We consider two categories of sensors, data sensors and information sensors. Figure 1 shows the relationship of these sensors to the cloud and to information needs. Data sensors operate outside the cloud to bring external data into the cloud to make it available for use directly or as a source of information for other sensors. Information sensors may operate inside or outside the cloud. Those operating outside the cloud mine other data sources for information and write this information into the cloud. Information sensors operating inside the cloud search data sources for information.

Figure 1. Sensors and information needs.
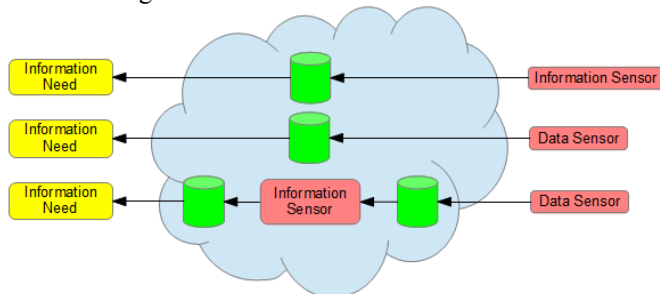
2.2   Architecture Components

Figure 2 shows the primary components of the sensor planning process. We begin by searching the cloud for published information needs. These needs get published to the cloud by mission planners and/or mission participants. New information needs trigger a search for already published data

that satisfies the need. If this data exists then we simply report its location in the cloud. Otherwise, we have an unmet information need that we report to the sensor planner. The sensor planner matches the information need to a sensor able to collect the data then schedules the sensor to collect the data. The sensor publishes the collected data to the cloud where it gets found and the unmet information need becomes satisfied. When we no longer have a need for the information, the published need gets reported to the sensor planner which stops the sensor tasked to collect the data.

Figure 2. Sensor planning components.
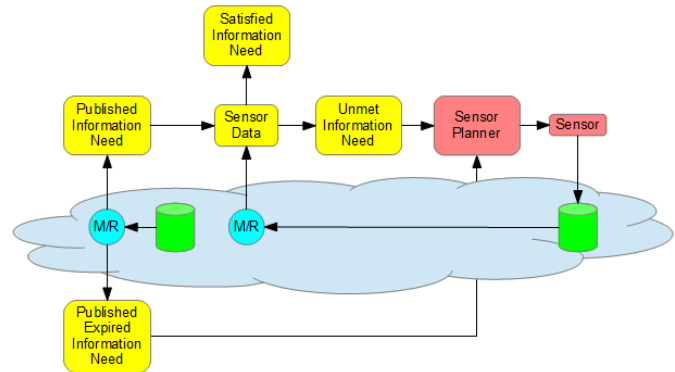
This solution works reasonably well 1) to find explicit information needs, 2) to prevent sensors from adding redundant information to the cloud, 3) to have only needed sensors collecting data, 4) to schedule new sensor collection tasks, and 5) to stop sensors from collecting data no longer needed.

This solution relies on purposeful, dependable, and timely publication of information needs. This does not necessarily present a problem for a well-integrated and managed organizations able to follow the required procedures. However, this solution works less effectively for ad hoc coalitions of organizations trying to coordinate and plan disaster response and recovery operations.

We would like to add to this basic solution additional mechanisms to provide predictive analytics to support discovery of both information needs and data no longer needed. Furthermore, we would also like to provide cloud performance metrics to inform sensor planners about the current status of the cloud and how starting and stopping sensor collection tasks will affect system performance.

Figure 3 shows the discovery engines used to find information needs and expired information needs in the cloud. We assume during a disaster response/recovery mission that relevant events get published to the cloud. We know that certain events or combinations of events will create information needs. The information discovery engine mines cloud databases for such events and generates discovered information needs from them. We can discover expired information needs by examining

system data for access times for published sensor data. We assume data not accessed after some expected time has lost its value and we can signal the sensor planner to stop the sensor from streaming additional data into the cloud.

Figure 3. Discovery engines.
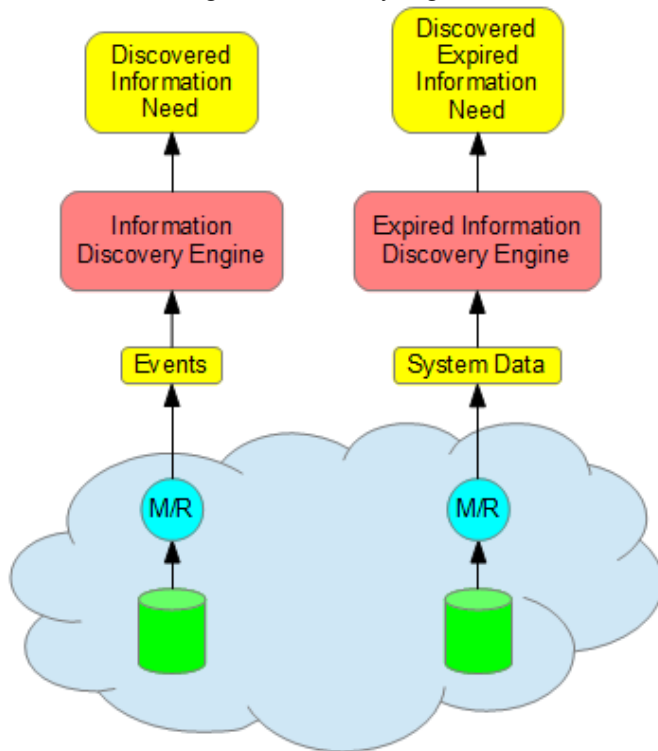
3 Methods

Here we describe our thoughts and methods regarding sensor tasking, performance, prediction, data discovery, and knowledge representation.

3.1 Sensor Tasking

Sensor planning has several challenges. Sensors can only perform one collection task at a time. Data collected by the sensor may only have value if collected by a certain deadline or within a specific period. Sensor might need to operate during certain times, e.g. daylight hours, or cannot operate under particular weather conditions. Some sensors, like aircraft or people, must to travel to a location, collect data, return, and recover before becoming available for another collection effort. The time and cost of travel (e.g. fuel, maintenance, etc.) affects the amount of time spent collecting data.

All of this means that some needs may simply not get met. Sensor planners need to know early when they cannot get the information they want when they want it. They need to know who needs the data or information so they can prioritize

collection efforts and notify those who will not have their request processed.

An information need may depend on information from an information sensor that itself depends on one or more sources of data. Each of these data sources may have its own data sensor. Sensor tasking requires knowing the dependencies between sensors so that data sensors get scheduled collect data before the information sensor that depends on them begins its collection task.

3.2 Performance

Sensors that add data or information to the cloud affect cloud performance by reducing available storage and slowing performance while the data or information distributes throughout the cloud. Sensors that collect information from data stored in the cloud reduce cloud performance while consuming CPU cycles and network bandwidth during search for and of files.

Sensors that interact with the cloud may impact the entire system or conflict with each other. Too many sensors adding data to the cloud and searching for information in the cloud can reduce performance in the cloud, both to receive new data and to search for information in the data.

Having metrics about the current status of the cloud and how the operation of each sensor impacts cloud performance can assist sensor planners in making informed sensor scheduling decisions.

3.3 Prediction

Optimal sensor planning depends in part on predicting future sensor data collection needs to give planners time to re-plan sensor schedules, and to predict the impact on cloud performance as sensor collect tasks start and end.

Existing best practices and doctrine have rules describing what data and information should get collected under specific conditions. When these conditions exist within files in the cloud, we express production rules to match sets of conditions with sensor data needs. We monitor these files for these conditions and determine if any production rules fire. A rule that fires generates a sensor data need that ultimately gets routed to a sensor able to collect the needed data and that sensor gets scheduled.

Sensors write or stream data to files during the mission. Sensor planners might expect these files to get modified or accessed within certain time intervals. We can use this information provided by the file system to identify both unneeded data and malfunctioning sensors.

A sensor data file that does not get accessed within an expected time interval might indicate that a need no longer

exists for the data and that the sensor planner can stop the sensor collection task. Ending the sensor data collection task stops the sensor from adding unneeded data to the cloud which consumes cloud memory and resources needed by other sensors and activities.

A sensor data file that does not get modified within the expected time interval might indicate a malfunction in the sensor. This will require the sensor planner to schedule a replacement sensor to resume data collection.

We can collect data of cloud performance when a sensor adds data to the cloud or searches data in the cloud and use this data in a learning system to predict the relative change in cloud performance when the sensor starts and stops.

3.4   Data Discovery

During the course of a mission, new data sources may get added to the cloud. Time may not exist for sensor planners to become aware of the characteristics of these new data sources and match them to sensors and information needs. We can make a step in this direction by developing a data recommendation component to notify sensor planners that a new data source exists and it matches the kinds of data sources that satisfy particular information needs.

We can apply variations on standard document clustering algorithms [2] to data sources, either on the content itself for text documents or on the (semi-) structured meta-data for other data sources. New data sources found to have content and structure similar to known data sources will receive a possible match to the information need that depends on similar data sources. Sensor planners then have the option to include this data source in their data collection activities.

3.5   Knowledge Representation

Discovering information and managing information flows efficiently depends on having a shared, formal representation of the concepts, entities, and data. This applies to descriptions of sensor needs, sensor data products, and to sensor data collection requests.

An ontology represents a formal representation of a knowledge domain. Such a formal representation provides a vocabulary of terms, a hierarchy of concepts, relations between entities, and properties of these entities [3]. The Web Ontology Language (OWL2) allows us to express an ontology using description logic, a subset of first-order logic [4], that makes it possible both to query for explicit facts and to infer with automated reasoners information implied by the ontology but not explicitly encoded [5].

The kinds of things data consumers need to express sensor data needs include

```
owl:Class DataNeed
owl:Class DataConsumer
owl:DatatypeProperty hasDataCharacteristics
owl:ObjectProperty hasGeographyArea
owl:ObjectProperty hasTimeRequirement
owl:ObjectProperty hasDataNotification
owl:DatatypeProperty hasPriority
```

The ontology should express meta-data for sensor data products. For each class of sensor data product, the meta-data describes such things as the class of sensor able to collect the data, the data schema, and a URL of the location of the data product.

```
owl:Class SensorDataProduct
owl:Class Sensor
  Aircraft
    Piloted
    UAV
  SocialMedia
    Twitter
owl:Class DataSchema
owl:DatatypeProperty hasURL
```

In addition to providing classes, relations, and properties to express individual sensor data needs and products, the ontology should allow for expressing relations between a sensor data need and the kind of sensor that can collect data to satisfy that need. We include in the ontology statements that relate classes of information needs to classes of sensors able to collect the information.

```
owl:Class NeedAB_Sensor1
  owl:equivalentTo (DataNeedA or DataNeedB)
  owl:subClassOf (hasCollector some Sensor1)
```

4   Implementation

Here we discuss the applications and APIs we believe useful for implementing the ideas we have presented in this research.

Hadoop [6] provides the framework for implementing the data cloud itself. Hadoop offers an open-source an open-source framework for coordinated clusters of computers to process large distributed data sets. Data processing occurs via map-reduce jobs. These jobs get mapped to each node in the cloud and process a subset of the data locally. Results get returned to their parent jobs that aggregate and reduce the data before passing it back to their parents. This process continues until the final results get returned to the user.

Protégé, the ontology editor and knowledge-base system [7], provides a tool for developing and maintain the ontologies needed to describe information needs and sensors, and to reason about their relationships.

The Hadoop subproject Chukwa [8] provides an API for collecting cloud performance data from system logs. Chukwa allows us to collect performance data from nodes in the cloud and use these to generate performance metrics.

Drools Expert [9] provides an API to write production rules and apply them to data. We integrate Drools Expert with Hadoop map/reduce jobs in the information discovery engine. This allows the information discovery engines to apply business logic to their searches for new information needs and sources.

Events occur duing missions that might affect the sensor planning. Knowing that a particular event occurred allows us to in some cases anticipate the information needs the event creates or negates. Drools Fusion [10] allows for complex event processing that includes temporal reasoning and we use this to implement event processing rules for predicting information needs.

Mahout provides a machine learning library for Hadoop [11]. Mahout offers a number of machine learning algorithms, including ones for a recommendation engine and document clustering. Our data discovery engines use the Mahout library to mine the cloud for new data sources similar to known data sources and recommend these to satisfy information needs.

Our sensor planning module uses the Drools Planner, an API to an embedded planning engine [12], to schedule sensor data collection activity. The Drools Planner provides a suite of efficient algorithms for solving large and complex planning problems involving many hard and soft constraints.

5   Discussion

We have presented a solution to the problem of sensor planning for large-scale disaster response and recovery. Our solution implements cloud-based predictive analytics that 1) apply predictive algorithms for real-time distributed discovery of sensor data needs and sensor data products, 2) route sensor data needs to sensors able to collect the data, 3) perform real-time sensor management prioritize unfulfilled information requirements and optimally task sensors to collect this information, and 4) learn and anticipation of future sensor data collection needs.

In addition to these primary objectives, we hope to consider in the future the inclusion of security mechanisms into the representation of information. This will enable sensor planning by many people at different organizational levels. Security markup that includes classification, release groups, and need-to-know will protect sensors from improper use or from information leaking outside its intended scope, thereby reducing bandwidth consumption and general information overload that could create difficulties for disaster response and recovery.

6   References

[1] Wu, Michael. The Big Data Fallacy And Why We Need To Collect Even Bigger Data. TechCrunch, Sunday, November 25th, 2012. http://techcrunch.com/2012/11/25/the-big-data-fallacy-data-%E2%89%A0-information-%E2%89%A0-insights/

[2] Nicholas O. Andrews and Edward A. Fox (2007). Recent Developments in Document Clustering. Department of Computer Science, Virginia Tech. http://eprints.cs.vt.edu/archive/00001000/01/docclust.pdf

[3] Thomas R. Gruber. A translation approach to portable ontology specifications, Knowledge Acquisition, 5(2):199–220, 1993. http://tomgruber.org/writing/ontolingua-kaj-1993.pdf

[4] OWL 2 Web Ontology Language Document Overview. W3C Recommendation 27 October 2009. http://www.w3.org/TR/owl2-overview/

[5] OWL 2 Web Ontology Language Primer (Second Edition). W3C Recommendation 11 December 2012 http://www.w3.org/TR/owl2-primer/

[6] Hadoop. http://hadoop.apache.org/

[7] The Protégé Ontology Editor and Knowledge Acquisition System. http://protege.stanford.edu/

[8] Chukwa 0.4 Documentation. http://incubator.apache.org/chukwa/docs/r0.4.0/index.html

[9] Drools Expert User Guide. Version 5.5.0.Final. http://docs.jboss.org/drools/release/5.5.0.Final/drools-expert-docs/html/index.html

[10] Drools Fusion User Guide. Version 5.5.0.Final. http://docs.jboss.org/drools/release/5.5.0.Final/drools-fusion-docs/html/index.html

[11] Mahout. http://mahout.apache.org/

[12] Drools Planner User Guide. Version 5.5.0.Final. http://docs.jboss.org/drools/release/5.5.0.Final/drools-planner-docs/html/index.html

Association Discovery Framework in WebTAS

M. Heidi McClure[1], **Larry Rose**[1], **Roger J. Dziegiel**, Jr[2]

[1]Intelligent Software Solutions, Inc
Colorado Springs, CO 80919
[2]Air Force Research Laboratory (AFRL)
Rome, NY 13441

Abstract—*Association Discovery Framework (ADF) allows rules engines to be integrated with WebTAS so that the rules engines may associate related records. The associations, or relationships as they are known in WebTAS, include system generated confidence values and also allow for user specified confidences. Filtering may be applied to views in WebTAS so that only records with a user confidence are honored, so that only confidences above a threshold are displayed and so that a combination of user and system confidences are used. This paper describes one implementation of this framework that uses the Seer complex event processing system as its rules engine.*

Keywords: WebTAS, Seer, complex event processing, association discovery

1. Introduction

WebTAS, the Web-Enabled Temporal Analysis System, is software that provides data access, query, visualization, analysis and reporting capabilities to government customers. The Association Discovery Framework (ADF) provides WebTAS with an optional server component that searches for related records based on customizable rules or complex event processing (CEP) implementations. ADF is able to examine data, searching for results that match defined patterns. Once discovered, these results are captured by ADF with a relationship or association being created between related records. The relationship between the records is given a confidence or score between 0 and 100% that identifies the system's confidence in the existence of the discovered relationship. Since analysts do not generally trust everything computers tell them, ADF also provides the ability for users to add their own confidences to any relationship.

The rest of this paper will provide some background about WebTAS, about its ability to capture and visualize relationships between records, about its ability to capture confidences both in relationships and in records, about the Seer complex event processing system that is a component of WebTAS and about the named entity extraction capability that is also a component of WebTAS. Next, this paper will discuss the ADF design including server components, its configuration, its pluggable nature and its visualization enhancements for confidence filtering. Then this paper will provide discussion of a practical example that has been demonstrated using the ADF and Seer

as a reference implementation of a rules engine. Finally, some conclusions and ideas for future work will be discussed.

2. Background

2.1 WebTAS

Web-Enabled Temporal Analysis System (WebTAS) is described in [1]. WebTAS allows visualization and analysis of data that may come from many disparate sources. The sources may be, for example, relational databases, live streams of data or text data from files on a file system. In addition to being able to map in and visualize data from external sources, WebTAS also contains a native database which may be based off of many standard database solutions - Oracle, Sybase, SQLServer, PostGres, Access, etc. The WebTAS native database provides a place for users to create additional supporting database tables and to collect any other information of interest to them. In some installations, only the native database is used, but usually a mixture of native and external sources are viewed and analyzed in WebTAS.

Once the mapping of data sources is complete, users may visualize data in tables, timelines, graphs, charts, maps or link analysis charts. WebTAS includes a rich access control capability that is role based and may be tied into existing LDAP environments. The controls may grant or deny users' access to types of data and may grant or deny users' ability to run parts of the application.

In addition to basic visualization of data, WebTAS also supports a plug in architecture and is delivered with additional optional features like text categorization (also known as text classification), text clustering, named entity extraction (discussed in Section 2.5), Seer (discussed in Section 2.4).

2.2 Relationships

Any record viewable in WebTAS may have a relationship generated between that record and any other record. This means that any two records including records that reside in different databases may be related using WebTAS's relationship functionality. The relationship records are stored inside WebTAS's native database and contain enough reference information to link to the associated records - that is, the relationship records contain primary key information along with the data source and

table of the related records. WebTAS also supports data driven or derived relationships which are similar to database joins where relationship definitions describe how to relate records based on existing data - for example, if two tables contain automobile make, model and year, relationships may be defined that will associate matching records in the different tables where the make, model and year of the vehicles match.

When relationships are available, data may be visualized in link analysis charts. These are charts where nodes represent the records and the lines or links between the nodes represent the relationships. Link analysis charts help in visualizing how data are related and are the primary visualization tool for network analysis.

2.3 Confidences in WebTAS

A confidence is a number between 0% and 100% representing the system's or user's impression of how correct or accurate the data is - their belief in the existence of the data. In WebTAS, these confidences may be added to any record that WebTAS can see. Since relationships stored in the WebTAS native database are just another type of record, they, too, may have confidence values. In other words, both records and the relationships between them may have confidence values associated with them. There are two types of confidences - System and User - ADF creates the system confidences and users of WebTAS add or modify the user confidences. Confidences may be applied to both native and external records.

2.4 Seer

Seer is described in [4]. Seer is a fuzzy complex event processing system which means that it can describe and search for patterns in data that may include patterns in temporal and geo-spatial data in addition to patterns in its other data (non-spatial, non-temporal). Being fuzzy means that Seer does not need to perform exact matches when comparing data in time and space. Seer helps intelligence analysts make sense of complex data sets and when used with WebTAS, the data sets may include data from many disparate sources. Seer supports prediction using Bayesian reasoning.

Seer is able to represent models that use more complex probabilities or ones that use fairly simplistic confidence increments or factors when reasoning on event states. That is, the assessment strategy used may be probability based, confidence increment based or confidence factor based. Note that Seer confidences are different from the confidences in WebTAS described earlier. Confidences in Seer are a part of the model describing the pattern of interest. Each event state may have a confidence or probability setup to help with Seer's calculation of a total confidence for a pass through the model which is also known as an assessment. The assessment confidence will be used by ADF when Seer is configured to be a processing

engine but the Seer confidences and WebTAS confidences are different entities.

Probability based reasoning uses Bayesian algorithms for calculating probabilities based on each state transition having a probability based on the success (true state) of the preceding state [2]. Confidence increment reasoning sums each event state's confidence and presents a confidence number for the overall success of the model. In this case, negative confidence increments are also allowed and basic logic is used when there are alternative paths through the model (ORs) or collaborative paths (ANDs). The logic for confidence increments is to simply add the confidences and when there are ANDs use the lowest confidence sum along a branch and if ORs, use the highest. For confidence factors, a slightly different logic is used when there are missing states. The equation for confidence factors is $PrevConf_T + CurConf_S - (PrevConf_T * CurConf_S)$, where $PrevConf_T$ is the total previous confidence and $CurConf_S$ is the current state's confidence.

In addition to Seer computing a confidence or probability number for the current state of a model, alerts may also be generated from Seer so that timely and perhaps preemptive actions may be taken when confidence or probability has passed a threshold number.

2.5 Named Entity Extraction

WebTAS contains an optional plug-in for performing named entity extraction on text fields in databases or on text documents. Examples of named entity types are people, organizations, dates and locations. Once extracted, the named entities are stored in an "Entity Sources" table with links back to the documents or records that contained the named entity. The named entity extraction plug-in supports various entity extraction implementations including one based on GATE [3], one based on SRI's PAL Semantic Extraction[1] and one on Janya's Semantex[2].

Named entity extraction is just one way to make more sense out of free-text data. By having the extraction find people and organization names, for example, analysts are able to reduce the numbers of documents they must read as they do their analysis. The ADF example described in Section 4 uses the results of named entity extraction performed on messages.

3. Design

As noted earlier, ADF adds to WebTAS the ability to discover relationships between records and provides visualization enhancements that assist the user in managing the discovered relationships. The general architecture of ADF is as shown in Figure 1. The server components include processing and

---

[1]https://pal.sri.com/Plone/framework/Components/learning-methods/semanticextraction

[2]Janya is no longer in business
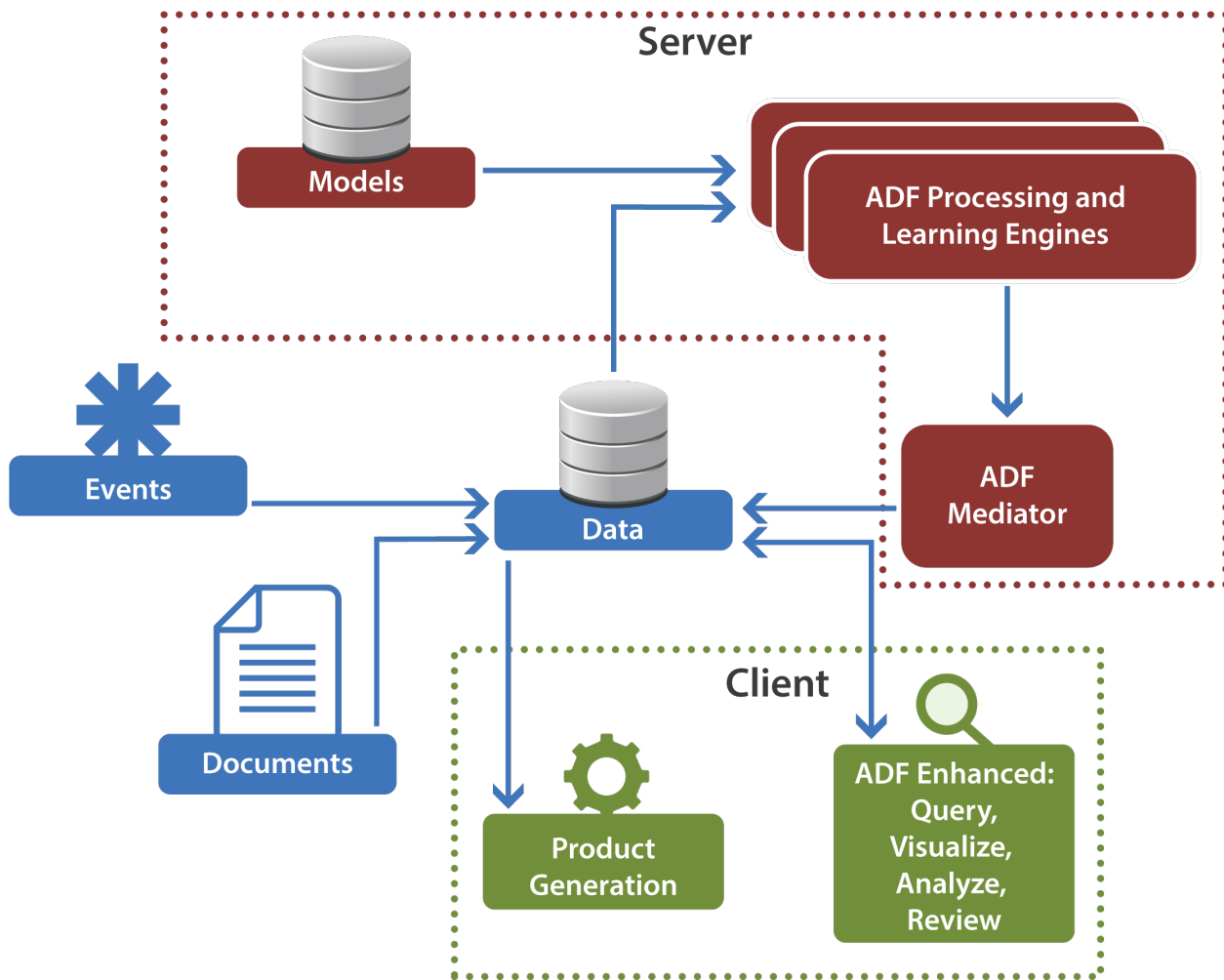
Fig. 1: Association Discovery Framework

learning engines which adhere to interfaces that the ADF mediator understands. The role of the ADF processing and learning engines is to discover the related records of interest. The ADF mediator takes information from the processing engines and creates WebTAS relationship objects based on mediator configuration. The client enhancements to support ADF include filtering of visualized results based on WebTAS confidence levels - either user-only confidences or a combination of user and system generated confidences. Intelligence analysts are skeptical of computer discovered relationships, so it was very important to provide a way to filter relationships that humans had not yet reviewed and validated.

The rest of this section discusses some of these components in more detail and also discusses the configuration and pluggable capability of ADF.

3.1 ADF Mediator

The ADF mediator is a server component that monitors model results and creates relationships appropriately (see Section 3.3). In our initial implementation, a model is directly related to a Seer model. The mediator is also able to examine relationships that have already been created and will update them based on current model execution results. In addition to creating the relationships, additional information is created by the mediator that captures the pedigree of the relationship. This includes the name of the processing or learning engine's model that created the relationship and the explanation of the relationship.

The ADF mediator runs as a deployed application in a JBoss
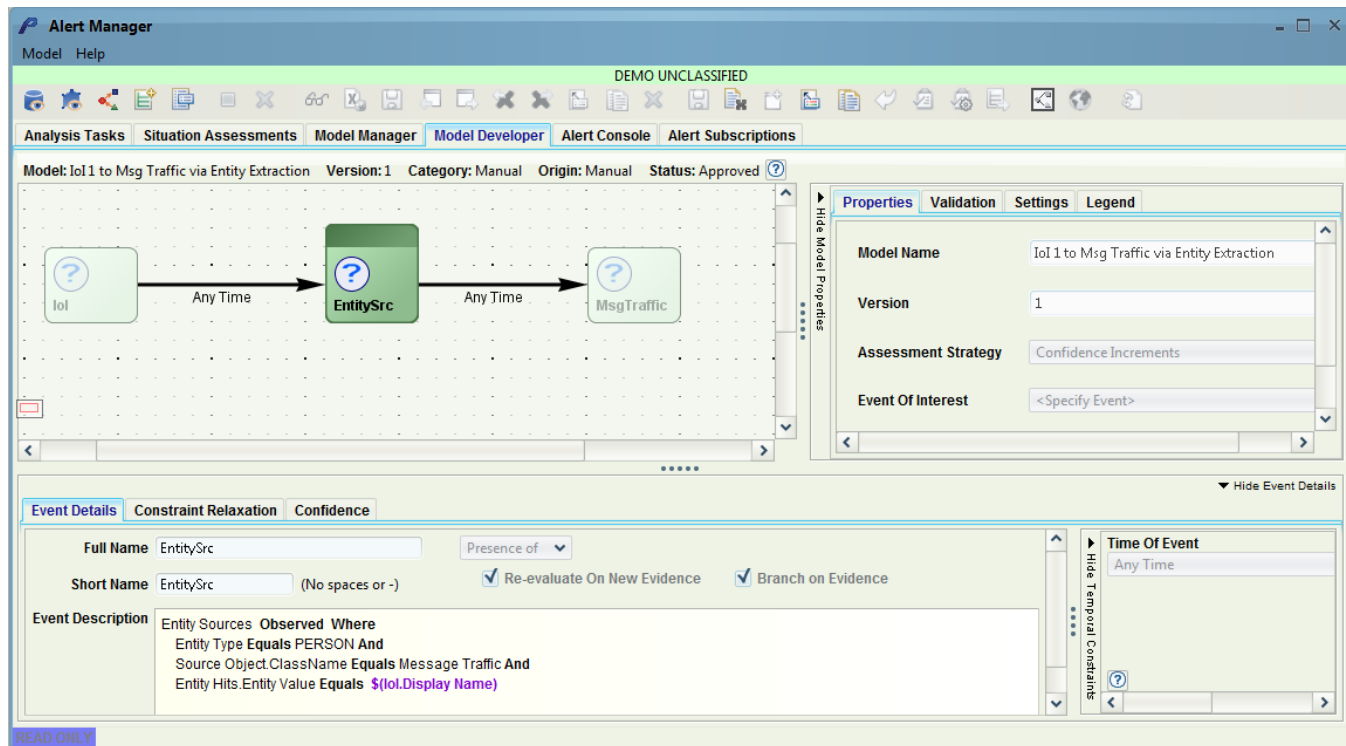
Fig. 2: Seer Model

application server [3]. The mediator is configured using Spring (described in Section 3.3) and is intended to connect to one or more processing or learning engines. Seer is an example of a processing engine.

3.2 Models

The ADF processing and learning engines are assumed to use a model (like a Seer model) that will present to the ADF mediator suggestions for relationship creation. Each model responsible for one type of relationship between two record types, for example, between people and message traffic. Multiple models may create the same type of relationship between the same types of records. Many models may be running that create a variety of relationships between records.

3.3 Configuration and Being Pluggable

For relationship configuration, at start up, the ADF server reads a configuration file which specifies models, record types and the relationships to create between the records. The ADF uses the Spring Framework[4] and the relationship configuration is based off of constructor type injection in Spring. Additionally, the Spring Framework is used to allow a pluggable framework so that various processing or learning engines may

be used. In our current implementation, the Seer complex event processing environment is plugged into the ADF framework using the Spring configuration. Other supporting pieces of the ADF mediator are defined and configured in Spring.

The ADF allows for additional rules or processing engines to be placed into the system. The idea is that Seer is an initial implementation of a processing or learning engine. Other engines based perhaps on JBoss BRMS (previously Drools)[5] or Jess[6] may be plugged into the ADF to provide alternative means of identifying related objects. To plug in additional processing or learning engines, each engine will need to provide a small amount of code that implements the interfaces that the mediator understands. Then the system is configured to know about the new engine.

3.4 Client Enhancements to Support Confidence Filtering

WebTAS has been enhanced so that it provides better ways to filter based on confidences and better viewing of related object details. Figure 3 shows a details pane for showing details of relationships including both analyst and system or calculated confidences and Figure 4 the display after confidence filtering has been applied to only show analyst verified confidences

---

[3]http://www.jboss.org
[4]http://www.springsource.org/

[5]http://www.redhat.com/products/jbossenterprisemiddleware/business-rules/
[6]http://herzberg.ca.sandia.gov/
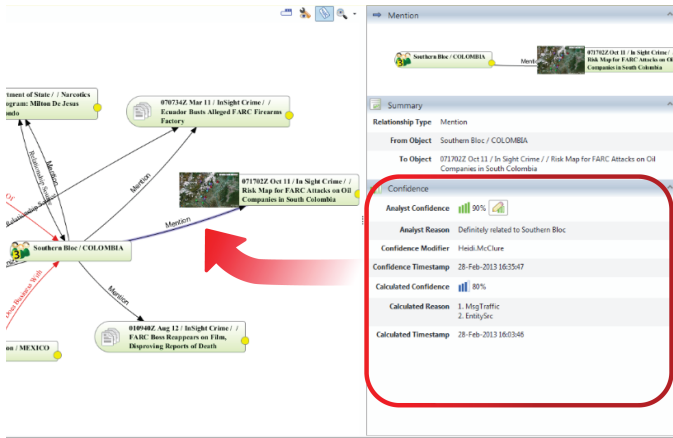
Fig. 3: Confidence Filtering and Details Pane
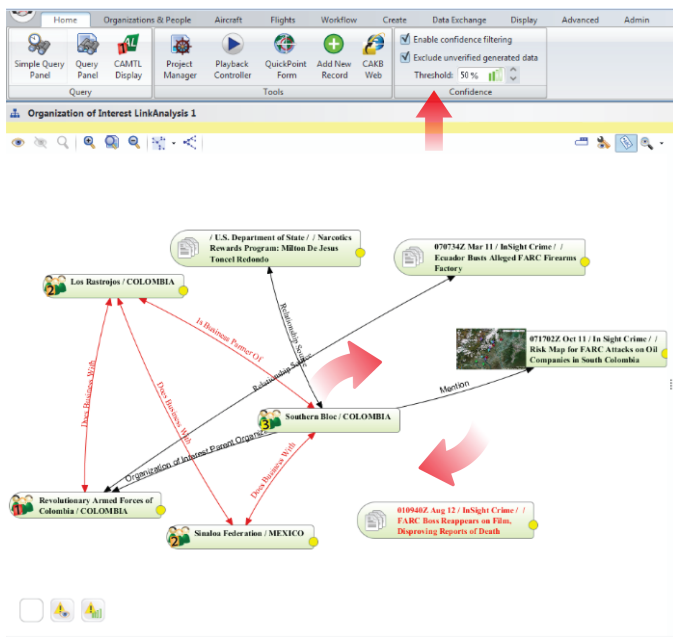
Fig. 4: Confidences Filtered

over 50%. Notice in this figure, there are missing "Mention" relationships. These are relationships that the analyst has not yet verified and that have been filtered.

4. Application with Extracted Entities

This section describes an implementation of the ADF framework that uses Seer as a processing engine. The goal of this implementation is to create relationships between Identities of Interest (people) and Message Traffic records that mention those people. In this case, the Identities of Interest have a primary name and a list of aliases. The models will be configured to place more confidence on a primary name match

than on an alias match. Models will look at all known names when trying to find matches in the Message Traffic.

With this ADF implementation, analysts can be readily directed to the Message Traffic that mentions the people they are looking for. ADF does not replace the job of reading the messages, it only helps to point the analyst to documents that may be of most interest to their current focus.

4.1 Seer Model for Person to Message Traffic

This section will describe what Seer models look like using an example that will identify messages (Message Traffic) that mention people's names based on previously performed named entity extraction.

The model first looks for the names of the people we are interested in finding in the extracted information - these people are stored in an Identities of Interest table in the system. Second, the model looks for those names in the named entity extraction records. Third, in this last step, the model looks for the actual message that mentions the person.

In Figure 2, the Seer model builder is displayed. The boxes and arrows represent the three steps just described. The first state labeled IoI does a query "Identity of Interest Observed". The second state does a more complex query "Entity Sources Observed Where Entity Type Equals PERSON And Source Object.ClassName Equals Message Traffic And Entity Hits.Entity Value Equals $(IoI.Display Name)". This is a query of the Entity Sources records. Entity Sources are the named entity extraction results. This query is constrained to only look for PERSON named entity types that are referring to Message Traffic records and that have the named entity value that matches the Identity of Interest queried in the first state. The $(IoI.Display Name) syntax means that the query, at run time, will refer to the results of the IoI event state (the first state) and will use the Display Name attribute of the Identity of Interest. The third state uses the "Source Object ID" information from the "EntitySrc" state (second state) in order to find the actual Message Traffic record that mentioned the Identity of Interest's name. The third query looks like this: "Message Traffic Observed Where Object Id Equals $(EntitySrc.Source Object.ID)".

This model uses the Display Name of the identity. Another model we use looks at the list of aliases for the identity following similar logic to identify any messages that mention the aliases of the identity.

The assessment strategy used in this Seer model is the confidence increment strategy. Here, each state may be given a confidence which will be summed to come up with the confidence in the instance of model execution. For example, if we have an exact name match, we may place an 80% on the last state. If we are running the model based on aliases, we may place a smaller confidence, say 60%, on the last state.

Since all states in the model must be met for success, the first and second states may be left at 0% confidence.

4.2 Configuration

To create the correct "Mentions" relationship, the mediator is configured to look for Seer-based ADF results where there are records of type "Identity of Interest" and of type "Message Traffic" and when they are found, it will create a "Mentions" relationship between them. As part of the relationship creation, the mediator will also get the system confidence identified by the Seer-based ADF processor. That is, as described in Section 4.1, 80% if it is a primary display name match, 60% if it is an alias match.

4.3 Visualization

When an analyst sits down at their WebTAS station, they are able to see the system generated relationships in a couple of places. The first is in the link analysis charts. Refer to Figures 3 and 4. The Mention relationships (lines between nodes) have all been generated using ADF. When a link is highlighted as shown in Figure 3, the details of this particular relationship are visible in the details pane on the right of this display. The user may right click on the Analyst Confidence and change their assessment of the relationship. As shown in the right side display, the analyst may also choose to show only those records someone has reviewed and validated - that is, only show records that have an Analyst Confidence assigned. When they filter this way, some of the Mention links between nodes disappear..

When analysts drill down into the details of a record, they will have a related objects tab as shown in Figure 5. This figure shows an example of an organization record ("Southern Bloc") with its related objects tab highlighted. In this view, users can examine the related object (in this case, "In Sight Crime") and expand the type of relationship ("Mention") to see more information about the pedigree of this relationship. This is the same relationship highlighted in the details pane view in Figure 3.

5. Conclusion

We have presented our current work that provides an association discovery framework (ADF) as part of WebTAS. ADF assists users by discovering relationships between records in the user's domain of data. The computer generated relationships are given a system confidence so that the user may distinguish between relationships that the ADF-created and ones that users have vetted. Because the ADF is pluggable, this framework provides an environment where other processing, rules or learning systems may be integrated and analyzed.

6. Future

Now that the basic framework is in place, we would like to try other rules engines like Jess or JBoss BRMS and perhaps test out other learning algorithms or association mining algorithms.

Acknowledgments

References

[1] Webtas overview, 2013. Intelligent Software Solutions - http://www.issinc.com/programs/webtas.html.
[2] D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.
[3] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. Gate: A framework and graphical development environment for robust nlp tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.
[4] M. Gerken, R. Pavlik, C. Houghton, K. Daly, and L. Jesse. Situation awareness using heterogeneous models. In *Collaborative Technologies and Systems (CTS), 2010 International Symposium on*, pages 563 –572, may 2010.

Fig. 5: Related Objects Tab

Temporally Aligning Clusters of Social Media Reaction to Speech Events

Brian Amanatullah, Greg Barish, Matthew Michelson and Steve Minton

InferLink Corporation

2361 Rosecrans Avenue, Suite 348

El Segundo, California 90245

Abstract— *This paper describes our approach to analyzing social media response to speech events, such as the President's State of the Union address. Our approach clusters Tweets by topic and in time, and this paper specifically focuses on analyzing the temporal dynamics of the topically clustered Tweets. We classify topical clusters as either being temporally clustered around a specific period of time or not (e.g., referring to the whole speech). For instance, a topical cluster related to a specific line in the speech is likely clustered in time as well. In contrast, a cluster about the speakers suit could be composed of Tweets written anytime during the speech. We demonstrate the approach on a number of real world data sets, and show that an analysis of these temporal dynamics can lead to a structuring of the social media responses that can support deeper analysis than just topical clustering alone.*

1. Introduction

Recently, the analysis of social media and politics has become an interest in the data mining community. However, much of this previous work focuses on the sentiment of the social media reaction (e.g., [1], [2], [3]). In this work we take a different approach, focusing instead on finding the points of the speech that generate the most reaction and the topics that correspond to these spikes in activity. Specifically, this paper presents an approach for analyzing social media responses to live speech events, such as the US President's State of the Union address. In particular, we focus on Twitter users' responses to the speech by analyzing their Tweets during the duration of the speech. We designed the approach to allow analysts, social scientists, and policy researchers to measure public reaction to various talking points (and visuals) in a speech. One can measure which points generate the most reaction, including those that may be surprising to the speaker and his/her staff. Further, marketers, journalists, political junkies and the general public can also use the system (and its related, public facing website) to better understand the speech's effect on different groups of people, their opinions, etc.

Our method clusters Tweets both topically and temporally. By topically, we mean that Tweets are clustered because they refer to the same subject. For instance, one cluster of Tweets discusses the "President's tie," while another cluster focuses on "universal pre-school education." If a cluster refers specifically to a line (or lines) in the speech, we call this cluster "referent." This is in contrast to clusters whose topic is not directly related to the speech itself, such as reaction to what the viewers are seeing on the screen at that moment (e.g., the President's tie). We call these clusters "non-referent." By temporally clustered, we mean clusters whose Tweets exhibit "burstiness" within a short interval of time. For instance, if many of the Tweets in a cluster happen to fall within a short time-window of one another, we call them "temporal." This usually happens when there is reaction to something specific in the speech's time-line (such as a particular line in the speech or non-referent event). In contrast are "non-temporal" clusters, which are clusters that refer to the speech generally. For instance, a cluster where people are simply noting that they are watching a speech seem to occur at various points throughout the speech, and do not exhibit the similar "bursty" behavior. By definition, we note that non-temporal clusters are also non-referent.

By aligning Tweets both topically and temporally, we can then overlay the topic-clusters onto the time-line of the speech itself to get a two-dimensional representation of responses to the speech. Then, for a given time in the speech, we can see the major topics of discussion at that particular time. For instance, if the cluster is Temporal/Referent, then we know it refers to that part of the speech at that time, and therefore that part sparked social reaction. If a cluster is Temporal/Non-Referent, then something in the broadcast outside of the speech itself, such as what is on-screen at that time, prompted reaction. Finally, we can exclude Non-Temporal clusters from the timeline analysis, since they would provide broad color (possibly), but not provide much deeper temporal analysis. The key then is to classify the clusters.

This paper focuses on the particular classification task of determining whether a cluster of Tweets is temporal or non-temporal.[1] Intuitively, our algorithm works by assuming that temporal clusters will exhibit bursty behavior, while non-temporal clusters will not. For instance, if during a speech the speaker says something controversial or resonating, then we assume lots of Tweets on that specific topic will be generated, and therefore should be grouped in time (e.g, classified as temporal). The key to the classifier, therefore, is to define "burstiness" and then measure it. We define such a

---

[1] While there exists a body of past research on determining topical-clustering of text, for example, k-means clustering [4], Latent Dirichlet Allocation (LDA) [5], Hierarchical Agglomorative Clustering (HAC) , (e.g., [6]) among others, that is not the main focus of this paper.

measurement and show that it can form a classifier for defining temporal and non-temporal clusters, and that indeed it can even find non-temporal clusters that are also non-referent.

Table 1 makes the clustering clear with a few examples taken from the 2012 State of the Union speech. The first cluster in the table is non-temporal. It reflects a number of Tweets from users watching the State of the Union speech, and Tweeting that they are doing so. The Tweets occur at various times throughout the speech, and do not, as a whole group, refer to a specific time period in the speech. The second cluster reflects Tweets about a specific topic that occurs at a specific time period in the speech. That is, the cluster is both temporal and referent (it refers to a line about the auto industry). The final example cluster in the table is temporal but non-referent. Instead, the time period reflected by the cluster is a point during the speech when the camera showed President Obama giving Gabrielle Giffords a hug.

Our overall approach for analyzing the social media response to a speech is given in Figure 1. While we focus on the temporal classification aspect in this paper, briefly, the full system works as follows. During a speech, the system sources, collects and then cleans a set of Tweets. The Tweets are then clustered by topic, and also broken down by cohort, where each sub-cohort represents a group of users responding (for instance each cluster is further sub-divided into cluster members provided by men and those provided by women). Finally, the Tweet clusters are aligned temporally. This temporal alignment is the temporal classification we will focus on in this paper. Once the data is processed, we display the results on a webpage[2] where users can explore and analyze the results.
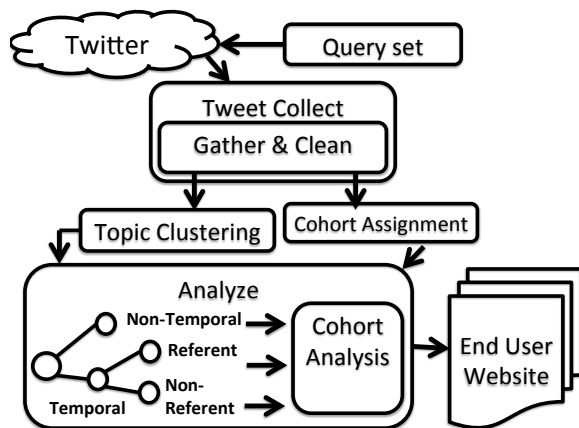
Fig. 1

THE 2SR ARCHITECTURE

As stated above, we implemented the approach in a public-facing website, where users can select various speeches and

analyze the output themselves. One of the important aspects of our user interface is the alignment of the clusters to specific times and parts of the speech. This enables users to dig into the social media reaction at specific times of the speech. The Website is shown in Figure 2. The top of the figure shows a timeline with vertical bars. The bars represent clusters linked temporally to that part of the speech. The size of the bar reflects the relative volume of Tweets such that a taller bar reflects more Tweets (larger cluster) than a smaller bar. At a glance, this allows users to zoom into the sentences of the speech that generated the most reaction. In the figure, a user clicked on one of the taller bars (highlighted in grey) and the site automatically scrolled to the line in a transcript of the speech reflected by the time of the cluster. On the bottom right, example Tweets from the cluster scroll by the user to examine.

The rest of this paper is organized as follows. Section 2 describes our approach in detail, Section 3 presents our results and discussion, and Section 4 contains our conclusions and future directions for this research.

2. Temporal Clustering of Social Media Reaction

In this section, we detail our approach to temporal clustering. As we stated above, we assume that Tweets are already clustered by topic, and that forms the input to our process. Before we give the algorithm a more formal treatment, we discuss the intuition behind the approach. Intuitively, a set of responses to a particular item in the speech will cluster around (though after) that item in the speech. That is, assume we have a given cluster, and assume that we choose to define bursty behavior as having most of the tweets in a cluster fall within a 4-minute window. Then, we can analyze a time-line of all of the Tweets that belong to the cluster, and if some defined proportion (such as simply the majority) of them falls within the rolling window, we would say the cluster exhibits bursty behavior. This situation is shown in Figure 3.

In the figure, we see the Tweets that define the cluster, $\{T_0, \ldots, T_{10}\}$. Each Tweet is then aligned in time, and the timeline is shown along the bottom of the figure. Since our given rolling window size is four minutes in the example, the figure also shows example rolling windows as horizontal bars across the top, along with the range they represent. The first cluster, 8:00 to 8:04, contains 8 of the 10 cluster members and we therefore classify this cluster as temporal. If none of the rolling windows for a given cluster satisfy the constant that the proportion of Tweets contained in the window is above the proportion threshold, then we classify the cluster as non-temporal.

Therefore, we define define bursty clusters as those that contain Tweets that flare up in a specific time range, and more or less remain local to that time range. In some sense, one can think of members of a bursty set as having a tight temporal relationship. We define a tight temporal relationship as one where a majority (or a proportion above a threshold) of the

Table 1

DIFFERENT CLUSTER CLASSIFICATIONS

Non-temporal cluster
RT @tjholmes: Unless ur watching CSPAN, u might not know President of the United States is delivering State of the Union address n 50 mi...
What did you think of President Obama's State of the Union address?
#NowWatching President @BarackObama's "State Of The Union Address"...& you should be too!

Referent & Temporal cluster: reflects specific part(s) of the speech
Reacting to the line: "And tonight, the American auto industry is back"
#SOTU - Cites General Motors return to #1."Tonight, the American auto industry is back" -applause.
The American auto industry is back. #manufacturing #SOTU
The American auto industry is back! (to making cars that last 30k miles) #SOTU #honda4life
RT @DPerkDT: Republicans initially opposed the bailout of the American auto industry, now applaud jobs created. #SOTU
"The American auto industry is back. What is happening in Detroit can happen in other industries." -POTUS #SOTU
Pres. Obama says General Motors is "back on top as the world's number one automaker....the American auto industry is back." #SOTU #Detroit

Non-Referent & Temporal: Does not reflect specific part(s) of the speech
#whchat..just saw the President embrace Gabby. I'm grabbing my tissue?snif...So good to see her.
RT @jbarro: Obama hugging Gabrielle Giffords very sweetly.
#SOTU it was nice seeing President Obama hug Gabrielle Gifford.
Obama hugs Rep. Giffords. #SOTU #WHTweetup
President gave a huge hug to Gabby Giffords. How sweet. #sotu
Obama and Gabby Giffords hugging...#tearjerker
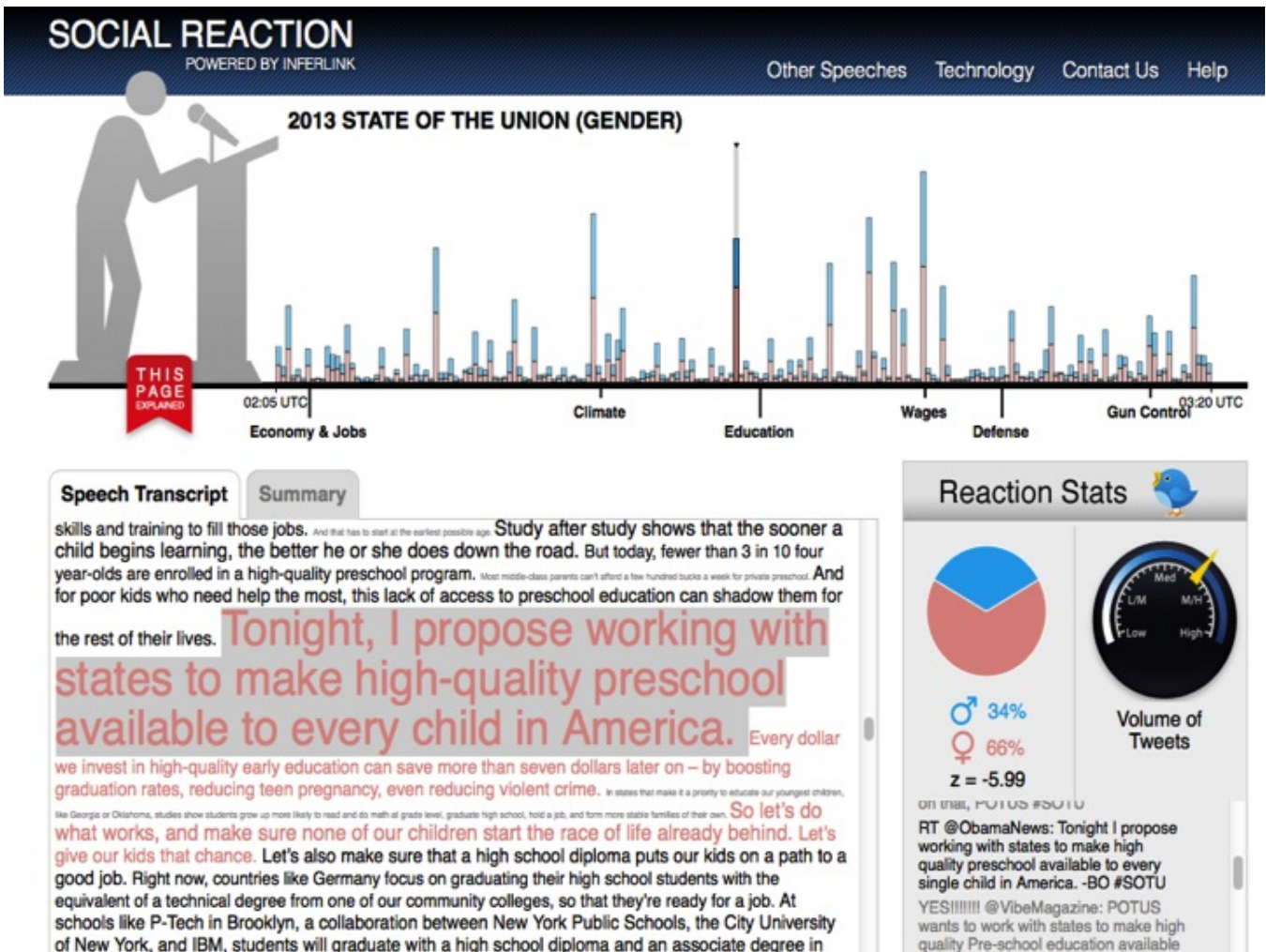
Fig. 2

THE SOCIALREACTIONGROUP WEBSITE
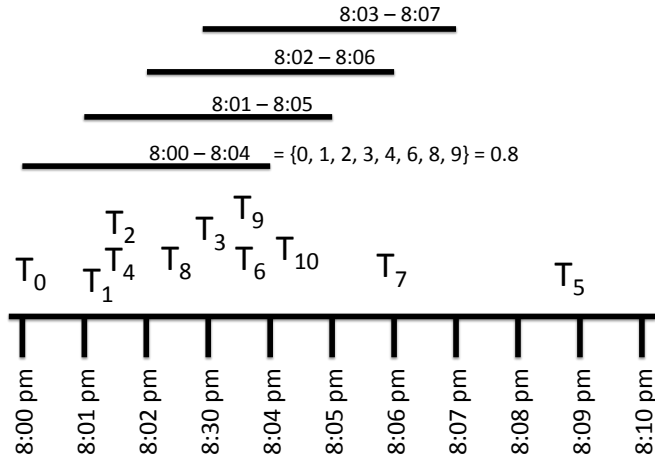
Fig. 3

A CLUSTER OF TWEETS ON A TIME LINE

cluster members (Tweets) occur within a short time period. This implies two parameters: the window size of the short time period, and the proportion of members in the cluster that must occur within that window. This definition allows us to apply a discrete measurement of tightness. Namely, the proportion of cluster members that fall within this window. This forms the basis of our classifier, if the proportion is above a threshold, we define the cluster as temporal, and otherwise not.

More formally, given a rolling window size, W, and a set of topical clusters, $\{C_0, \ldots, C_n\}$, each of which, C_i, is defined by the Tweets $\{T_0, \ldots, T_n\}$ contained within it (noting that each T_j also has a time-stamp), we define the cluster's temporal tightness as:

$$\text{TEMPORALTIGHTNESS}(C_i) = \frac{\|T_j^{time} \in W\|}{\|C_i\|}$$

Where T_j^{time} is the time differential from the earliest time-stamped Tweet in the cluster, C_i to this Tweet, T_j. Then, given a proportion threshold P_{thresh}, we define a cluster C_i as being temporally clustered if $\text{TEMPORALTIGHTNESS}(C_i) \geq P_{thresh}$.

The whole algorithm is given in Table 2.

Table 2

CLASSIFYING TEMPORAL CLUSTERS

$\forall C_i \in \{C_0, \ldots, C_n\}$
　　If $\text{TEMPORALTIGHTNESS}(C_i) \geq P_{thresh}$
　　　　$C_i \leftarrow$ Temporal Cluster
　　Else
　　　　$C_i \leftarrow$ Non-Temporal Cluster

The key parameter, is therefore the window size (the proportion can simply be a majority), and this has certain impli-

cations. For instance, if a window size is set to a ridiculously large value, then general comments about the whole speech will be assigned as temporally relevant, but the point-in-time they refer to would be the whole speech. If the window size is way too small (e.g., 20 seconds) this would imply that the social response is almost instantaneous. Understanding the properties of this parameter informs a reasonable choice of a few minutes. That is short enough to capture the dynamics of temporal tightness, but long enough to give social media users time to respond to the same point, even if they response time differs. As we show below in our experiments, five minutes is an adequate choice.

One advantage of our approach is that it can discover tightly temporal clusters that refer to something about the speech, but not in the text itself (e.g., Non-referent clusters). For example, in Table 1 the algorithm discovered that the cluster describing President Obama's hug was temporally clustered as well, reflecting that this action had a specific time-period, namely when it was on screen during the televised speech. That is, by examining the temporal dynamics separately from the topical clustering, the approach does not need to take the content of the cluster into account.

3. Experimental Results

Above we described our approach to classifying topical clusters as having a temporal relationship or not. Here we apply our approach to a number of real world speeches and analyze the resulting clusters. We find that indeed, we are able to discern topics that have both a topical relationship and a temporal relationship to the speech.

We analyzed four specific speeches: President Obama's State of the Union (SOTU) address in 2012, Benjamin Ne-
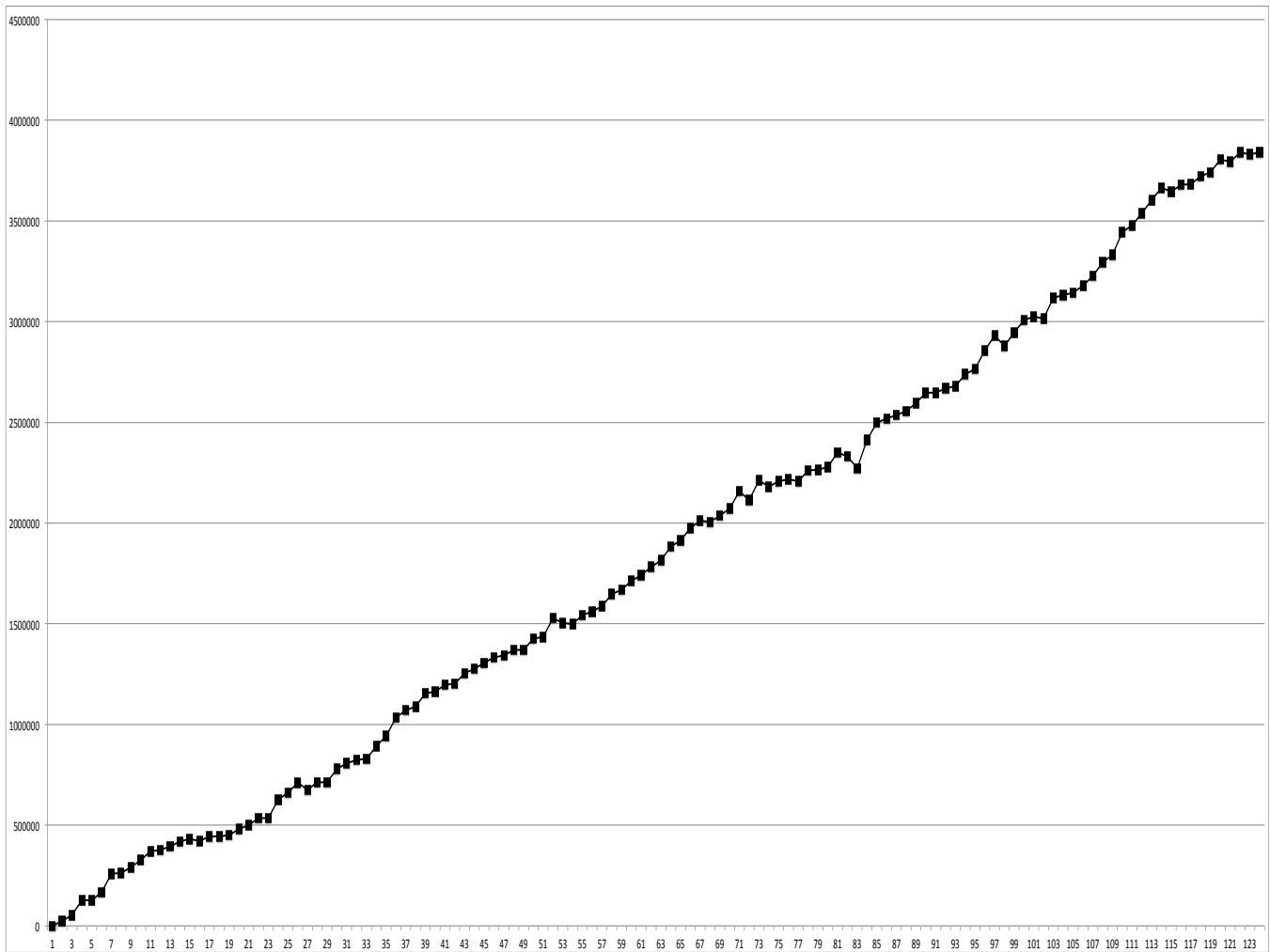
Fig. 4

SOTU 2012 TWEETS VS SPEECH TIME

Table 3

SPEECH DATA FOR ANALYSIS

Speech	Start Time (UTC)	Stop Time (UTC)	Tweets	# Clusters
SOTU 2013	2/12/2013 2:05:00	2/12/2013 3:20:00	145,913	182
Inaguration 2013	1/21/13 16:55	1/21/13 17:12	38,937	55
Netanyahu UN 2012	9/27/12 17:40	9/27/12 18:18	11,800	43
SOTU 2012	1/25/12 2:05	1/25/12 3:25	164,547	124

tanyahu's address to the UN General Assembly in 2012, President Obama's Inauguration address in 2013, and President Obama's State of the Union (SOTU) address in 2013. Table 3 describes the data, showing the start and end time of the speech (in UTC), along with the number of Tweets analyzed during that time period, for the given speech.

As shown in the table, we were able to discover a number of temporal clusters for each speech. We also found a number of clusters that were temporal but non-referent. As we stated above, these may correspond with events shown on the screen during the televised speech. For instance, for the Netanyahu speech, the temporal/non-referent cluster occurred after he showed a picture of a cartoon bomb during the speech. For each speech we found 34, 4, 1, and 15 temporal/non-referent clusters, respectively.

Figure 4 validates our premise that our window size and threshold are reasonable. In the figure, we see a plot of the time of the speech-line of the 2012 SOTU speech and the time associated with the topical cluster assigned to that speech line. The y-axis of the figure shows the number of milliseconds

from the start of the speech, increasing up the axis (origin is the start of the speech). The x-axis is the minutes from the start of the speech of the cluster associated with that line of the speech (e.g., the cluster that represents that line in the speech), again, ordered from the start of the speech. For instance, when Obama says, "And tonight, the American auto industry is back." this is associated with a specific cluster. The figure shows the time of the first Tweet in this temporal cluster (x-axis), plotted against the timeline of the speech (y-axis). Therefore, if the topical clusters were well aligned in time with the speech, then the line should be relatively straight and sloping up and to the right (we assign time as monotonically increasing in seconds from the start of the speech). Indeed, we see this is the case in the figure, and there are very few dips downward, which would signify a mis-aligned cluster in time with the speech.

4. Conclusion

This paper describes an approach for clustering Social Media responses to speech events not just by topic, but also temporally. We present an approach for taking a topical cluster and deciding whether the members of that cluster refer to a specific point in time, or do not. The approach relies on dynamically setting a window of time, within which we consider the set of social media responses to be "temporally tight," and therefore clustered in time as well. Clusters whose members fall outside of this range are non-temporal and may refer a more general time-frame, such as the whole speech itself.

We validated our approach both empirically, and also through a public facing website, where users can analyze social media reaction to speeches themselves, in both the topical and temporal clustering dimensions. In the future we plan to investigate this topic further to try and more deeply understand the temporal dynamics of social media reaction.

References

[1] N. A. Diakopoulos and D. A. Shamma, "Characterizing debate performance via aggregated twitter sentiment," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. New York, NY, USA: ACM, 2010, pp. 1195–1198. [Online]. Available: http://doi.acm.org/10.1145/1753326.1753504

[2] A. Livne, M. P. Simmons, E. Adar, and L. A. Adamic, "The party is over here: Structure and content in the 2010 election," *Proc. of 5th ICWSM*, 2011.

[3] P. H. Calais Guerra, A. Veloso, W. Meira, Jr., and V. Almeida, "From bias to opinion: a transfer-learning approach to real-time sentiment analysis," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '11. New York, NY, USA: ACM, 2011, pp. 150–158. [Online]. Available: http://doi.acm.org/10.1145/2020408.2020438

[4] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 281-297. California, USA, 1967, p. 14.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[6] F. Murtagh, "A survey of recent advances in hierarchical clustering algorithms," *The Computer Journal*, vol. 26, no. 4, pp. 354–359, 1983.

A Cloud Computing Framework with Machine Learning Algorithms for Industrial Applications

Brian Xu, D. Mylaraswamy, and P. Dietrich

Honeywell Aerospace, Golden Valley, MN, USA

Abstract. *In this paper, a novel cloud computing framework is presented with machine learning (ML) algorithms for aerospace applications such as condition based maintenance, detecting anomalies, predicting the onset of part failures, and reducing total lifecycle costs. This cloud framework has been developed by using MapReduce, HBase, and Hadoop Distributed File System (HDFS) technologies on a Hadoop cluster of OpenSUSE Linux machines. Its ML algorithms are based on Mahout ML Library and its web portal is built using JBoss and JDK. Importantly, the big data from various Honeywell data sources are managed by our HBase and analyzed by various ML algorithms. Users can use this cloud based analytic toolset through web browsers anytime and anywhere. More analytic results of using this framework will be published later.*

Keywords: Cloud Computing, Big Data Analytics, Machine Learning, Algorithms, MapReduce, CBM.

1 Introduction

To deal with data overabundance and information overload, big data analytics and cloud computing technologies are being used by the world top IT and software companies such as Google, IBM and Microsoft [1]. Currently these technologies are being adopted by other industries. In this paper, we present our prototype of cloud framework with machine learning (ML) algorithms for aerospace applications, specifically, Condition Based Maintenance (CBM), monitoring, diagnostics, and product reliability and performance. In Honeywell, there are big (volume, velocity, variety) data that are collected and streamed from thousands of aircraft (operational datasets, maintenance data, etc.), test cells (hundreds of sensors and measurements, etc.), and repair shops (records of electronics, avionics, mechanical repairs, etc.). For instance, one test cell can generate 300 MB test data daily per engine. Our approach is to combine the best strengths and synergies of both cloud computing and machine learning technologies, in order to effectively analyze the big data and develop capabilities of predictive analysis, actionable information, better CBM and decision making.

Technically, by combining and leveraging cloud computing and ML technologies, our major goals are included (not limited to): (1) detecting anomalies from parts, components and systems, (2) predicting the onset of failures of parts (e.g., components, LRUs, etc.) to maximize asset usages and availability, minimize the downtimes, and (3) sustaining better and effective CBM policies, and (4) reducing total lifecycle costs of our aerospace assets and networks. Our primary tasks are to realize these goals by analyzing the big data and transforming information into knowledge. In our CBM applications, after we developed our Hadoop cluster by leveraging Apache ecosystems [2], we have focused on analyzing and mining our data sources by using open source ML algorithms including Mahout Library [3] and by developing our ML algorithms using R and Matlab.

This paper is organized as follows: Section 2 describes our cloud-based ML framework and components. Apache Hadoop, MapReduce, and HBase are used to develop an effective cloud computing infrastructure on which our machine learning framework is built, and the technical details are described in Section 3. Mahout ML algorithms are briefly introduced in Section 4. Our conclusions are presented in Section 5.

2 Architecture and components of Cloud-based ML Framework

Our specific tasks are to find valuable insights, patterns and trends in big data (large volume, velocity, and variety) that can lead to actionable information, decision making, prediction, situation awareness and understanding. To complete these technical tasks, we have developed a cloud framework with machine learning technologies for cyber-learning, leveraging machine learning algorithms (SVM, random forests, PCA, K-means, etc.), knowledge mining, and knowledge intensive problem solving.

We developed our cloud-based ML framework, by developing a *Cloud Controller*, *Cluster Controllers*, and *Node Controllers* on our Hadoop cluster of Linux machines. We used Eucalyptus cloud tool [4] to develop our primary software framework. The framework architecture and key components are shown in Figure 1.

In Figure 1, we implemented the HBase that is a scalable, distributed database and supports real-time access large data repositories such as Oracle, MySQL, etc. Currently, we have 5 major HBase tables (more big tables can be created as needed):

1. *CBM_use: This table manages user credentials and access privileges.*

2. *Field_reports: This table contains data from operating assets installed on various aircrafts and operating vehicles.*
3. *ListOfValues: This table contains variables (typically vehicle installed sensors) and sampled historical data. Each data set has a unique timestamp associated with it.*
4. *Repair_reports: This table contains data collected during the repair of a component. Typically data includes removal data, field observations (free text), parts replaced/repaired, and shop observations (free text)*
5. *Testcell_reports: This table contains data from the laboratory acceptance and qualification testing.* Most of the components we track undergo an acceptance test before they are shipped back to the field.

In general, the HBase has two technical components: (a) Convenient base classes that support Hadoop MapReduce jobs and functions with HBase tables; and (b) Query predicate pushes down via server side scan and gets filters that will select related data for track management systems.

As seen in Figure 1, HBase tables can work with relational databases such as SQL Server or MySQL and achieve the highest speed in processing and analyzing the big data. The following is an example of the code in Listing 1 for our HBase to get data from our SQL Server, e.g., Honeywell Predictive Trend Monitoring and Diagnostics (PTMD) database, and others.
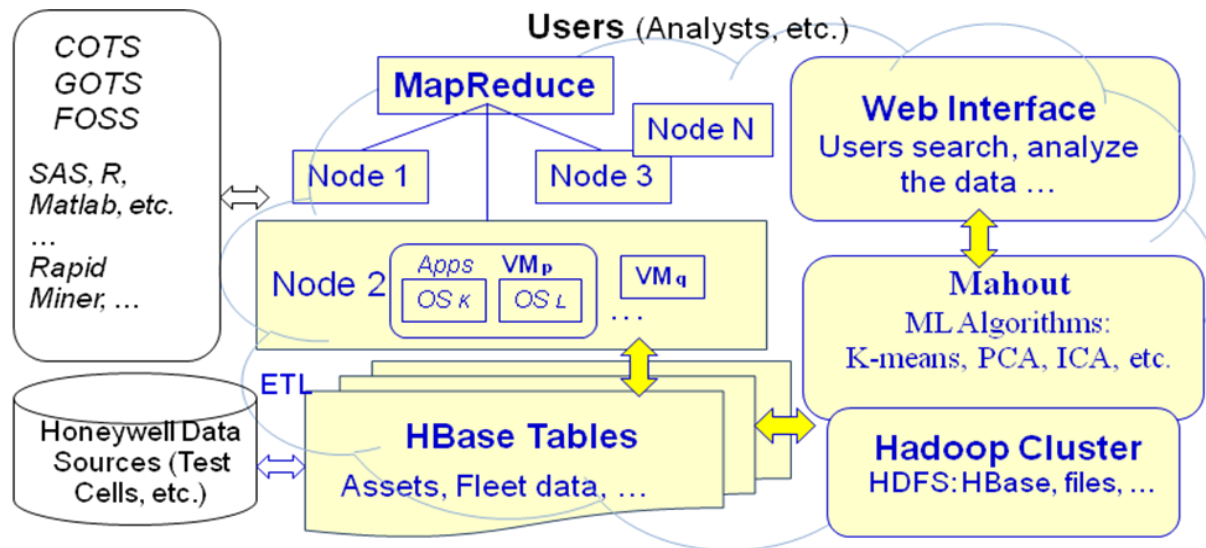
Figure 1.
Architecture and Components of our Cloud-based ML Framework.

Listing 1. A Code Segment for our HBase to get data from the SQL Server.

```
… …
HADOOP_CLASSPATH='/opt/hbase-0.92.1-security/bin/hbase classpath'
${HADOOP_HOME}/bin/hadoop jar /opt/hbase-0.92.1-security/hbase-0.92.1-security.jar importtsv -Dimporttsv.columns=HBASE_ROW_KEY, asset:model, asset:serialnumber, test:device, test:type, test:objective, test:operator,    …,
event:time, algorithm:name, algorithm:date 'Device_test'
hdfs://RTanalytic…/hadoop/scrap/DeviceSQL4HBase.Tag.txt
… …
```

Our cloud-based ML framework works with our existing SQL databases and analytic tools as seen in Figure 2.

Major existing data sources include stream datasets from test cells of aircraft engines, Auxiliary Power Units (APUs), assets (e.g. electronic parts, mechanic parts, etc.) repair shops, and aircraft fleets. We have our SQL servers, MySQL databases, and analytic tools (MatLab, proprietary toolbox, etc.). Newly developed HBase tables are populated by ETL, and the selected datasets from existing RDBMS and the HBase provide the data column families for Mahout ML tools to analyze.

3   Cloud-Based ML Framework Built Using Apache Ecosystem

Our commodity computers were virtualized by using Xen Hypervisor (www.xen.org/) as a virtualization

platform. OpenSUSE Linux operation system was installed on these virtualized computers. Our Apache Hadoop [2] software framework is installed as seen in Figure 3 on these virtual machines to support data-intensive distributed applications in aerospace industries. Our first Hadoop cluster consists of three nodes, one of which was designated as the *NameNode* and *JobTracker*. The other two machines acted as both *DataNode* and *TaskTracker*. A distributed filesystem was configured and formatted across the three nodes.

MapReduce is the core of the Hadoop technology for easily writing applications to process vast amounts of data in-parallel on Hadoop clusters. Our Hadoop cluster consists of a single master *JobTracker* and one slave *TaskTracker* per cluster-node. The master node is responsible for scheduling the jobs' component tasks on the slave nodes, monitoring them and re-executing the failed tasks. The slave computers execute the tasks as directed by the master.
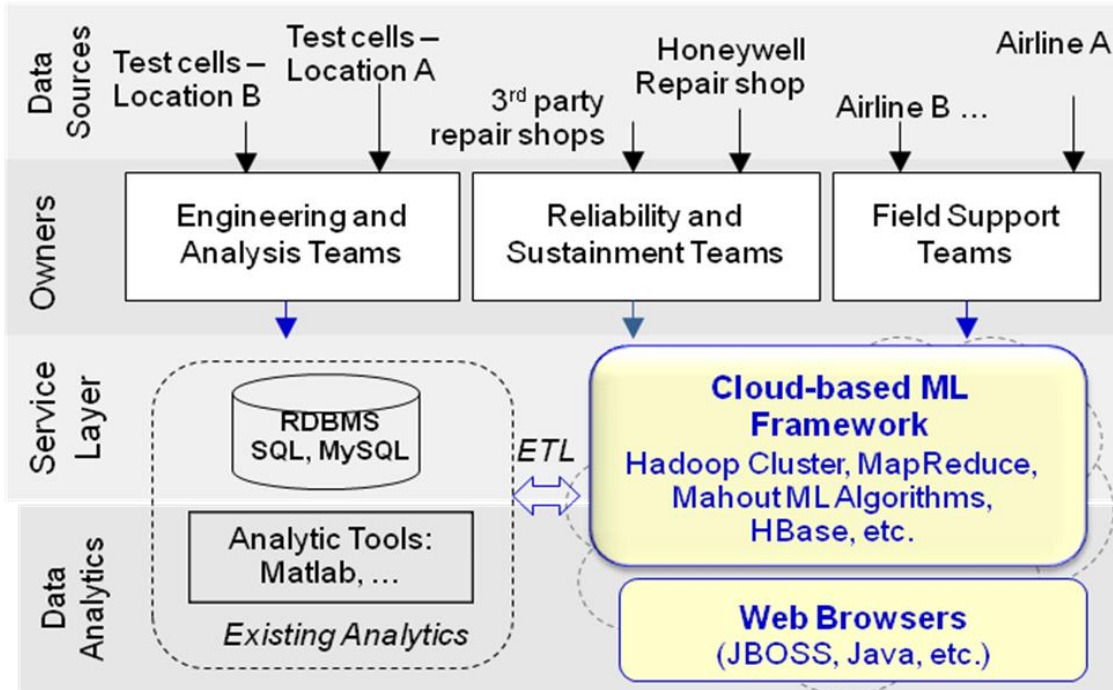
Figure 2.
The Cloud-based ML Framework working with our existing data sources and analytic tools.

Figure 3.
Screenshot of the Apache Hadoop Framework.

```
hadoop@cbmanalytic:/opt/hadoop-1.0.2> bin/hadoop fs -put input input
hadoop@cbmanalytic:/opt/hadoop-1.0.2> bin/hadoop jar hadoop-examples-1.0.2.jar g
rep input output 'dfs[a-z.]+'
13/01/29 10:52:58 INFO util.NativeCodeLoader: Loaded the native-hadoop library
13/01/29 10:52:58 WARN snappy.LoadSnappy: Snappy native library not loaded
13/01/29 10:52:58 INFO mapred.FileInputFormat: Total input paths to process : 7
13/01/29 10:52:58 INFO mapred.JobClient: Running job: job_201301150947_0002
13/01/29 10:52:59 INFO mapred.JobClient:  map 0% reduce 0%
13/01/29 10:53:36 INFO mapred.JobClient:  map 28% reduce 0%
13/01/29 10:54:15 INFO mapred.JobClient:  map 57% reduce 0%
13/01/29 10:54:32 INFO mapred.JobClient:  map 57% reduce 19%
13/01/29 10:54:47 INFO mapred.JobClient:  map 85% reduce 19%
13/01/29 10:54:53 INFO mapred.JobClient:  map 85% reduce 23%
13/01/29 10:54:56 INFO mapred.JobClient:  map 85% reduce 28%
13/01/29 10:54:59 INFO mapred.JobClient:  map 100% reduce 28%
```

Figure 4 (a).
A Screenshot of the MapReduce running on our Hadoop Cluster.

```
13/01/29 10:56:11 INFO mapred.JobClient:        Data-local map tasks=1
13/01/29 10:56:11 INFO mapred.JobClient:        SLOTS_MILLIS_REDUCES=18018
13/01/29 10:56:11 INFO mapred.JobClient:     File Input Format Counters
13/01/29 10:56:11 INFO mapred.JobClient:        Bytes Read=111
13/01/29 10:56:11 INFO mapred.JobClient:     File Output Format Counters
13/01/29 10:56:11 INFO mapred.JobClient:        Bytes Written=11
13/01/29 10:56:11 INFO mapred.JobClient:     FileSystemCounters
13/01/29 10:56:11 INFO mapred.JobClient:        FILE_BYTES_READ=25
13/01/29 10:56:11 INFO mapred.JobClient:        HDFS_BYTES_READ=243
13/01/29 10:56:11 INFO mapred.JobClient:        FILE_BYTES_WRITTEN=43521
13/01/29 10:56:11 INFO mapred.JobClient:        HDFS_BYTES_WRITTEN=11
13/01/29 10:56:11 INFO mapred.JobClient:     Map-Reduce Framework
```

Figure 4 (b).
A Screenshot of our MapReduce working with HDFS.

Specifically, our MapReduce runs on the Hadoop cluster and jobs can be submitted from the command line as shown in the examples, Figures 4 (a) and (b). The MapReduce searches for a string in an input set and writes its result to an output set. Each job can be split up into parallel tasks working on independent chunks of data across the nodes.

In addition, our HBase supports massively parallelized processing via MapReduce for using HBase as both source and sink. Hadoop Distributed File System (HDFS) is a distributed file system that is well suited for the storage of large files. The HDFS also handles failovers, and replicates blocks. Our HBase is built on top of the HDFS and provides fast record lookups and updates for large tables (see Figure 1). In addition, Eucalyptus cloud development tool [4] was used for building Amazon Web Services compatible cloud, by leveraging your existing virtualized infrastructure.

Several key Apache HBase tables are developed. The HBase provides column families of the data for Mahout ML algorithms to analyze. Sample results of our HBase tables are shown in Listings 2 and 3, respectively.

Our cloud-based ML framework is integrated and used for our CBM analytics. The workflow of this framework used in our CBM analytics applications is shown in Figure 5. The CBM workflow can be broadly described as a set of application and web-servers. The application servers are intended to serve Matlab users, while the web-service is used primarily to display summary conclusions made by Matlab-based analytics. Both these servers are implemented within the Hadoop cluster and use a Meta-data Index file to retrieve the necessary data from our Hadoop-cloud. Application servers also allow remote method invocations so that analytic calculations can be done within Matlab – making full use of its mathematical and statistical libraries.

Listing 2. Partial results of '*scan RMSS_report'*
HBase table.

Listing 3. Partial results of '*scan PTMD_report'*
HBase table.

```
… …
37 column=event:removedEngine,
timestamp=1340260441203, value=0
   ...
9 column=asset:aircraftSerial,
timestamp=1340260441203, value=F900EX
9 column=asset:engineModel,
timestamp=1340260441203, value=331-350
9 column=asset:engineSerial,
timestamp=1340260441203, value=R0122B
… …
```

```
… …
98 column=asset:airline, timestamp=1340261260821,
value=Singapore Airlines
98 column=asset:airlinecode,
timestamp=1340261260821, value=SNG
98 column=asset:mission, timestamp=1340261260821,
value=Flight
98 column=asset:model, timestamp=1340261260821,
value=331-350
98 column=asset:serial, timestamp=1340261260821,
… …
```
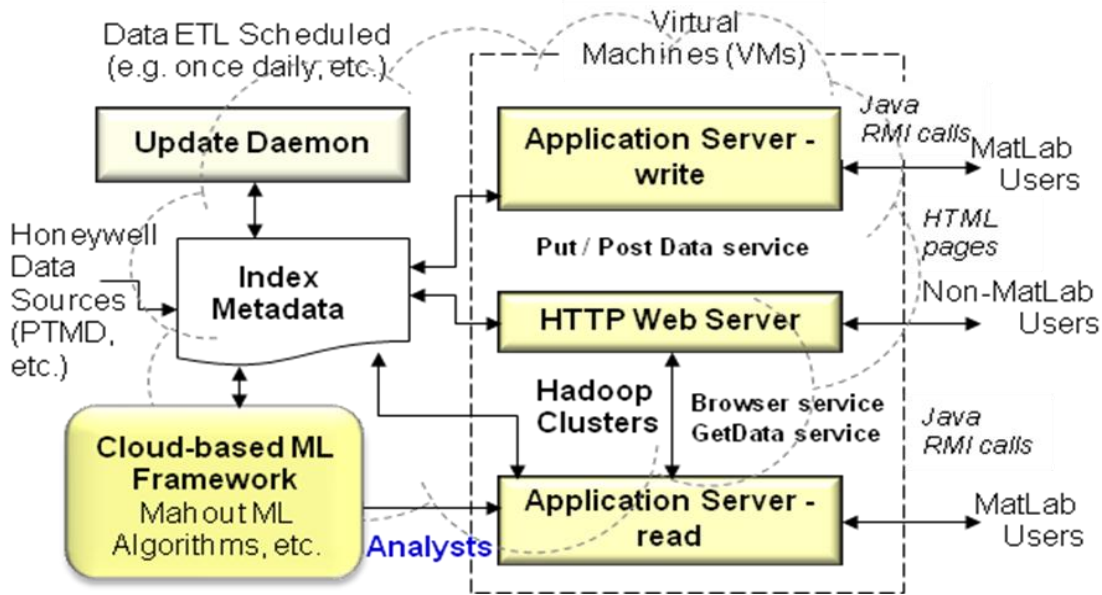

Figure 5. Workflow of the Cloud-based ML Framework for CBM Analytics.

4   Mahout Machine Learning Algorithms

Our Mahout library has been integrated in our cloud-based ML framework in Figure 1. We have integrated and tested the Mahout ML algorithms, see Figure 6 (a) and (b).

Most important applications using the Mahout ML algorithms include classification, clustering and recommendation, along with Pattern Mining, Regression, Dimension Reduction, and Evolutionary Algorithms. By analyzing over **100 millions** of data examples [5] successfully, Mahout becomes more important ML tool to use with Hadoop and MapReduce. Therefore we decided to integrate Mahout ML algorithms into our cloud-based ML framework for real world applications such as anomaly detection, fault mode prediction, etc. The Mahout machine learning algorithms are written using MapReduce paradigm:

(a) supervised learning algorithms including neural networks, support vector machines (SVMs), Naive Bayesian classifiers, decision trees, random forests, and logistic regression;

(b) unsupervised learning algorithms including k-means, hierarchical clustering, self-organizing maps, fuzzy k-means, Dirichlet, PCA, ICA, expectation-maximization, and mean-shift, etc..

One code sample of Naive Bayes in Mahout is shown in Listing 4.

We found that Mahout ML algorithms can be used for better classification by working with ZooKeeper. More specifically, when serializing stochastic gradient descent ML models for deployment as classifiers, it is usually best to only serialize the best performing sub-model from *Adaptive Logistic Regression*. The resulting serialized file will be **100 times** smaller [5] than would result from

serializing the entire ensemble of models contained in the *Adaptive Logistic Regression* object. Mahout ML algorithms are integrated into our cloud based ML framework including, Support Vector Machines, and Naive Bayes classifiers, K-Means, hierarchical clustering, and self-organizing maps, fuzzy k-Means, Dirichlet, and Mean-Shift. All these algorithms are written in Map-Reduce paradigm.

Listing 4. A code snippet of the Naive Bayes classifier in Mahout.

```
public class Starter {    public static void main( final
String[] args )
     { final BayesParameters params = new
BayesParameters(); ... ...
        params.set( "classifierType", "bayes" ); ... ...
        try { Path input = new Path( "/tmp/input" );
        TrainClassifier.trainNaiveBayes( input,
"/tmp/output", params );
        Algorithm algorithm = new BayesAlgorithm();
        Datastore datastore = new
InMemoryBayesDatastore( params );
        ClassifierContext classifier =
        New ClassifierContext( algorithm, datastore );
... ...
```

Experiments: Specifically, we tested Mahout algorithms against our data sources such as PTMD datasets, test cells, and others. One of our experiments was to test Mahout K-Means for the PTMD data. Details are briefed in the following. One of the models we want to build is a trend model that describes how the *EGT_MARGIN* changes as a function of the APU age (e.g., *flight.AHRS*, etc.), by using the PTMD data.

We believe this the *flight.TAT* is a random perturbation to this model.

Our initial steps include:

(1) Mahout Vectors are converted as a Hadoop sequence file, by using *seqdirectory* tool to convert text file into a Hadoop sequence file for Mahout to run;

(2) feature vectors are created with three dimensions,

(3) *RandomSparseAccessvector* is used for vectorizing the data, and

(4) the understandable sequence file is saved in Hadoop.

Figure 6 (a). Mahout algorithms are installed in our Framework.

Figure 6 (b). Mahout algorithms are tested in our Framework.

Technically, we
(a) got 17 data files from our HDFS;
(b) vectorized the data (for making Mahout to understand the data):
  - parsed values of *astart.EGTP*, *mesone.EGTA* and *algout.EGT_MARGIN* from the data files
  - took average of the values, since each of this parameter has an array of double values.
  - preated feature vectors with 3 dimensions. *RandomSparseAccessvector* was used for vectorizing the data
  - paved in Hadoop understandable sequence file
(c) identified initial clusters – K-Means mandates guessing initial centroids. Since the datasets have 17 files, we guessed just two centroids;
(d) ran the K-Means algorithm – using convergence delta as 0.0001, number of iterations as 10, and cluster classification threshold as 0.5.

We got the result of two classifications; 5 files fall under the classification "1" and remaining 12 in classification "0" correctly. Our future work and tasks will include the following areas:
(1) Classification: predicting a category, e.g., discrete, finite values with no ordering;
(2) Regression: predicting a numeric quantity, e.g., continuous, infinite values with ordering.

With our data sources (e.g. PTMD, etc.), we will evaluate and select the best of the ML algorithms, including Linear Regression, Logistic Regression, Linear and Logistic Regression with regularization, Neural Networks, Support Vector Machine, Naive Bayes, Nearest Neighbor, Decision Tree, Random Forest, and Gradient Boosted Trees. In addition, we have also tested well known open source ML tools such as Rapid Miner and Weka. Although Rapid Miner and Weka are not designed for cloud computing and big data analytics, we have tested them in order to evaluate and select the best possible ML algorithms for our CBM applications.

5   Conclusions

In this paper, our cloud-based ML framework is developed by using Apache Hadoop, MapReduce, HBase, and others in Apache ecosystem. Mahout Machine learning algorithms are integrated with this framework in order to analyze real word data sources from Honeywell engines, auxiliary power units (APUs), line replaceable units (LRUs), and many other products.

We found that cloud computing tools: Apache Hadoop, MapReduce, and cloud software OpenStack or Eucalyptus, can be used to develop cloud based analytics tools for various industries including aerospace and manufacturing venders, in order to provide big data analytics capabilities. They can complement traditional RDBMS and analysis technologies that are currently good for a number of commercial applications (e.g., banks, retails, etc.).

Our future work will focus on evaluating and improving more ML algorithms for specific tasks and datasets. Much more data will be bought into our cloud and analyzed by our big data analytics tools using our top machine learning algorithms more effectively and efficiently.

6   References

1. Cloud MapReduce
   http://code.google.com/p/cloudmapreduce/
2. Apache Ecosystem, http://apache.org/
3. Mahout http://mahout.apache.org/
4. Eucalyptus Cloud Software, www.eucalyptus.com/
5. Sean Owen, etc., "Mahout in Action", Manning Publications Co., 2012.

SESSION

SWARM OPTIMIZATION ALGORITHMS AND APPLICATIONS

Chair(s)

TBA

A Quantitative Approach For Ant-Based Clustering Performance

Savas Ozturk[1], Ahmet Emin Kuzucuoglu[2]
[1]TUBITAK BILGEM, Gebze, Kocaeli, Turkey
[2]Department of Computer and Control Education, Marmara University, İstanbul, Turkey

Abstract - Recent years has witnessed the frequently utilization of swarm intelligence and bio-inspired computing techniques for solving complex problems. These techniques are stochastic and it is hard to obtain quantitative performance metrics. Ant based clustering is one of the heuristic clustering techniques and completion of the clustering process can take a long time. This study basically aims to formulate algorithm's performance dependent to number of ants and data diversity.

In order to compare different configuration scheme results, a generic framework is developed for defining simulation parameters in a useful way and running visual simulations. An empirical termination criteria is defined to measure the time of complete clustering.

Experiment results has shown that number of ants employed linearly effects the clustering performance. Another important inference of this study is the correlation between object diversity and clustering performance.

1 Introduction

Clustering means bringing similar data together in a given data. Clustering achieves to ease the solution of a big problem by dividing that into little independent problems with minimized integration risk. This is a difficult problem and there are several techniques such as hierarchical clustering, fuzzy clustering and evolutionary approaches [1]. Ant-based clustering has become one of these techniques as an evolutionary way employed mostly for robotics, graph partitioning and data (specifically web data) clustering problems [2][3].

Ant-based clustering is a biologically inspired data clustering technique. Pheidole pallidula ants gather items to form heaps in order to organize a cemetery [3]. Clusters are formed as a result of very simple ant behaviors. Ants wander around and when they encounter a corpse, they decide whether to pick up it or dropping its load or go on wandering. Decision is based on according to probabilistic equations.

Ant-based clustering was first used by Deneubourg at collective robotics research and it is coined as Basic Model(BM) [4].

$$P_p = \left(\frac{k_1}{k_1 + f} \right)^2 \qquad (1)$$

$$P_d = \left(\frac{f}{k_2 + f} \right)^2 \qquad (2)$$

P_p refers to Pickup probability equation(Eq 1), k_1 is Pickup threshold parameter and its default value is 0.1. f is named as disparity function representing the amount of recently encountered similar data. f can be assumed as a short term memory holding content of recent, for example, last 100 operations handled. If $f << k_1$ then ant picks up the encountered object. As understood from this probability, pickup probability cares for current clusters by directing ants not picking an object from them. On the other hand, drop probability (P_d) attemps an ant to drop its object if there is a similar one in ant's adjacent area (Eq 2). If $k_2 << f$ then ant drops the carried object. f can be tailored to change the shape or the density of the resulting clusters [5].

Lumer and Faieta extended BM to cluster complex datasets [6]. Monmarche integrated k-means and ant-based clustering to remove small classification errors. Also he proposed an algorithm allowing several objects at the same cell and allowing one ant to carry more than one objects at a time [7]. Handl and Meyer preprocessed the data to be clustered to improve performance [8]. Handl et al also made an extensive comparison between legacy clustering methods and ant-based clustering [9].

Jafar and Sivakumar described a brief survey on ant-based clustering and presented some applications [10].

In this study, we developed a simulation framework for ant-based clustering experiments. Framework enabled us to define characteristics of ants to be employed and characteristics of data objects to be clustered. We simply applied a ant decision mechanism for ants based on BM. We had an ability to try on different pick up and drop probability parameters.

The main focus was when to stop simulation. At each simulation iteration (time step), cluster analysis helped to find whether the termination criterions were satisfied. Termination criterion was simply verifying if all the clusters were homogenous and number of clusters was same as number of different object types.

By using this obtained termination criterion, two main experiments were performed: a) relationship between time of termination and number of ants b) relationship between time of termination and number of different data

object types. After analysis of the experiments, mathematical inferences were obtained.

2 Description of The Framework

Ant-based Clustering Experimental Framework (ABCEF) achieves parametric definition of a generic ant based clustering system components and visual simulation of clustering process.

2.1 Terrain

Terrain is a simulation area that is often a square-grid, data to be clustered, and ants to be employed.
Dimensions of the terrain area is defined according to the data size. We used the empirical suggestion for the side length of the square which is noted as $\sqrt{10xn_0}$ where n_0 is number of data objects [11].

2.2 Data

Data is assumed as distributed objects on the terrain grid. Objects can be defined with unlimited characteristics at various heights. We have worked on maximum six types of objects for this study. Objects are represented by different symbols in the simulation to achieve visual perception.

2.3 Ants

Ants can be thought as simple agents which are capable of three types of actions at each time step : 1) Move 2) Pick Up 3) Drop. Movement policy of the ants are modeled according to Brownian walk [12]. At each time step, ants decide what to do by scanning neighbor cells. As seen from the Figure 1, there is no pick up and drop probability calculation.

Ants have two behavior modes: Self-organized and manual. Although complex pick up and drop probabilities can be defined by ABCEF with manual mode, for this study a rule-based mechanism is preferred as setting behavior mode to self-organized.

2.4 Decision Mechanism

Although ABCEF is designed in order to define flexible sense capability for ants, in this study ants can sense only near cells (neighborhood range is 1). This means that ant can see 8 cells at a time. The content of these cells can be "Empty", "Matching", "Other" or "Border".

According to the Figure 1, "Matching" refers to the number of cells those contain objects fully compliant to ant's pick up&drop criteria. "Other" refers to the number of objects those are not as labeled as "Matching" (Total objects = "Matching" + "Other"). "Border" refers to the number of border cells, this variable can only be 0, 3 or 5. If a cell does not contain any ant, object or border, it is labeled as "Empty".

Match function compares ant's goal(i.e. clustering red marbles) and nearby objects' attributes. One ant can only transport only one object at a time.

For better understanding of the decision mechanism, an example would be useful.

Fig. 1. Terrain and ants' neighborhood
(M: Matching, O: Other, E: Empty, B:Border, A1 : ant carries object, A2,A3: Ants carry nothing)

According to sample map at Figure 1 and decision mechanism at Figure 2, A1 is carrying an object and there is a matching object nearby. Expected behavior will be dropping object into one of empty cells. Ant2 is near the border, there is no matching cell, so its behavior will be obstacle(border) avoidance (returns back into the terrain center). Ant3 carries nothing, so it will decide picking one of the two "Matching" objects and go on.

To summarize; assumptions and constraints of this study are listed below :
- Ants are assumed to be self-organized (No specific task is given)
- Ant's visibility is limited to 1 (can scan only adjacent 8 cells)
- No obstacles in the terrain except terrain borders
- Transport capacity is 1.
- Data Object diversity is limited to 6 types.

3 Termination Policy

Ozturk and Esin previously worked on termination criterion by empirical observation of cluster formation [13]. They examined cluster formation at some phases of the simulation and defined some coefficients related with density of cluster (cluster dimensions and number of data objects in the cluster). They obtained the Eq. 3, where CC was meant as correlation coefficient, CW_i and CH_i represented each cluster width and height, O_i represented number of objects in the cluster and n represented number of clusters.

$$CC = \sum \left(\sqrt{CW_i x CH_i} / \sqrt{O_i} \right) / n \qquad (3)$$

Figure 3 shows that the graphical representation of the final clusters when simulation was terminated according to pre-defined termination criterion.
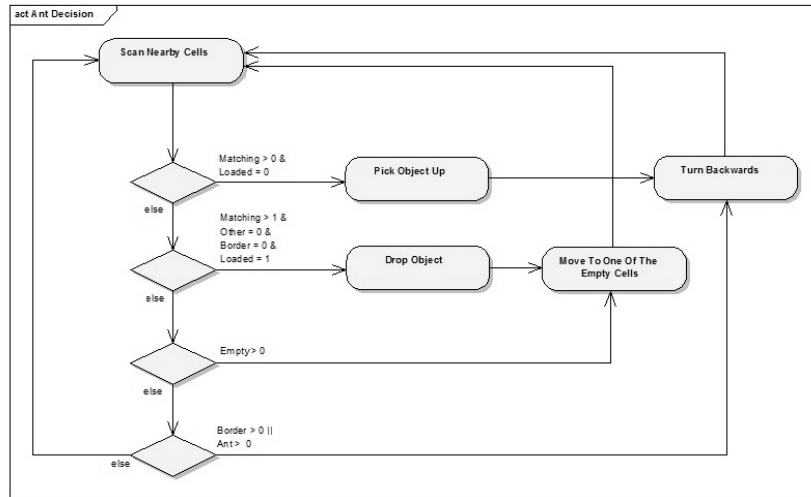
Fig. 2. Ant decision at each time step

CC=1,72
t=8314*10^3
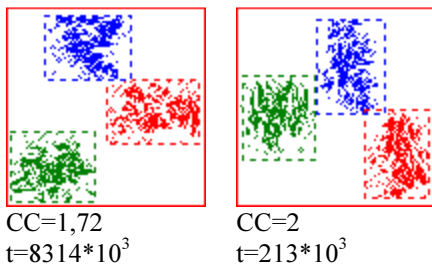
CC=2
t=213*10^3

Fig. 3. Cluster forms at termination [13]

Looking at both snapshots in Figure 3, cluster formations are approximately same and termination time at CC=2 is approximately 40 times less than termination time at CC=1,72.

It can be said that waiting too much does not help expected density. However, this approach requires initial information about data types and amount of data objects.
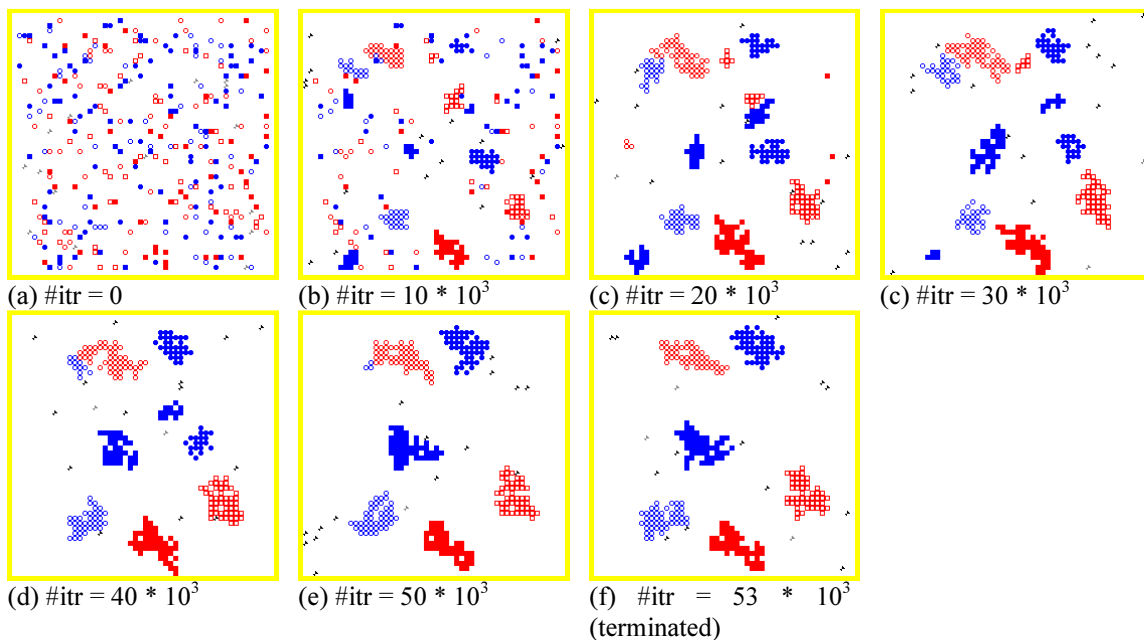

(a) #itr = 0

(b) #itr = 10 * 10^3

(c) #itr = 20 * 10^3

(c) #itr = 30 * 10^3

(d) #itr = 40 * 10^3

(e) #itr = 50 * 10^3

(f) #itr = 53 * 10^3 (terminated)

Fig. 4 Clustering process by time steps (#itr = number of iterations)

Similar termination criterion is used at this study. At each simulation step(iteration), cluster analysis is performed and it is checked that if the number of actual clusters equal to the cluster count defined by user operator. Software is capable at defining parameters for picking up and dropping, so that varied objects can be collected in a

cluster for certain purpose. But in this study, object types are limited and finally a cluster should include only one type of object. To verify this limitation, standard deviation of object attributes in the clusters are calculated whether they are equal to zero. This solid rule extends the simulation time   but we are satisfied with results, because at the end there exist clusters as same as we want. We know all the clusters' structures, they are ready to be used.

Figure 4 shows the clustering process for 6-object and 16 ants. Simulation is terminated automatically at step 53 * 10^3 by forming 6 homogenous and different types of objects. There is a little difference between step 20 * 10^3 and step 50 * 10^3 at the images. At that time, by changing termination criterion, simulation could be stopped among step 20 * 10^3 . To obey termination criterion exactly, simulation can be run up to millions times or more. Shortly, it is important to define right termination criterion for each type of data.

4 Experimental Setup

We can experiment so many cases using ABCEF. Specific settings and limitations on ABCEF peculiar to this study are mentioned at description of the framework section.

In all experiments we use totally 300 objects, so width and length of the terrain is defined as 54*54.

Terrain has a border, no other obstacles are used in the terrain.

Experiments are performed by changing number of object types and number of ants at each time (Table 2). Six object types are defined (Table 1).

Table 1. Attributes of Object Types

Object Type	Attributes	Symbol
A	Red, Full, Square	■
B	Blue, Empty, Circle	◻
C	Red, Empty, Square	◻
D	Blue, Full, Circle	●
E	Red, Empty, Circle	○
F	Blue, Full, Square	■

Table 2. Experiment Characteristics

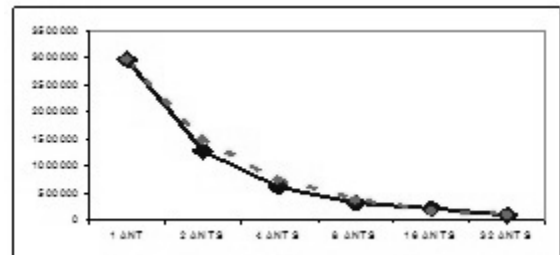| Exp No	#Objects	#Ants
1..6	300 * A	1,2,4,8,16,32
6..12	150 * (A+B)	1,2,4,8,16,32
7..18	100 * (A+B+C)	1,2,4,8,16,32
19..24	75 * (A+B+C+D)	1,2,4,8,16,32
25..30	60 * (A+B+C+D+E)	1,2,4,8,16,32
31..36	50 * (A+B+C+D+E+F)	1,2,4,8,16,32
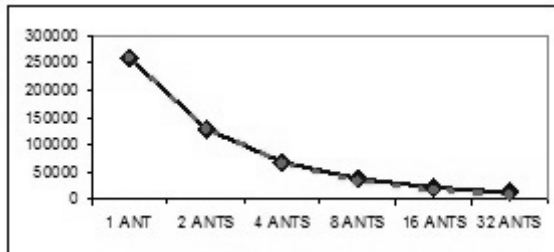
a) #Object types = 1
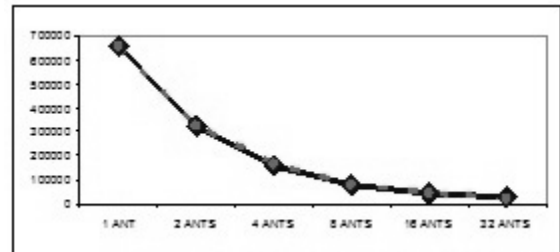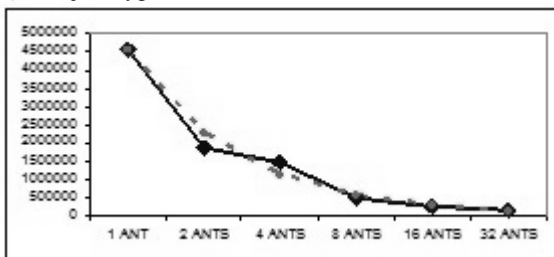
b) #Object types = 2
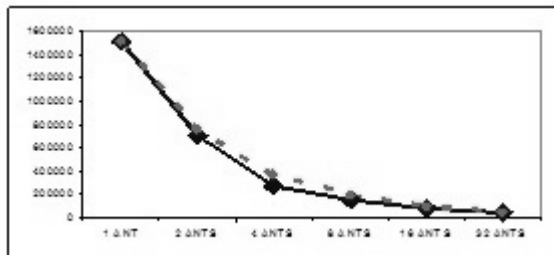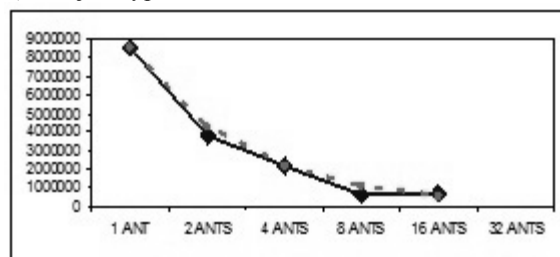
c) #Object types = 3

d) #Object types = 4

e) #Object types = 5

f) #Object types = 6

Fig. 5. Effect of the ant number to clustering time (#Iterations & # ants)

Each experiment is executed 100 times to eliminate stochastic randomness of the algorithm. Totally 3600 simulation run executed for 36 different experiments.

Termination criterion for the process is defined as obtaining the number of clusters each including the same objects. At a sample step like 10000, cluster analysis is made. If number of clusters is same as achieved final cluster number and each cluster has a standard deviation of attribute weights like zero (all objects in the cluster is same according to their attributes), termination criteria is obtained.

At the end of each experiment, the iteration number is noted if simulation is automatically stopped. Means are calculated and graphs are sketched for analysing the results.

5 Analysis of The Results

Figure 5 presents the ant number - object diversity relationship based on ant number. Object type number is assumed constant and expected values are calculated according to the first data on the graph. For example; at Figure 5a, object type number is 1. Assume that we cluster only apples. One ant brings apples into one cluster at 250000 iterations. Then expected iteration count for 2 ants for same job will be half of it.

Figure 5 helps to find out the relationship between ant count and clustering time. Martin et al. explained that single ant could achieve the task but M times slower than M ants [14]. According to them, time of completing the job is linear to number of ants.

Although we estimate a different result taking swarm intelligence characteristics and synergy concepts into account, our work has been resulted similar to their work. The results show that clustering times vary linearly according to number of ants. Figure 5. shows the linearity clearly. As seen from the figure, straight lines and dotted lines approximately bend together although object diversity is changed. This situation does not change when number of different types of objects are increased. Two ants finish the job at half time of only one ant is employed when either clustering the apples or clustering both the apples and pears respectively.

On the other hand, increasing the number of ants means more computational effort and more collisions. From the view of conventional paradigm, computation and collisions causes latency which is a bad performance factor. Results show that although there seems there is no synergy, it is an advantage that there is no latency when more ants are used. Distributed and decentralized nature of swarm intelligence eliminates such side effects.
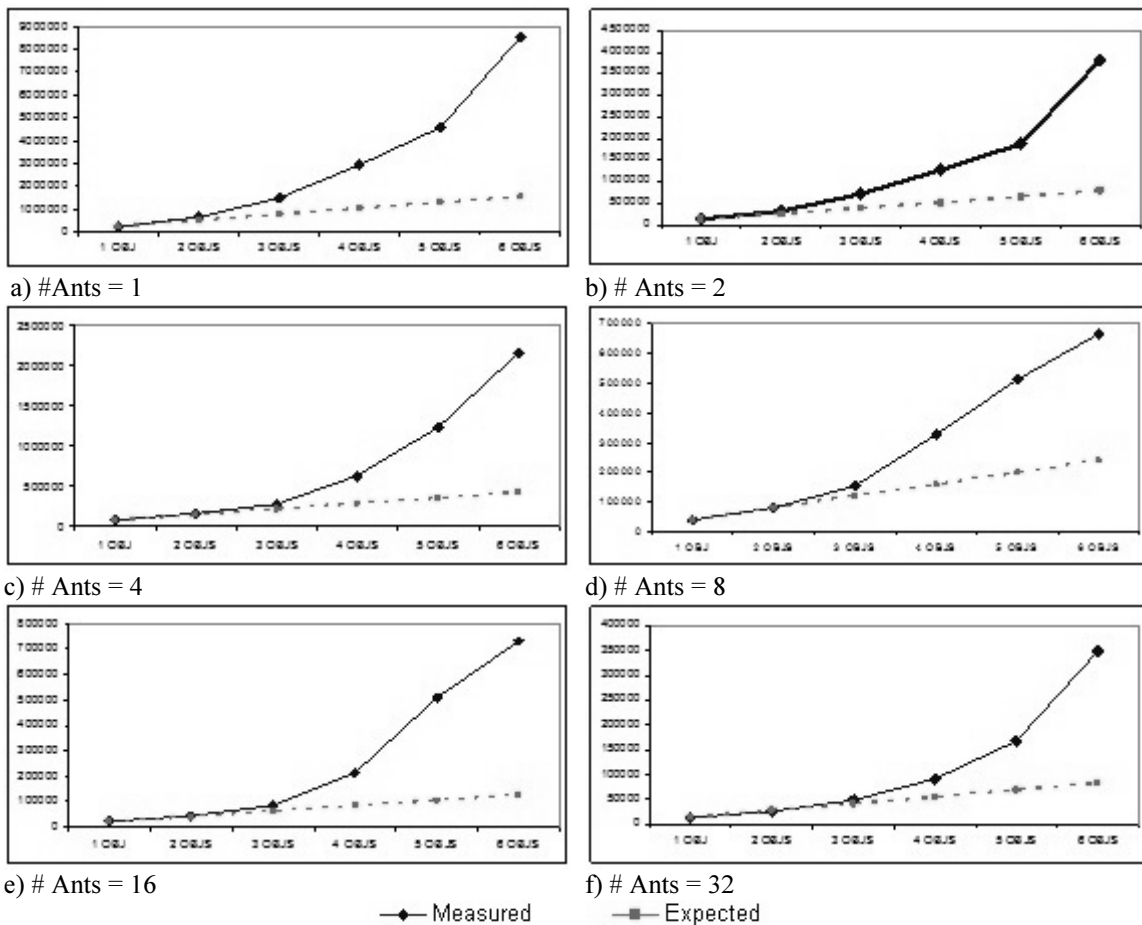

a) #Ants = 1   b) # Ants = 2

c) # Ants = 4   d) # Ants = 8

e) # Ants = 16   f) # Ants = 32

━◆━ Measured   ━■━ Expected

Fig. 6. Effect of the object type count to clustering time (# Iterations & # object types)

Finally, we can obtain the equation

$$t_n = t_1 / n \qquad (4)$$

where t_n is termination time when n ants are employed, t_1 is termination time when only one ant is employed, n is number of ants employed.

Figure 6 presents the ant number - object diversity relationship based on object diversity. Similar to Figure 5, this time ant count assumed to be constant. This time we see that one ant clusters both apples and pears in more time than when clustering only apples. As a result It is exciting that increase in the object diversity, affects the clustering in an exponential way. We can obtain equation

$$t_n = 2^{(n_o - 1)} * t_1 \qquad (5)$$

t_n is the clustering time spent for n object types, n_O is number of the object types, t_1 is the clustering time spent for only one object type. For example, if clustering time for one object is 1000, then clustering time of 6 different objects will be $2^5 * 1000$. Increasing the number of the ants does not effect this situation in any way as well.

6 Conclusion

In this paper we present an a quantitative approach to clustering performance of ants. After defining a stopping criteria for a clustering process, we count the iteration numbers until stopping criterion is met when a specific clustering problem is given.

In addition to previous research, we research object diversity at the data to be clustered. We try to obtain deterministic formulas related to ant count and object diversity.

This study concluded by producing two important relationship definitions. a) M ants accomplish one job M times faster than one ant. b) Clustering N different types of objects is 2^{N-1} times slower than one type object.

For this work, we force to form completely distinguished and homogenous clusters. We have been working for automating termination of ant-based clustering and standardizing the process to improve overall performance of the algorithm.

7 Acknowledgement

8 References

[1] Jain, A.K., Murty, M.N., Flynn, P.J., *"Data Clustering: A Review"*, ACM Computing Surveys, Vol. 31, No. 3, September 1999

[2] Berkhin, P., *"Survey of Clustering Data Mining Techniques"*, Grouping Multidimensional Data, Springer Berlin Heidelberg, pp 25-71, 2006, ISBN : 978-3-540-28349-2

[3] Bonabeau, E., Dorigo, M., Theraulaz, G., *"Swarm Intelligence : From Natural to Artificial Systems"*, Oxford University Pres, New York 1999,ISBN : 0-19-513159-2

[4] Deneuborg, J.L.,Goss, S., Franks,N., Sendova, A. , Detrain, C., Chretien, L., *"The Dynamics of Collective Sorting: Robot-Like Ants and Like-Ant Robots"*, From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behaviour, MIT Pres, 356-365

[5] Aranha, C.,C., *"A Survey on using Ant-Based techniques for clustering"*, survey paper, IBA institute's seminar, IBA Lab. - Research Laboratory of Genetic and Evolutionary Computations (GEC) of the Graduate School of Frontier Sciences, The University of Tokyo, Japan, 20 Jan. 2006

[6] Lumer, E., Faieta, B., *"Diversity and adaptation in populations of clustering ants"*, In Proceedings of the Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 3, pages 501–508. MIT Press, Cambridge, MA, 1994.

[7] Labroche, N., Monmarche, N., Venturini, G., "A New Clustering Algorithm Based On The Chemical Recognition System Of Ants", In Proc. ECAI, 2002, pp. 345-349

[8] J. Handl and B. Meyer. ,*"Improved ant-based clustering and sorting in a document retrieval interface"*, Seventh International Conference on Parallel Problem Solving from Nature, volume 2439 of LNCS, pp 913–923. Springer-Verlag, Berlin, Germany, 2002.

[9] Handl, J., Knowles, J., Dorigo,M., *"Ant-based clustering: a comparative study of its relative performance with respect to k-means, average link and 1D-som "*. Technical Report, TR/IRIDIA/2003-24

[10] Jafar, O.A.M., Sivakumar, R., "Ant-based Clustering Algorithms : A Brief Survey", International Journal of Computer Theory and Engineering, Vol. 2, No. 5, 2010, 1793-8201

[11] Handl, J., Knowles, J., Dorigo,M., *"Ant based clustering and topographic mapping"*, Artificial Life, 11(2) , 2005

[12] Gutowitz, H., *"Complexity-seeking ants"*, In Proc. Of theThird European Conference on Artificial Life, 1993

[13] Ozturk, S., Esin, E.M., *"A New Approach For Clustering and Sorting Algorithms In Swarm Intelligence"*, 20. TBD Information Congress, Istanbul, September 2003

[14] Martin, M., Chopard, B., Albuquerque, P., *"Formation of an Ant Cemetery: Swarm intelligence or statistical accident?"*, Future Generation Computer Systems, Vol. 18, no. 7, August 2002, pp. 951-959

Optimal Placement of Wind Turbines: Improved Honey Bee Mating Optimization

H. A. Shayanfar [*]

Electrical Eng. Department, Islamic Azad University,
South Tehran Branch, Tehran, Iran

O. Abedinia

Electrical Engineering Department,
Semnan University, Semnan, Iran

N. Amjady

Electrical Engineering Department,
Semnan University, Semnan, Iran

A. Tohidi

Advanced Control Systems Lab, K. N. Toosi
University of Technology, Tehran, Iran

hashayanfar@yahoo.com, oveis.abedinia@gmail.com, n_amjady@yahoo.com, akbar.tohidi64@yahoo.com

Abstract— **Small scale wind turbines can be used in indoor facilities that are mainly either stand-alone or grid-connected systems. With grid contiguous turbines, the output from the wind turbine is directly connected to the existing mains electricity supply. For this purpose Improved Honey Bee Mating Optimization (IHBMO) technique is considered as an optimization algorithm to power system. Also the wind farm in form of doubly fed induction generator penetrating is formulated into a power system. Several technical issues arise with the close spacing of multiple wind turbines in a wind farm, individually one with a severely limited spatial footprint. Wake effect is one of the most important agents under consideration that can lead to fluctuations in the output power of a wind farm and significantly decrease its energy production. Thus it is desired to determine optimal positions for installing multiple wind turbines. In this paper we minimize the fuel cost and emission of thermal units simultaneously by changing the location and varying the sizes of wind farm. Simulation results show the effectiveness of our propose algorithm.**

Keywords: Wind Farms, IHBMO, Optimal Placement.

I.   INTRODUCTION

The power system requirements for wind power mainly depend on the power system formation, the installed wind power valence, and how the wind power production varies. Wind resources vary on each time scale: second, minutes, hours, days, months and years. On all these time scales, the varying wind resources change the power system an analysis of this impact will be based on the geographical area that is of interest. The relevant wind power production to analyses is that of lager areas, like synchronously operated power system, comprising several countries or states [1].

Wind energy occupies a prominent position among renewable energy sources, and will gain in significance as governments worldwide strive to reduce the environmental footprint in the energy sector. Following this trend, optimal wind turbine placement on a selected wind farm site is of major importance, since it can lead to a remarkable increase in the produced power [2]. While dense configurations appear as an intuitive solution, the so called wake effect is a known side-effect of tight spacing shadowing. It is caused by the fact that when extracting energy from the wind, each turbine makes a cone of more turbulent and slower air behind it, and hence the wind speed encountered by downstream wind turbines decreases, leading to reduced energy yield [3].

The problem of maximizing the power produced by a wind farm by properly locating the wind turbines so as to minimize the wake effect (taking into account the physical constraints), is a typical optimization problem. Due to the highly nonlinear dependency of the produced power on the wind we resort to randomized optimization techniques that allow representing wind turbines with detailed and nonlinear models and/or include non-convex cost functions in the optimization process. In this framework, [4], [5], [6], [7] and references therein, addressed the problem in discrete space by considering a gridded version of the wind farm site and designating the corresponding cell centers as candidate wind turbine locations. A simplified wind turbine model was used and genetic algorithms were employed to solve the optimization problem. This approach was hampered though by the conservatism introduced by the discrete space optimization.

The advantage of wind power in economic is very low operation and maintenance cost. When wind farm is built to generate electricity, it can produce continually electricity into power system without many times of interruption [8-9]. Moreover, wind energy system operations do not generate air or water emissions and do not produce dangerous waste. Nor do they deplete natural resources such as coal, oil, or gas, or require significant

---

amounts of water during operation. Wind's contamination-free electricity can help reduce the environmental damage caused by power generation from all country [10]. Also, in [11-12] the author has described some problems in power system in market environment.

Accordingly, Improved Honey Bee Mating Optimization (IHBMO) technique is applied to power system to find appropriate location of wind power. The proposed technique is a new meta-heuristic algorithm which inspired by mating of the honey bee. Actually, the mentioned technique is used recently in power systems which demonstrate the good reflex of this algorithm. The effectiveness of the proposed technique is applied over IEEE 30 bus power system which considers the fuel cost and emission for minimization simultaneously. Achieved results demonstrate the efficiency of the proposed technique.

II. PROBLEM DESCRIPTION

The objective of environmental/economic power dispatch with varying location and sizing of wind farm is to minimize the economic and environmental cost function of power system while satisfying various equality and inequality constrains [3].

A. Objective Function

Minimization of generator cost
The total US$/h fuel cost $f(P_G)$ is presented as follow:

$$f(P_{Gi}, P_w) = \sum_{i=1}^{N} a_i + b_i P_{Gi} + c_i P_{Gi}^2 + d_i P_w \qquad (1)$$

Where a_i, b_i, c_i and d_i are the cost coefficients of the i_{th} generator thermal units and wind farm, and P_{Gi} and Pw are the real power output of the i_{th} generator thermal units and wind farm at bus w. N is the number of generators which can be defined as:

$$P_{Gi} = [P_{G1}, P_{G2},..., P_{GN}, P_W]^T \qquad (2)$$

Minimization of environmental emission
The collected ton/h emission $E(P_G)$ of atmospheric pollutants such as sulfur oxides SO_X and nitrogen oxides NO_X caused by fossil-fueled thermal units can be expressed as

$$e(P_{Gi}, P_w) = \sum_{i=1}^{N} \begin{aligned} &10^{-2}(\alpha_i + \beta_i P_{Gi} + \gamma_i P_{Gi}^2) \\ &+ \xi_i \exp(\lambda_i P_{Gi}) + \rho P_W \end{aligned} \qquad (3)$$

Where, α_i, γ_i, β_i, ζ_i, λ_i and ρ are coefficients of the *ith* emission characteristics of thermal units and wind farm.

B. Constraints

Generation capacity constraints
For stable operation, real power output of each generator is restricted by lower and upper limits as follows:

$$P_{Gi}^{min} \le P_{Gi} \le P_{Gi}^{max}, i = 1,..., N \qquad (4)$$
$$P_w^{min} \le P_w \le P_w^{max}, 1 \le w \le N_B$$

Where, *NB* is the number of buses.

Power balance constraints
Power balance is an equality constraint. The total power generation must cover the total demand *PD*. Hence,

$$\sum_{i=1}^{N} P_{Gi} + P_W - P_D - P_L = 0 \qquad (5)$$

Then, power loss in transmission lines can be calculated as

$$P_{loss} = \sum_{k=1}^{NL} g_k [V_i^2 + V_j^2 - 2V_i V_j \cos(\delta_i - \delta_j)] \qquad (6)$$

Where,
Vi and Vj : the voltage magnitudes at bus i and j .
δ_i and δ_j : the voltage angles at bus i and j .

Line loading constraints
For secure operation as follows:
$$S_{Li} \le S_{Li}^{max}, i \in N_L \qquad (7)$$
Where,
Sl and $L N$: The transmission line loading and the number of transmission lines.

C. Formulation of Multiobjective optimization

Accordingly, all of the objectives and constraints for the problem can be mathematically formulated as a nonlinear constraint multi-objective optimization problem as follows [13-14].
$$Minimiza \quad [f(x,u), e(x,u)] \qquad (8)$$
Subject to:
$$g(x,u) = 0 \qquad (9)$$
$$h(x,u) \le 0$$
Where,
$g(x,u)$: the equality constraints ,
$h(x,u)$: the system inequality constraints.

D. Multiobjective optimization principle

In multi-objective optimization problems, each of the answers of $x1$ and $x2$ can have one of two possibilities:
One overcomes the other or none overcomes the other. In a minimization problem, outside loss of popularity, a

solution $x1$ dominates $x2$ if the following two situations are satisfied [15]:

$$\forall i \in \{1,2,...,N_{obj}\}: f_i(x_1) \le f_i(x_2)$$

$$\exists i \in \{1,2,...,N_{obj}\}: f_i(x_1) \le f_i(x_2) \quad (10)$$

If any of the above condition is violated, the solution x_1 does not overcome the solution $x2$. If $x1$ overcomes the solution $x2$, $x1$ is named the non-dominated resolution. The solutions that are non-dominated within the entire search space are denoted as Pareto-optimal and constitute Pareto-optimal tuning. This tuning is also known as Pareto-optimal front [16-17].

III. IMPROVE HONEY BEE MATING OPTIMIZATION

The honey bee is a social insect that can survive only as a member of a community, or colony. This means that they tend to live in colonies while all the individuals are the same family. In the more highly organized societies there is a division of labor in which individuals carry out particular duties. In fact, a colony consists of a queen and several hundred drones, 30,000 to 80,000 workers and broods in the active season. Each bee undertakes sequences of actions which unfold according to genetic, ecological and social condition of the colony [18]. The queen is the most important member of the hive because she is the one that keeps the hive going by producing new queen and worker bees and any colony maybe contain one or much queen in it life's. Drones' role is to mate with the queen. In the marriage process, the queen(s) mate during their mating flights far from the nest [19]. In each mating, sperm reaches the spermatheca and accumulates there to form the genetic pool of the colony. The queen's size of spermatheca number equals to the maximum number of mating of the queen in a single mating flight is determined. When the mate is successful, the genotype of the drone is stored. In start the flight, the queen is initialized with some energy content and returns to her nest when her energy is within some threshold from zero or when her spermatheca is full. A drone's mate probabilistically is [20]:

$$P_{rob}(Q,D) = e^{-(\Delta f)/(S(t))} \quad (11)$$

Where,

Prob (Q, D) = The probability of adding the sperm of drone D to the spermatheca of queen Q

$\Delta(f)$ = The absolute difference between the fitness of D and the fitness of Q (i.e., $f(Q)$)

$S(t)$ = The speed of the queen at time t

After each transition in space, the queen's speed, and energy, decay using the following equations:

$$S(t+1) = \alpha \times S(t)(2), \quad \alpha \in [0,1] \quad (12)$$

$$E(t+1) = E(t) - \gamma$$

γ = The amount of energy reduction after each transition. The flowchart of Classic HBMO is presented in "Fig. 1", [20].

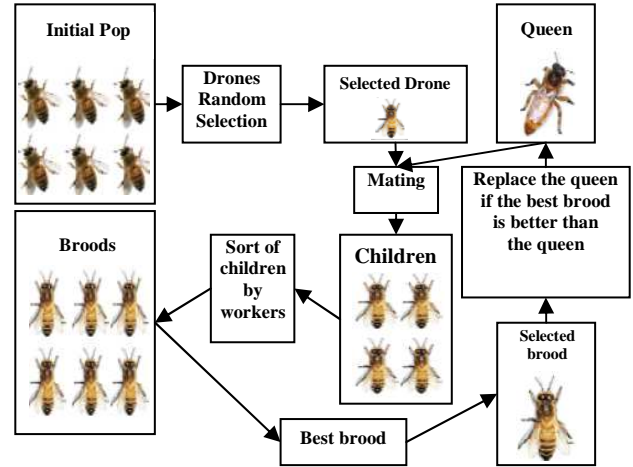Thus, HBMO algorithm may be constructed with the following five main stages [19]:

Figure 1. The Classic HBMO technique

- The algorithm starts with the mating–flight, where a queen which is the best solution, chooses drones probabilistically to form the spermatheca (list of drones). After that a drone is selected from the random list for the creation of broods.
- Existence of new broods by crossoverring the drones' genotypes with the best solutions.
- Employ the workers to conduct local search on broods.
- Modification of workers' fitness based on the achieved improvement on broods.
- Alteration of weaker queens by fitter broods.

A. Improve HBMO

In this technique the capacity of the queen's spermatheca is decreased by each mating. Hence, α is defined as:

$$\alpha(t) = \frac{M - m(t)}{M}$$
$$where \quad t = 1,...,T \quad (13)$$

Where, α is between 0, 1.

M = the capacity of spermatheca

$m(t)$ = All of the selected drones for i_{th} stage.

B. Reducing Pareto set by FCM clustering

Fuzzy c-means (FCM) is a data clustering technique in which a data set is grouped into n clusters with every data point in the dataset belonging to every cluster to a certain degree [10]. For example, a certain data point that lies close to the center of a cluster will have a high degree of belonging or membership to that cluster and another data point that lies far away from the center of a cluster will have a low degree of belonging or membership to that

cluster. It is based on minimization of the following objective function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| x_i - c_j \right\|^2, 1 \le m < \infty$$

(14)

Where m is any real number bigger than 1, $u\ ij$ is the degree of membership of $x\ i$ in the cluster j , $x\ i$ is the i_{th} of d-dimensional scaled data, $c\ j$ is the d-dimension center of the cluster, and $\| \ \|$ is any norm imparting the similarity between any measured data and the center. Fuzzy segmentation is carried out through an iterative optimization of the objective function presents above, with the upgrade of membership $u\ ij$ and the cluster centers $c\ j$ by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - c_i \right\|}{\left\| x_i - ck \right\|} \right)^{\frac{2}{m-1}}}$$

(15)

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m . x_i}{\sum_{i=1}^{N} u_{ij}^m}$$

(16)

This iteration will stop when max_{ij} $\left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < \varepsilon, \varepsilon$ is a termination criterion between 0 and 1, and the k presents the iteration step. This execution converges to a local minimum or a saddle point of J_m .

By iteratively updating the cluster centers and the membership grades for each data point, FCM iteratively shifts the cluster centers to the "right" thorugh a data set. Once the group centers have been obtained, the next solution to each center is selected, and the other solutions are eliminated. Reducing Pareto set by the FCM method is shown as Fig 2.
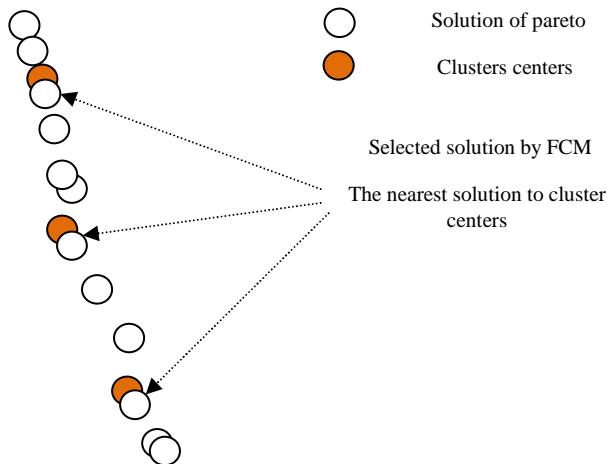

Figure 2.   The FCM method for reducing Pareto set

C.   Best Compromise Solution

After obtaining the Pareto-optimal solution, the decision maker may need to choose one best compromised solution according to the specific preference for different applications. However, because of the inaccurate nature of human arbitration, it is not conceivable to explicitly define what is really needed. Thus, fuzzy set is defined here to handle the dilemma. Here a linear membership function $i\ u$ is defined for each of the objective functions $i\ F$:

$$u_i = \begin{cases} \dfrac{F_i^{\max} - F_i}{F_i^{\max} - F_i^{\min}}, F_i^{\max} > F_i > F_i^{\min} \\ 1, F_i \le F_i^{\min} \\ 0, F_i \ge F_i^{\max} \end{cases}$$

(17)

In the above formulation, $F_i^{\max}$ and $F_i^{\min}$ is the value of the maximum and minimum in the objective functions, respectively. It is evident that this membership function indicates the degree of achievement of the objective functions. For every non-dominated solution k, the membership function can be normalized as follows:

$$u^k = \frac{\sum_{i=1}^{O} u^k i}{\sum_{k=1}^{S} \sum_{i=1}^{O} u^k i}$$

(18)

Where,

O= The number of objective functions
S= The number of non-dominated solutions.

The solution with the maximum membership uk can be seen as the best compromised solution.

D.   Implementation

The proposed technique has been developed in order to make it suitable for solving nonlinear constraints optimization problem. A computation process will check the feasibility of the candidate solution in all stage of the search process. This ensures the feasibility of the non-dominated solution.

The proposed HBMO is a recently developed powerful evolutionary algorithm, inspired from the mating process of honey bees, for solving single or multi-objective optimization problems with real-valued or discrete parameters [15]. In this paper, a new enhanced version of HBMO algorithm named Improved Honey-Bee Mating Optimization (IHBMO) is proposed for finding optimal location of wind power. The proposed IHBMO technique is tested over IEEE a 30-bus 6-generator test system considering the problem objective functions. More information of the proposed power system is presented in [9]. The values of fuel cost and emission coefficients are given in Table I.

TABLE I. GENERATING UNIT FUEL COST AND EMISSION COEFFICIENTS.

Unit	G_1	G_2	G_3	G_4	G_5	G_6
P_{min}	0.05	0.05	0.05	0.05	0.05	0.05
P_{max}	0.5	0.6	1.00	1.2	1.00	0.6
Cost						
A	10	10	20	10	20	10
B	200	150	180	100	180	150
C	100	120	40	60	40	100
Emission						
α	4.091	2.543	4.258	5.326	4.258	6.131
β	-5.55	-6.04	-5.09	-3.55	-5.09	-5.55
γ	6.490	5.638	4.586	3.380	4.586	5.151
ζ_i	2e-4	5e-4	1e-6	2e-3	1e-6	1e-5
λ_i	2.857	3.333	8.000	2.000	8.000	6.667

E. WIND FARM

Wind farm consist of a number of wind turbines connected to bus of power system topology through power transformer. Wind turbines use a doubly-fed induction generator (DFIG) consisting of a wound rotor induction generator and an AC/DC/AC IGBT-based PWM converter. For example, a wind farm is shown in Fig. 3. The stator winding is connected directly to the 60 Hz grid while the rotor is fed at variable frequency through the AC/DC/AC converter. The DFIG technology give clearance to extract maximum energy from the wind for low wind speeds by optimizing the turbine speed, whenever minimizing mechanical stresses on the turbine within gusts of wind. The optimum turbine speed begetting maximum mechanical energy for a given wind speed is proportional to the wind speed.

For optimization problem in this paper the cost and emission coefficients of wind farm are zero. Large wind turbine is selected to produce electric power up to 1.5 MW. Minimum capacity of wind farm is set as 4.5 MW or 3 wind turbines and Maximum capacity of wind farm is set as 105 MW or 70 wind turbines. These turbines speed are 12 m/s.

IV. SIMULATION RESULT

At first step the best fuel cost and emission of power system is calculated without wind farm. Accordingly the proposed IHBMO is applied to power system in situation that the wind farm is not penetrated into power system network. The achieved result is presented in Table 2.

TABLE II. RESULTS OF BEST SOLUTION WITHOUT WIND FARM

Unit (MW)	Best solution
P_{G1}	105.234
P_{G2}	72.513
P_{G3}	18.345
P_{G4}	35.355
P_{G5}	22.444
P_{G6}	29.811
Total of thermal units (MW)	283.702
Fuel Cost ($/h)	836.552
Emission (ton/hr)	0.240

For the second step, the proposed optimization problem is solved by IHBMO technique considering the wind farm in power system network. The achieved results demonstrate the optimum location of wind farm and the capacity of that. Table. 3, shows the achieved results of simulation.

Capacity of wind farm is 98.24 MW or approximately 66 wind turbines. This amount is high influence of wind farm on test system. Fig. 3 shows the best solution on tradeoff surface with wind farm in power system network. According to the presented tables it can be said that connecting the wind power could reduce the fuel cost and emission of pollution as 316.382 and 0.029, respectively.

Accordingly, the best location of wind farm is calculated to bus 21 considering the best solution of cost and emission. Fig. 5 shows the proposed power system with wind farm connection.
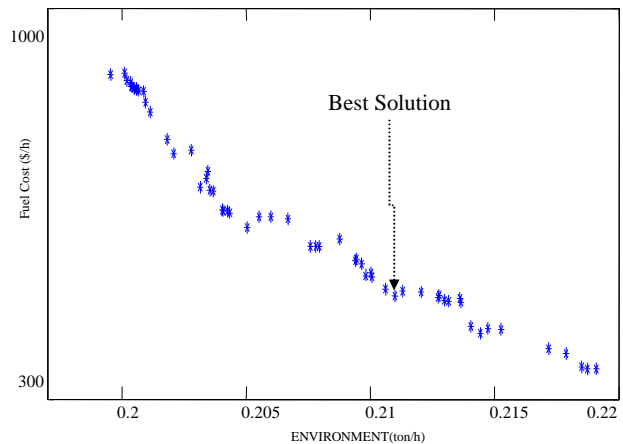

Figure 3. Best solution on tradeoff surface with wind farm in power system network

TABLE III. RESULTS OF BEST SOLUTION WITH WIND FARM

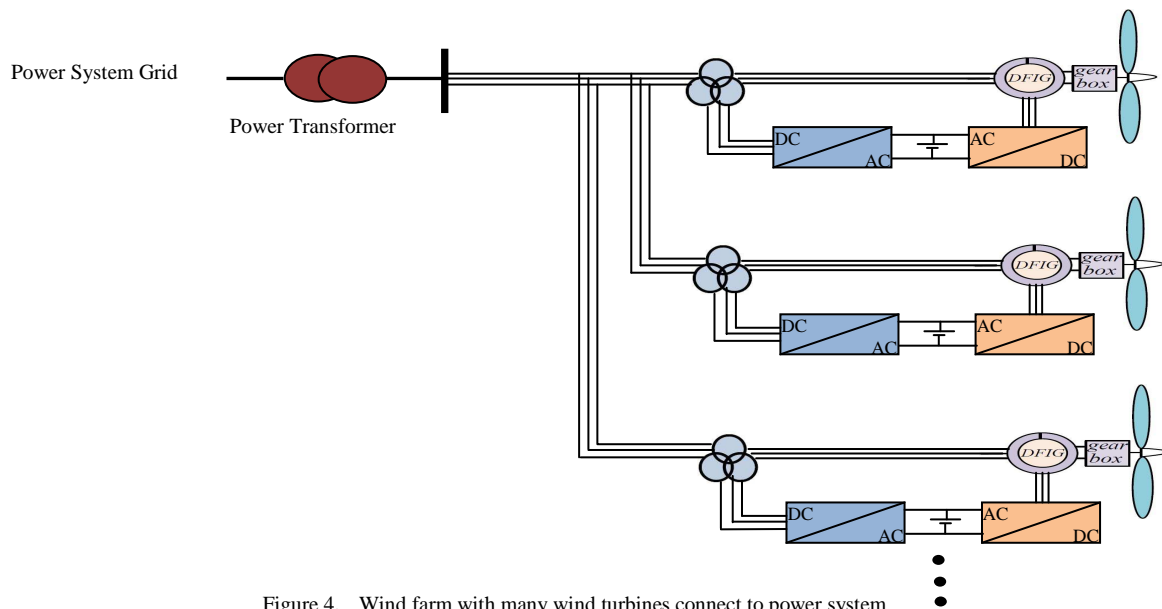| Unit (MW)	Best solution
P_{G1}	45.662
P_{G2}	29.834
P_{G3}	28.932
P_{G4}	29.147
P_{G5}	19.661
P_{G6}	25.638
Total of thermal units (MW)	178.874
Fuel Cost ($/h)	520.17
Emission (ton/hr)	0.211
Wind farm	
Location (b\Bus)	21
Size (MW)	98.24

Figure 4.   Wind farm with many wind turbines connect to power system
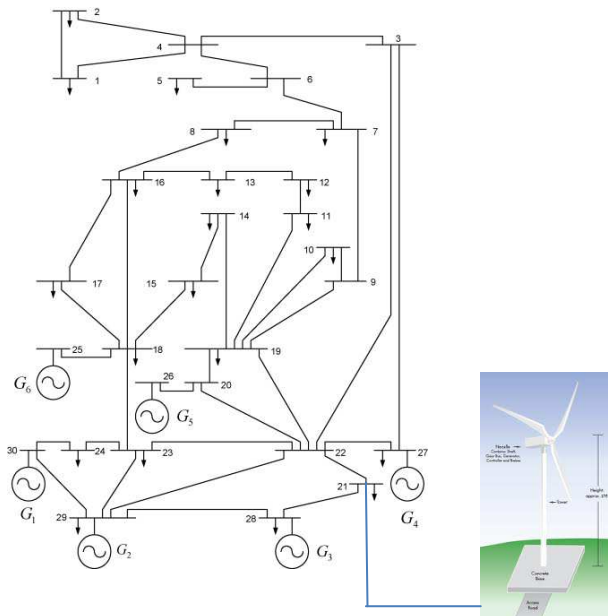

Figure 5.   Position of wind farm on power system

V. CONCLUSIONS

In this paper, Improved Honey Bee Mating Optimization (IHBMO) is applied to power system to find best location and sizing of wind farm on through minimization of generations economic and emission. Small scale wind turbines can be used in indoor, community and smaller wind energy projects and these can be either stand-alone or grid-connected systems. Wind farm is formulated in form of doubly fed induction generators to inject electric power into power system. Also, the proposed IHBMO consists of the high ability, great potential and good perspective for solving optimization problems. In this paper connecting the wind farm in optimum sizing and location could reduce the fuel cost and emission.

REFERENCES

[1]   G. Marmidis, et. al., Optimal placement of wind turbines in a wind park using Monte Carlo simulation, Renewable Energy 33, pp. 1455-1460, 2008.

[2]   E. Demeo, et. al. , Wind plant integration, Power and Energy Magazine, IEEE 3(6), pp. 38-46, 2005.

[3]   S. Grady, et. al., Placement of wind turbines using genetic algorithms, Renew Energy (Elsevier), 2005.

[4]   R. Bansal, et. al., On some of the design aspects of wind energy conversion systems, Energy Convers Manag, 43(16), pp. 2175-87, 2002.

[5]   M. Patel, Wind and Power Solar Systems, CRC Press, Boca Raton, 1999.

[6]   N. Wood, Wind flow over complex terrain: a historical perspective and the prospect for large-eddy modeling, Boundary-Layer Meteorology, 1996.

[7]   G. Mosetti, et. al., Optimization of wind turbine positioning in large wind farms by means of a genetic algorithm, J Wind Eng Ind Aerodyn, 1994.

[8]   K. Ayotte, et. al., The impact of turbulence schemes on predictions of the mixed spectral finite-difference model of flow over topography, Boundary-Layer Meteorology, 1994.

[9]   N. Jenkins, Electrical design of wind farms, Proceedings IEEE/ NTUA Athens Power Tech Conference, Athens, Greece, pp. 990-994, 1993.

[10] S. Frandsen, On the wind speed reduction in the center of large clusters of wind turbines, Proceedings of the European Wind Energy Conference, pp. 375-80, 1991.

[11] M. Esmalifalak, Z. Han, and L. Song "Effect of Stealthy Bad Data Injection on Network Congestion In Market Based Power System" IEEE Wireless Communications and Networking Conference (WCNC 2012), Paris, France, Apr. 2012.

[12] M. Esmalifalak, G. Shi, Z. Han, and L. Song "Attack Against Electricity Market–Attacker and Defender Gaming" IEEE Global Communications Conference Exhibition Industry Forum (Globecom 2012), Anaheim, USA, Dec. 2012.

[13] G.Mosetti, C.Poloni and B.Diviacco, "Optimization of wind turbine positioning in large wind farms by means of a genetic

algorithm", Journal of Wind Engineering and Industrial Aerodynamics, vol.51, no. 1, pp. 105-116, 1994.

[14] I. Katic, J. Højstrup, N.O. Jensen, "A simple model for Cluster Efficiency", Proc. European Wind Energy Conference, 1986.

[15] S. A. Grady, M. Y. Hussaini, M. M. Abdullah, "Placement of wind turbines using genetic algorithms",Renewable Energy 30(2): 259-270, 2005.

[16] J. Serrano Gonza´lez, A´ .G. Gonza´lez Rodr´ıguez, J. Castro Mora, J. Riquelme Santos and M. Burgos Pay´an,. "A New Tool for Wind Farm Optimal Design", IEEE Bucharest Power Tech Conference, 2009.

[17] G. Marmidis, S. Lazarou and E. Pyrgioti,. "Optimal placement of wind turbines in a wind park using Monte Carlo simulation", Renewable Energy, vol. 33, pp. 1455- 1460, 2008.

[18] A. Ghasemi, H.A. Shayanfar, Mohammad. S. Naderi, O. Abedinia, "Optimal placement and tuning of robust multi-machine PSS via HBMO", In: Proceedings of the international conference on artificial intelligence, Las Vegas, Nevada, pp: 1-6, USA, 2011.

[19] HA. Abbass, "Marriage in honey-bee optimization (MBO): a haplometrosis polygynous swarming approach", The Congress on Evolutionary Computation,vol. 1, pp: 207-214, 2001.

[20] O. Abedinia, Mohammad. S. Naderi, A. Ghasemi, "Robust LFC in deregulated environment: fuzzy PID using HBMO", Proceeding of the IEEE International Power & Energy Society Power Systems Conference and Exposition, Italy, Rome (EEEIC), pp: 74-77, 2011

BIOGRAPHIES

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 410 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

Akbar Tohidi was born in Ardabil, Iran, in 1985. He received the B.Sc. and M.Sc. degrees in electrical engineering K. N. Toosi University of Technology, Tehran, Iran, in 2006 and 2009, respectively. Currently, he is lecturer in Electrical Engineering Department, Parsabad Moghan Branch, Islamic Azad University, Parsabad Moghan, Iran. His current research interests include modeling and control of renewable energy applications, nonlinear controller design, adaptive and robust multivariable control systems and chaos control, sliding mode control in the field of ac drives.

Diversity Guided Particle Swarm Optimization algorithm based on Search Space Awareness Particle Dispersion (DGPSO)

Anvar Bahrampour[1], Omid Mohamad Nezami [2]
[1]Department of computer engineering, Sanandaj Branch,
Islamic Azad University, Sanandaj, Iran, Anvar.Bahrampour@gmail.com
[2] Department of computer engineering, Bijar Branch,
Islamic Azad University, Bijar, Iran, Nezami@iaubijar.ac.ir

Abstract - *Diversity control in the particle swarm optimization (PSO) algorithm is one of the important issues that influence the process of finding global optimal solution. In this study we create a historical process to find best area of the search space for population dispersion guide on PSO algorithm, and name Diversity Guided Particle Swarm Optimization algorithm (DGPSO) algorithm. Hence we propose a mechanism to guide the swarm based on diversity by using a diversifying process in order to detect suitable positions of the search space (points with fairly good fitness, and good distance from current distribution of the swarm particles) to disperse or relocating some of existing particles, hoped to increase diversity level of the swarm and escape from local optimal by detecting better area of the search space. This model uses a diversity measuring, and swarm dispersion mechanism to control the evolutionary process alternating between exploring and exploiting behavior. The numerical results show that the proposed algorithm outperforms other algorithms in most of the test cases taken in this study.*

Keywords: Evolutionary Algorithm (EA), Particle Swarm Optimization (PSO) Algorithm, Population Diversity, and Premature Convergence.

1 Introduction

Particle Swarm Optimization (PSO) applies to concept of social interaction to problem solving and it was invented by Russ Eberhart and James Kennedy in 1995 [1, 2]. Each particle represents a point of search space or a solution of problem, denoted by X_i. The PSO algorithm iteratively modifies the point and the velocity of each particle as it looks for the optimal solution based on Equation (1).

$$V_{id} = \omega * V_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id})$$
$$X_{id} = X_{id} + V_{id}$$
(1)

Where V_i in the first equation is the velocity of Particle i that represented as V_i. The first part of the Equation (1) is the inertia of the previous velocity, ω is predefined by the user, and the second part represents the cognition of the particle that shows personal thinking of the particle. The third part is the social component. In this equation c_1 and c_2 are acceleration constants. They represent the weighting of the stochastic acceleration terms that pull each particle toward personal best and global best positions. The constants r_1, r_2 are the uniformly generated random numbers in the range of (0, 1]. Although PSO is simple, but it is a powerful search technique, Many researchers have shown empirically in many studies that it works well [3].

The rate of convergence of particles in PSO is good through the fast information flowing among particles, so its diversity decreases very quickly in the successive iterations and lead to a suboptimal solution. This situation was said that an evolutionary process was trapped in a local optimal or premature convergence of the process.

The standard PSO algorithm can easily get trapped in the local optimal when solving complex multimodal problems. These weaknesses have restricted wider applications of the PSO [4, 5, and 6]. Some reasons cause to this problem, one of that is decreasing diversity of population. A number of variants of PSO algorithm have been proposed to overcome the problem of diversity loss. One of the common methods to increase the diversity is mutation. Mutation causes an improvement in exploration abilities, which can be applied to different elements of a particle swarm. The effect of mutation depends on which elements of the swarm are mutated [7]. Velocity vector mutation is equivalent to particle's position vector mutation, under the condition that the same mutation operator is considered.

In [7] a negative feedback mechanism into particle swarm optimization has proposed and developed an adaptive PSO. This mechanism takes advantage of the swarm-

diversity to control the tuning of the inertia weight (PSO-DCIW), which in turn can adjust the swarm-diversity adaptively and contribute to a successful global search. Some other methods exist that using diversity measuring and mutation in the particle's position, to promote the performance of the algorithm include Gaussian Mutation [7,9,10,11,12], Cauchy [12,13], and Chaos Mutation [14,15,16].

There are other different ways of introducing diversity and controlling the degree of diversity; Riget and Vesterstorm proposed an algorithm named ARPSO. In ARPSO if diversity is above the predefined threshold d_{high} then particles attract each other, and if it is below d_{low} then the particle repel each other until they meet the required high diversity d_{high} [17]; repulsion to keep particles away from the optimum proposed by Parsopoulos and Vrahatis [18]; LoZvbjerg and Krink made dispersion between particles that are too close to one another [19]; and Blackwell and Bentley have reduced the attraction of the swarm centre to prevent the particles clustering too tightly in one region of the search space in order to escape from local optimal [20]. J. J. Liang and P. N. Suganthan proposed a dynamic multi-swarm particle swarm optimizer (DMS-PSO)[21]. In this method whole population is divided into many small swarms, these swarms are regrouped frequently by using various regrouping schedules and information is exchanged among the swarms.

In this paper we propose a mechanism to guide the swarm based on diversity by using a diversifying process in order to detect suitable positions of the search space (points with fairly good fitness, and good distance from current distribution of the swarm particles) to disperse or relocating some of existing particles, hoped to increase diversity level of the swarm and escape from local optimal by detecting better area of the search space.

The rest of the paper is organized as follows: Section 2 we have a definition of diversity definition and measuring. The DGPSO described in section 3. Experimental results are discussed in section 4. Finally, this paper concludes in section 5.

2 Diversity deffinition and measuring

Population diversity is a way to monitor the degree of convergence or divergence in PSO search process [3]. Though, there are several measures have been used to detect diversity level of the population. Shi and Eberhart in [23], [24], [25], gave three definitions of PSO population diversity measurements that include: position diversity, velocity diversity, and cognitive diversity. Shi Cheng and Yuhui Shi gave new definition of population diversity measurement called L1 norm base on both element-wise and dimension-wise diversity [3], and they have shown that useful

information on search process of an optimization algorithm could be obtained by using dimension-wise definition in L1 norm variant, so in this paper we use L1 norm of position diversity measurement. Let m shows the number of particles and n is the number of dimensions. Dimension-wise definition in L1 norm special is as Equation(2).

$$\overline{x} = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

$$\mathrm{D}^{p} = \frac{1}{m}\sum_{i=1}^{m} |\, x_{ij} - \overline{x_{j}}\,| \qquad\qquad (2)$$

$$D^{p} = \frac{1}{n}\sum_{i=1}^{n} D_{j}^{p}$$

Where vector $\overline{x}$ is mean of particle's position on each dimension, vector D^{P} is particle's position diversity vector based on L1 norm, and D^{P} is the whole population diversity value.

In [26], some other approaches for measuring of population diversity have been introduced in evolutionary computation including: hamming distance, Euclidean distance, information entropy, etc. In this paper we use Euclidean distance in selection process of particles to disperse in dispersion mechanism. The Euclidean distance is as Equation (3) for measures the distance between two particles X and Y:

$$D(X,Y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \qquad\qquad (3)$$

3 Diversity guided particle swarm optimization

The basic idea of this research is to measure the diversity level of the swarm during the evolutionary process and once the diversity of the population drops down to the predefined threshold level d, then the system start to disperse or redistribute some of the swarm's particles by relocating them to new suitable positions which have fairly good fitness and relatively high distance from convergence position. Note, since we use previous personal best positions of dispersed particles, this is not a replacement of some particles by new generated particles. Though in this approach we relocate existing particles to new suitable positions instead of replacement them. Therefore by this process, the diversity level of the swarm will be increased up to certain degree. The evolutionary process will consistently reduce the diversity level again, and the dispersion process should be repeated once the diversity level drops down the d value. In PSO algorithm the speed of convergence is very high, so the swarm dispersion process should be repeated very often. On the other hand, repetition of this process is relatively time consuming, and in addition exploitation ability of the algorithm will be decreased by high frequency

swarm dispersion. Therefore, we introduce a new parameter T, to define the duration that the dispersion system should be passive after each redistribution process. The following two sections illustrate the process in more details. The proposed method for dispersing or relocating swarm's particles was implemented independently from the problem characteristics to improve the global convergence behavior of PSO algorithms.

3.1   Swarm dispersion process

Figure 1 shows the process of Diversity Guided Particle Swarm Optimization algorithm (DGPSO); the steps of this process are the same as the steps of the standard PSO except steps 5 and 6. In order to detect target positions of selected particles for dispersion, we use the information that could be determined from previous generations of the PSO process using previous best particle's to determine good points in the search space. With this aim, we develop an external archive to store best particles of previous generations as good positions in the search space that have been visited, hopefully, that there are better points in the regions that this stored particles located in. In step 5 we update the External Archive if necessary; there is not necessary external archive to be updated in any iteration. When dispersion system is active, last dispersion took place more than T generations ago, in step 6 the swarm diversity is measured, and if diversity is higher than the predefined threshold this step didn't do anything else, otherwise dispersion process starts to disperse some of the swarm's particles. Dispersion process will increase the swarm diversity by relocating idle particles to new potent positions. We define idle particle as a particle that there is no change in its personal best position for a long time. Process of determining target positions will describe in the following section. The final step of dispersion process is to reset velocity of dispersed particles to zero, because we want each dispersed particle search very carefully for better solutions in the vicinity of new location. In this study we found that nonzero particle's velocity probably causes to move away from new position rapidly, and lead to have unsearched area in that new region. Finally each idle particle has a period of T generation to change its personal best position, if no change took place in that duration it would be an idle particle in next idle selection process too.

3.2   Target positions of idle particles

In this section we describe a mechanism for determining target positions of the selected idle particles to disperse over the search space. In this research we established an external archive with 100 particles, and initialized it with random particles. Firstly, we gather particles with best fitness in the first generations (about 100 generations in this research) of the PSO process and replace particles in the external archive which have bad fitness. Then we should establish a replacement policy in order to gather

effective particles in external archive. These particles should have good fitness and high distance from the center of current distribution of the external archive particles to avoid the convergence of external archive. In this study after first 100 generations, we only do replacement when fitness of global best particle changed notably, and remove one of the particles with low diversity. For detect low diversity particles to remove from external archive, we use a Euclidean distance described in section 2, and measure distance of each particle from the mean of external archive particles. One of the particles with less distance should be replaced by new particle.

Step 1: Initialize PSO parameters, Diversity Threshold d, Duration T, and DispersionRate R; establish External Archive, Generate initial Swarm, and their velocities

Step 2: Evaluate the Fitness of all particles in the Swarm

Step 3: Update particles personal best position and global best position (i.e. Pi and Pg).

Step 4: Update particles velocity and position vectors according to equations (1) respectively

Step 5: Update External Archive

Step 6: if (last Dispersion took place more than T generations ago)
        Evaluate the diversity of swarm
        If (diversity < d)
            Detect suitable locations in the search space
            Select numbers idle particles for relocation
            Relocate selected particles(R%) to the new positions
            Reset velocities to zero for relocated particles
        End if
    End if

Step 7: If (Stopping criteria is reached) then go to step 8
        Else go to step 2

Step 8: Print the global best particle and the corresponding fitness function value

Figure 1. Steps of DGPSO Algorithm

To determine new good positions for relocating of idle particles from information of the external archive, we add two new particles of xMax and xMin (vector of maximum or minimum value in each dimension) to external archive, for mutation purpose. Then we create a roulette wheel that weighted each particle based on fitness and distance of external archive particles from the center of the swarm. In order to generate new target location, for each dimension value we select one particle of the external archive randomly based on Roulette, and use value stored in the same dimension of selected particle. After value of each dimension was selected, we probably have new suitable position for dispersion process, but in this time we don't use this point as good position with certainty. We collect all generated points in one matting pool, and add external archive particles as good points of the search space to the pool too. Then we apply operators such as genetic crossover and mutation to produce new points probably with good fitness and good diversity. Then we select a numbers of best points (45% of the swarm in this research) based on fitness and distance

from the center of the swarm distribution, and return to dispersion process to relocate randomly the same numbers of selected idle particles of the swarm to this new positions. This process will increase diversity of the swarm notably and help to escape from local optimal trap. Figure 2 illustrates this mechanism.

```
Ex=Ex + MaxParticle; // MaxParticle is a vector of xMax
Ex=Ex + MinParticle; // MinParticle is a vector of xMin
Roulette= Roulettewheel(Ex); //External Archive
//Create a Roulette: weight (p)=(1-fitness(p)/sum(fitness)) + distance(p)/sum(distance)
For each new Point p do
 For each dimension d do
    Particle=select (Roulette);
    Pool (p, d) =Ex (Particle, d);
 End
End
Pool=Pool+ External_Archive;
Crossover selected Points from Pool and does Mutation operator
Select good points and return to Dispersion process
```

Figure 2. Mechanism of determining target positions of idle particles.

To illustrate impact of dispersion mechanism in diversity level of the swarm in DGPSO we use a 2-D Rastringin function (f_8 in Table II) with 30 particles in 100 generations, and dispersion rate 45%. Figure 3 represents diversity curves of standard global PSO and DGPSO, and shows how diversity level of the swarm changes in DGPSO in each 15 generations.
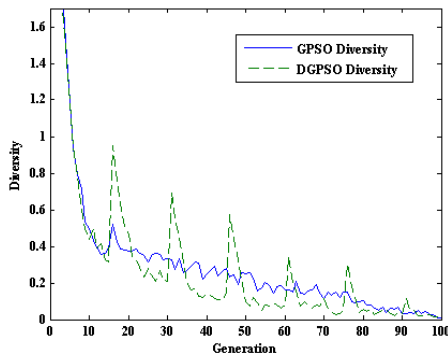

Figure 3. Diversity of the swarm of f_8 in GPSO and DGPSO Processes

4 Exprimental setting and numerical results

For comparison some variants of PSO and DGPSO algorithms, we have used a collection of 10 standard benchmark problems. Mathematical models of the problems along with the true optimum value are given in TABLE 2. In this problem set, we have a unimodal functions such as f_2, f_5, and f_7. f_5 is a noisy quadric function, where random $[0 , 1)$ is a uniformly distributed random variable in $[0 , 1)$. The others are multimodal [27]. The entire set of test problems taken for this study is scalable i.e. the problems can be tested

for any number of variables. However, for the present study we have tested the problems for dimensions 30 and 50.

In order to make a fair comparison of proposed DGPSO with some of other variants of PSO algorithm, we implement standard PSO algorithm in both global star structure and local ring structure named GPSO and LPSO, respectively. In addition to these comparisons we also implement PSO_DCIW and DMS_PSO algorithms, which proposed in [8, 21], and compared with DGPSO. We use the same initial population for all algorithms. The population size was taken as 20 while we have 30 variables (dimensions) for all the test problems, and 50 when problems should be tested with 50 variables. A linearly decreasing inertia weight is used which starts at 0.9 and ends at 0.4, with the user defined parameters c_1=2.0 and c_2=2.0. For each algorithm, the maximum number of iterations is set as 3000 iterations in the case of 30 dimensions, and 10000 for dimension 50. A new parameter T for DGPSO algorithms is set as 30 and 50 in cases of 30 and 50 dimensions respectively, with the external archive of size 100, dispersion rate R is set as 45%. In DMS-PSO we use group size 3 and regroup period 5. A total of 20 runs for each experimental setting were conducted, and the results are given in TABLE 1 and TABLE 3, in terms of mean of best fitness, standard deviation, and the improvement (%) of proposed DGPSO algorithm in comparison with original GPSO. Figure 5 through 6 show performance curves of the DGPSO in comparison with other variants of PSO for test functions f_1, f_5, and f_8 by mean fitness of best particles history found by the swarm in all runs. The numerical results show that the proposed algorithm outperforms other variants of PSO in most of the test cases taken in this study.
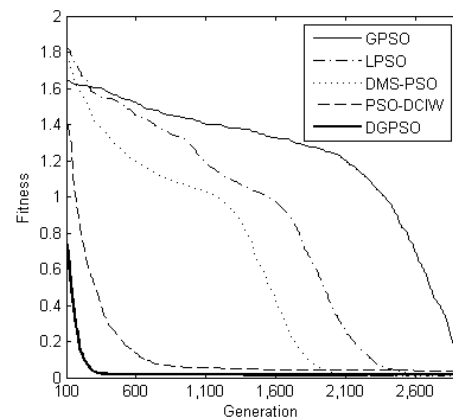

Figure 4. Performance curves of GPSO, LPSO, DMS-PSO, PSO_DCIW, and DGPSO for function f_1

5 Conclusion

Evolutionary algorithms (EAs) are best solutions for solving optimization problems; however those have different ability to investigate search space and attain optimum solution but have same behavior. One of the ideas for control

the behavior of these algorithms is a rein between exploration and exploitation, for this issue we should have good mechanism for control the diversity of population in different stages to achieve unsearched spaces. In order to control diversity to survey unsearched spaces we used search historical approach and implement that on the PSO algorithm, one of the powerful EAs. We propose a mechanism to guide the swarm based on diversity by using a diversifying process in order to detect suitable positions of the search space, this model uses a diversity measuring, and swarm dispersion mechanism to control the evolutionary process alternating between exploring and exploiting behavior and guide the algorithm to survey unsearched spaces, that was named DGPSO algorithm. The numerical results show that the proposed algorithm outperforms the basic GPSO, LPSO, DMS_PSO, and PSO_DCIW in most of the test cases with different properties, taken in this study. Of course we can use this model on the other EAs with a little modification.
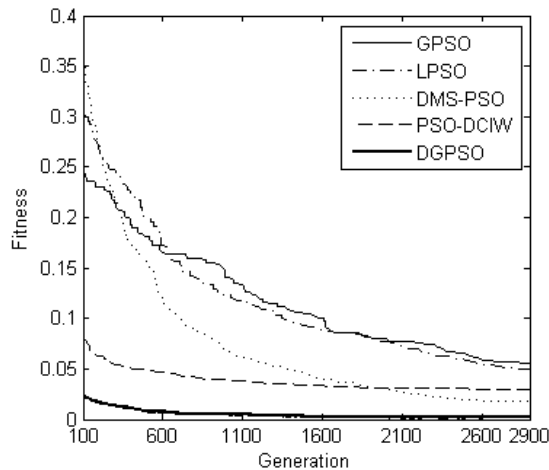

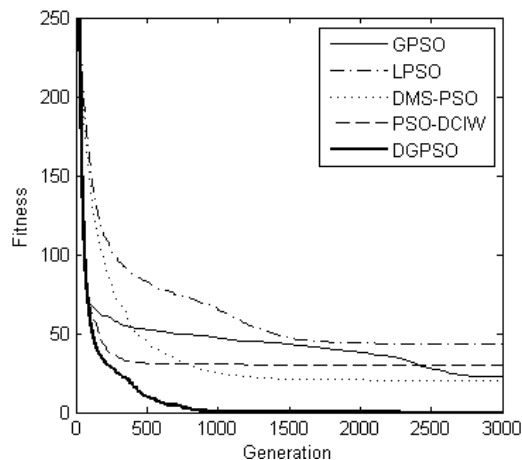Figure 5. Performance curves of GPSO, LPSO, DMS-PSO, PSO_DCIW, and DGPSO for function f_5

Figure 6. Performance curves of GPSO, LPSO, DMS-PSO, PSO_DCIW, and DGPSO for function f_8

6   References

[1] Eberhart, R. and Kennedy, J. "A new optimizer using particle swarm theory". In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1991.

[2] Kennedy, J. and Eberhart, R. "Particle swarm optimization". In *Proceedings of the IEEE International Conference on Neural Networks (ICNN),* pp. 1942-1948, 1995.

[3] Cheng, S. and Shi, Y. "Diversity control in particle swarm optimization". In *Proceedings of the IEEE Symposium on Swarm Intelligence (SIS),* 2011.

[4] Zhan, Z. H., Zhang, J., Li, Y. and Chung, H. S. "Adaptive particle swarm optimization"; *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*. vol. 39, no. 6, pp. 1362-1381, 2009.

[5] Liang, J. J., Qin, A. K., Suganthan, P. N. and Baskar, S. "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions"; *IEEE Trans. Evol. Comput.* vol 10, no 3, pp. 281-295, Jun. 2006.

[6] Li, X. D. and Engelbrecht, A. P. "Particle swarm optimization: *An introduction and its recent developments". In Proceedings of the Genetic Evol. Comput. Conf.* pp. 3391-3414, 2007.

[7] Wei, C., He, Z., Zheng, Y. and Pi, W. "Swarm directions embedded in past evolutionary programming". In *Proceedings of the IEEE Congress on Evolutionary Computation,*vol. 2, pp. 1278-1283, 2002.

[8] Jing, J., Jianchao, Z., and Chongzhao H. "Adaptive Particle Swarm Optimization with Feedback Control of Diversity". *In Proceedings of the ICIC, pp. 81 – 92*, 2006.

[9] Higashi, H. and Iba, H. "Particle swarm optimization with gaussian mutation". In *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 72-79. 2003.

[10] Secrest, B. R. and Lamont, G. B. "Visualizing particle swarm optimization–gaussian particle swarm optimization". In *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 198-204, 2003.

[11] Sriyanyong, P. "Solving economic dispatch using particle swarm optimization combined with gaussian mutation". In *Proceedings of the ECTICON*, pp. 885-888, 2008.

[12] Krohling, R.A. "Gaussian particle swarm with Jumps". In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1226-1231, 2005.

[13] Stacey, A., Jancic, M. and Grundy, I. "Particle swarm Optimization with Mutation". In *Proceedings of the IEEE Congress on Evolutionary Computation*. pp. 1425-1430, 2003.

[14] Dong, D., Jie, J., Zeng, J. and Wang, M. "Chaos-mutation-based particle swarm optimizer for dynamic environment". In *Proceedings of the 3rd Int. Conf. on Intelligent System and Knowledge Engineering*, pp. 1032-1037, 2008.

[15] Yang, M., Huang, H. and Xiao, G. "A novel dynamic particle swarm optimization algorithm based on chaotic mutation" In *Proceedings of the Workshop on Knowledge Discovery and Data Mining*. pp. 656-659, 2009.

[16] Yue-lin, G., Xiao-hui, A. and Jun-min, L. A. "particle swarm optimization algorithm with logarithm decreasing inertia weight and chaos mutation". In *Proceedings of the Conference on Computational Intelligence and Security*. pp. 61-65.

[17] Riget, J. and Vestrstorm, J. S. "*A Diversity Guided Particle Swarm Optimizer–The ARPSO".* Technical Report. Dept. of computer science, university of Aarhus, Denmark, 2002.

[18] Parsopoulos, K. E. and Vrahatis, M. N. "On the computation of all global minimizers thorough Particle Swarm Optimization"; *IEEE Transactions on Evolutionary Computation*. vol. 8, pp. 211-224, 2008.

[19] LoZvbjerg, M. and Krink, T. "Extending particle swarms with self-organized criticality". In *Proceedings of the IEEE Congress on Evolutionary Computation*. pp. 1588-1593, 2002.

[20] Blackwell, T. and Bentley, P. G. "Don't push me! collision-avoiding swarms". In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1691-1696, 2002.

[21] Liang, J. and Ponnuthurai, N. S. "Dynamic multi-swarm particle swarm optimizer with local search"; In *Proceedings of the IEEE Congr. Evol. Comput.,* pp. 522–528, 2005.

[22] Eberhart, R., Dobbins, R. and Simpson, P. "Computational intelligence PC tools". *Academic Press Professional*, 1996.

[23] Shi, Y. and Eberhart, R. "Population diversity of particle swarms". In *Proceedings of the Congress on Evolutionary Computation(CEC)*, pp. 1063-1067, 2008.

[24] Shi Y. and Eberhart, R. "Monitoring of particle swarm optimization"; *Frontiers of Computer Science*, vol. 3, no. 1, 31-37, 2009.

[25] Zhan, Z., Zhang, J. and Shi, Y. "Experimental study on PSO diversity". In *Proceedings of the Third International Workshop on Advanced Computational Intelligence*, pp. 310-317, 2010.

[26] Chang, P. C., Huang, W. H. and Ting, CH. J. "Dynamic diversity control in genetic algorithm for mining unsearched solution space in TSP problems"; *Expert systems with applications*, vol. 37, pp. 1863-1878, 2010.

[27] Yao, X., Liu, Y. and Lin, G. "Evolutionary programming made faster"; *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82-102, 1999.

TABLE 1. Comparison results of GPSO, LPSO, DMS_PSO, PSO_DCIW and DGPSO for 20 particles of 30 dimensions in 3000 iterations

Test Function	GPSO		LPSO		DMS_PSO		PSO_DCIW		DGPSO		P-value, Improvement(%) of DGPSO with GPSO	
	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	P-value (DGPSO)	Imp.(%)
f_1	0.149392	0.250608	0.010271	0.01099	**0.008371**	0.008079	0.038585	0.056493	0.016213	0.020973	**0.002639**	89.15%
f_2	0.042299	0.167624	0.000407	0.000231	2.61e-06	1.79e-06	0.002484	0.001037	**4.59e-13**	3.39e-13	**8.01e-06**	100%
f_3	40.8738	24.83168	26.59679	0.415812	26.27856	1.467588	74.70464	66.61298	**17.00601**	2.998574	**4.08e-16**	58.39%
f_4	7084.364	770.4405	5307.758	662.3028	5031.983	828.1021	6513.642	833.2667	**2538.357**	449.8559	**4.48e-16**	64.17%
f_5	0.056126	0.009203	0.049928	0.011532	0.018584	0.005755	0.029772	0.005758	**0.00162**	0.000735	**6.67e-09**	97.11%
f_6	1.159241	1.399235	0.607834	0.779671	0.00709	0.024057	0.22938	0.438579	**4.98e-12**	5.30e-12	**0.000478**	100%
f_7	0.875644	3.516198	6.75e-05	5.71e-05	4.41e-08	6.91e-08	0.000549	0.000944	**3.15e-20**	6.78e-20	0.051345	100%
f_8	22.74307	4.910085	43.22946	8.51089	20.49267	4.969791	30.06949	8.018577	**0**	0	**0**	100%
f_9	22.32039	4.791212	42.08557	8.236476	21.55	4.773557	39.25074	11.21443	**4.628837**	6.007897	**0.00271**	79.26%
f_{10}	2.255667	0.018616	2.129218	0.143674	1.411243	0.173116	3.14568	1.67769	**0.484916**	0.431148	**7.44e-05**	78.50%

TABLE 2. Benchmark Functions used in our experimental study

Function	Function Definition	Range	Optimum
Griewank Function	$f_1(x) = \frac{1}{4000}\sum_{i=0}^{n-1} x_i^2 + \sum_{i=0}^{n-1}\cos(\frac{x_i}{\sqrt{i+1}}) + 1$	[-600,600]	0
Schwefel function 2.22	$f_2(x) = \sum_{i=0}^{n-1}|x_i| + \prod_{i=0}^{n-1}|x_i|$	[-10,10]	0
Rosenbrock Function	$f_3(x) = \sum_{i=0}^{n-1} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$	[-30,30]	0
Schwefel Function	$f_4(x) = 418.9829n - \sum_{i=1}^{n} x_i\sin(\sqrt{|x_i|})$	[-500,500]	0
Noisy Function	$f_5(x) = (\sum_{i=0}^{n-1}(i+1)x_i^4) + rand[0,1]$	[-1.28,1.28]	0
Ackley Function	$f_6(x) = 20 + e - 20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i))$	[-32,32]	0
Schwefel function 1.2	$f_7(x) = \sum_{i=0}^{n-1}(\sum_{j=0}^{i} x_i)^2$	[-100,100]	0
Rastringin Function	$f_8(x) = \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i) + 10)$	[-5.12,5.12]	0
Noncaontinuous Rastringin Function	$f_9(x) = \sum_{i=1}^{n}[y_i^2 - 10\cos(2\pi y_i) + 10]$ $y_i = \begin{cases} x_i & |x_i| < \frac{1}{2} \\ \frac{round(2x_i)}{2} & |x_i| \geq \frac{1}{2} \end{cases}$	[-5.12,5.12]	0
Shaffer's Function	$f_{10}(x) = (\sum_{i=1}^{n}x_i^2)^{1/4}[\sin^2(50(\sum_{i=1}^{n}x_i^2)^{1/10}) + 1.0]$	[-32.767,32.767]	0

TABLE 3. Comparison results of GPSO, LPSO, DMS_PSO, PSO_DCIW and DGPSO for 50 particles of 50 dimensions in 10000 iterations

Test Function	GPSO		LPSO		DMS_PSO		PSO_DCIW		DGPSO		P-value, Improvement(%) of DGPSO with GPSO	
	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	Mean of Best Fitness	Std Dev	P-value (DGPSO)	Imp.(%)
f_1	0.171684	0.343501	**5.07e-06**	1.18e-05	0.010824	0.011451	0.026426	0.037654	0.000863	0.002685	0.167035	99.5%
f_2	0.022719	0.083965	5.92e-07	3.42e-07	7.42e-11	4.84e-11	0.000197	0.000102	**1.07e-20**	1.75e-20	**1.31e-02**	100%
f_3	87.84807	53.03229	45.0903	0.392029	44.42762	1.419095	94.80813	40.12472	**12.97845**	6.417684	**2.59e-08**	85.23%
f_4	12610.62	1265.407	8427.603	587.5423	8654.23	737.3883	10831.61	1234.096	**4780.807**	438.2983	**1.98e-21**	62.09%
f_5	0.144703	0.016286	0.094603	0.019196	0.028865	0.006778	0.071137	0.013351	**0.000553**	0.000203	**1.90e-10**	99.62%
f_6	4.910699	4.209546	0.092147	0.309743	1.45e-06	2.16e-06	0.029624	0.117226	**2.56e-14**	3.15e-15	**5.14e-19**	100%
f_7	11.31171	20.10893	1.03e-08	5.07e-09	7.54e-15	1.32e-14	9.15e-07	1.26e-06	**1.47e-33**	2.14e-33	**0.006363**	100%
f_8	28.93487	7.574004	62.79516	11.89742	21.39162	4.705682	36.22136	11.4106	**0**	0	**0**	100%
f_9	30.34149	6.003359	65.30312	9.873662	24.65	5.751659	43.6	11.91814	**1.4**	3.574766	0.095997	95.39%
f_{10}	2.787375	0.019692	2.576146	0.117281	1.898071	0.180956	2.169315	0.750711	**0.256608**	0.33295	**2.70e-03**	90.79%

Geometric Optimisation using Karva for Graphical Processing Units

A.V. Husselmann and K.A. Hawick

Computer Science, Massey University, North Shore 102-904, Auckland, New Zealand
email: k.a.hawick@massey.ac.nz
Tel: +64 9 414 0800   Fax: +64 9 441 8181

Abstract— *Population-based evolutionary algorithms continue to play an important role in artifically intelligent systems, but can not always easily use parallel computation. We have combined a geometric (any-space) particle swarm optimisation algorithm with use of Ferreira's Karva language of gene expression programming to produce a hybrid that can accelerate the genetic operators and which can rapidly attain a good solution. We show how Graphical Processing Units (GPUs) can be exploited for this. While the geometric particle swarm optimiser is not markedly faster that genetic programming, we show it does attain good solutions faster, which is important for the problems discussed when the fitness function is inordinately expensive to compute.*

Keywords: CUDA; geometric; genetic programming; gpu; parallel; particle swarm

1. Introduction

The advent of combinatorial optimisers saw the birth of genetic programming (GP) [17], [30], which is the term now widely representative of all algorithms intended to optimise in the search space of programs. John Koza first invented GP in 1995 alongside the pioneering work of Kennedy and Eberhart towards the Particle Swarm Optimiser (PSO) [15], [3], [16]. Combinatorial and parametric optimisers have largely evolved in tandem over time, and a great deal of research effort has been expended in improving them. These have resulted in a great many varieties of parametric optimisers, including the Cooperative PSO of van den Bergh [38], space exploration advancements [13] and notably, data-parallel optimisers [9]. Combinatorial optimisation in the space of programs have also gained Cartesian GP [23], Gene Expression Programming (GEP) [4] and a handful of others. The intentions behind these algorithms are usually to improve upon one or more aspects of the class of optimiser in question. GP has been applied to a variety of problem domains including intrusion detection [2], robotics [21], [20], geography [22], algorithm discovery [37], image enhancement [31], as well as data mining [40] and cooperative mult-agent systems [29].

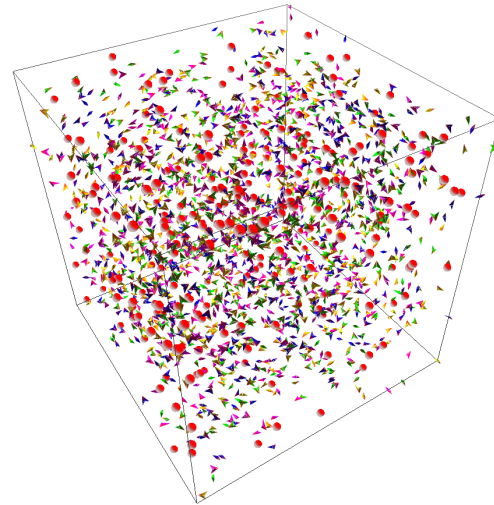Genetic Programming (GP) is an optimiser intended to successively evolve a generation of candidate solution pro-

Fig. 1: A 3D version of the classic Genetic Programming test environment named the Santa Fe Ant Trail. This is a visualisation of 2048 "ants" in their competitive coevolution in gathering the red food items.

grams to solve a particular problem. John Koza's work toward GP [17] was arguably inspired by the pioneering work of John Holland [6] earlier in the 20th century on the canonical genetic algorithm. Most variants of the original algorithm today make use of various representations, operators and other techniques such as elitism and alternative selection mechanisms. GP maintains a population of candidate solutions and computes a new population of individuals after some genetic operators have been executed based on relative fitness evaluations of these individuals. These operators are most commonly crossover, mutation and selection. These are analogous to biological processes, which was the primary source of inspiration behind *them*.

One modification worthy of note is Ferreira's Gene Expression Programming (GEP) [4] algorithm. A common problem with GP-based algorithms is in choosing a suitable representation for candidate solutions. The original made use primarily of abstract syntax trees (ASTs) in pointer-tree storage and directly for evolution. Linear GP algorithms [1] store their programs as linear strings of instructions, executed one after the other. This unfortunately suffers from

the Halting Problem [28]. GEP represents individuals as a string of symbols (codons) but this string is known as a genotype, since it must be interpreted to obtain a tree-based phenotype. This has a number of advantages, which are discussed in more detail in Section 2.

Data-parallelism on commodity-priced hardware such as Graphical Processing Units (GPUs) have gained much interest in the past few years for their relatively inexpensive and formidable processing power [19]. The concept of executing Evolutionary Algorithms (EAs) on GPUs is not new however; many modified algorithms (particularly EAs) have been proposed for a great variety of problems, but most of these focus on parallelising the fitness evaluation process [18]. This is commonly the most computationally expensive and time consuming aspect of EAs.

Geometrically unified EAs have recently become a field of interest due to Moraglio and colleagues [24], [27]. It has resulted in the generalisation of a handful of EAs to arbitrary search spaces. The only caveat of these *geometric* algorithms is that the user must be able to provide geometric operators suitable for the search space under consideration. Geometric Differential Evolution [26], [27] and Geometric Particle Swarm Optimisation [25] have already been proposed with varying success. While generalisation of these excellent parametric optimisers incite keen interest, their efficacy in comparison to Genetic Programming and other combinatorial optimisers have not been accurately determined yet [33].

In this article we seek to explore the dynamics of the Geometric Particle Swarm Optimiser (GPSO) using a modified representation named *k*-expressions (short for *karva*-expressions) from Ferreira's Gene Expression Programming (GEP) algorithm with modified genetic operators to suit this representation. We then accelerate this algorithm using NVidia CUDA-enabled GPUs. NVidia's Compute Unified Device Architecture (CUDA) allows access to an effective and highly efficient means of utilising data-parallelism to many algorithms including agent-based modelling [8], [14] and other situated agent parallelism [11] problems as well as optimisation [13].

Section 2 contains more details on the GEP *k*-expression representation, including the relevant genetic operators introduced by Ferreira, as well as a brief introduction on CUDA and GPU-based EAs. We also present a brief summary of the GPSO. Following this, in Section 3 we describe our algorithm and the combination of GEP and GPSO as well as the modified genetic operators we use. We then present some performance and convergence results in Section 5 and compare our results with a Genetic Programming algorithm. Finally, we discuss our results and conclude in Sections 5 and 6 respectively.

2. Background

As mentioned before, the canonical GP maintains a population of candidate solutions. In this work, we have elected to
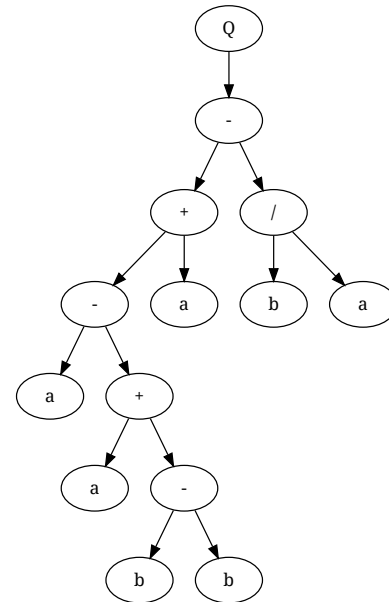

Fig. 2: The phenotypic AST built from the genotype represented by the *karva*-expression Q-+/-abaa+a-bbacda.

use the *k*-expression representation for the elegant simplicity it affords to genetic operators, and its inherent support for introns, or non-coding sections. The linear nature of this representation is also very desirable for parallelisation, even taking into account that an interpreter would still be necessary.

K-expressions are arranged in a string of fixed length, divided into a head section and a tail section. Function symbols can be of any arity, but can only appear in the head section of the expression. This serves the purpose of ensuring that the genotype is always interpreted into a syntactically correct phenotype tree. Terminal symbols may appear in both the head or tail sections. An example *k*-expression is shown below. When interpreted, this expression results in the tree shown in Fig. 2. The tree is built by reading the expression from left-to-right and filling in the arguments of each codon, level by level.

```
012345678901234567
Q-+/-abaa+a-bbacda
```

It is noteworthy that the terminal symbol d does not appear in the phenotype. This is the mechanism by which introns are supported. Once all arguments have been supplied in the tree, the rest of the expression is simply ignored, but not discarded [4]. A mutation in the tail section which swaps a terminal for a non-terminal could easily reactivate these ignored codons.

We now provide a brief overview of the original PSO by Kennedy and Eberhart. The PSO is characteristically known for maintaining a constantly updated global best solution, as well as a constantly updated personal best

solution for each candidate. The behaviour of the algorithm in geometrically moving a candidate through space, biased toward the personal best or global best is reminiscent of crossover and mutation behaviour, as it also contains a stochastic component. This random aspect of the algorithm has also been subject to improvements, as it is the main space exploration mechanism; without which, there could only be genetic drift [12]. Eqns. 1 and 2 show the recurrence relations which underlie the inertial PSO [35].

The stochastic component is introduced by the ephemeral random constants r_p and r_g (typically uniform random deviates in (0,1)), and user-defined constants ω, ϕ_p and ϕ_g. ω is known as the inertial constant, whereas the latter two constants determine a constructive bias towards either the personal best $\mathbf{p_i}$ or global best $\mathbf{g}$. Together, these equations attempt to create convergence, and diffusive space exploration similar to that of Brownian motion [5]. Recent work has led this toward more computationally expensive Lévy flights [34], [12], [7] for their improved convergence qualities.

$$\mathbf{v_{i+1}} = \omega\mathbf{v_i} + \phi_p r_p(\mathbf{p_i} - \mathbf{x_i}) + \phi_g r_g(\mathbf{g} - \mathbf{x_i}) \qquad (1)$$

$$\mathbf{x_{i+1}} = \mathbf{x_i} + \mathbf{v_i} \qquad (2)$$

It is prudent to also discuss the Geometric PSO (GPSO) and the implications that it puts forward. In the simplest implementation, this algorithm requires that a new mutation operator be defined, and a new crossover operator be defined for multiple parents; so as to mimic geometric movement toward a personal best solution and/or the global best solution. Considering that the original algorithm shown in Eqns. 1 and 2 requires geometric movement and specifically bias, toward either the global or personal best, it becomes clear that this new crossover operator must be able to bias towards one parent or the other, hence being weighted in some fashion.

Crossover and mutation in the context of linear genetic programs was investigated by Togelius and others [36] and several possible operators were proposed. The authors concede that significant research still remains in finding the most appropriate operators, but some effective ones presented include weighted subtree swap, weighted homologous crossover and weighted one-point crossover. Homologous crossover ensures that the common region between two candidates are kept intact [32]. Togelius and colleagues reports that common regions can sometimes be too small for this operator to be constructive [36]. The other two operators are more self-explanatory.

Finally, we now provide a brief summary of GPU-based simulation, especially using NVidia's (Compute Unified Device Architecture) CUDA platform [19]. The CUDA architecture arose from a potent arrangement of MIMD and SIMD computing, initially intended for processing large quantities of pixel data in parallel. Many researchers have spent years

using "ping-pong" buffering with pixel and fragment shaders to modify textures in order to accomplish General-purpose GPU simulation (GPGPU). CUDA makes this process far more accessible.

Essentially, the GPU is divided into several Streaming Multi-processors (SMs) with a certain number of "CUDA cores" which process work units known as "blocks". Each of these blocks would be sized by the user, up to a maximum (at the time of writing) of 1024 threads. When an SM executes a block, the threads are divided into groups of 16, which are named "warps". These warps are the smallest unit of execution in CUDA. They are subsequently executed in a SIMD fashion on the CUDA cores in each SM. This arrangement is sometimes known as Single-instruction Multiple-thread (SIMT).

GPUs generally have some idiosyncratic behaviour regarding memory access coalescence, scoping and penalties, among other aspects which usually require special consideration. The CUDA memory hierarchy provides a range of memories with varying access times and scope restrictions, but we omit an extensive discussion on this for brevity. The process of executing simulations while taking advantage of the vast computing power of CUDA usually involves copying data to the GPU global memory from the host, then performing the GPU-specific code ("CUDA kernel"), and finally copying the modified data back. There are more efficient ways of utilising CUDA-enabled GPUs, such as host page-locked memory, which remove the need for expensive memory copies between host and device.

GPU-specific code is written by using special syntax which NVidia released as an addition to the C language. This syntax is parsed and compiled by the NVidia compiler, and then the rest of the code is passed to the system C/C++ compiler for normal processing. The result is a C or C++ program with additional non-C syntax which is effectively removed by the `nvcc` compiler, and the rest is compiled as a regular program.

3. Method

Our method for combining CUDA, GEP and GPSO is summarised in Alg. 1. We modify the weighted crossover of The Particle Swarm Programming algorithm of Togelius and colleagues [36] to operate on *k*-expressions by following the multi-parent crossover scheme the authors proposed. This is taken from [36] and shown in Eq. 3.

$$\Delta GX((a, w_a), (b, w_b), (c, w_c)) =$$
$$GX((GX((a, \frac{w_a}{w_a + w_b}), (b, \frac{w_b}{w_a + w_b})), w_a + w_b), (c, w_c)) \qquad (3)$$

GX is the crossover operator, and ΔGX is the multi-parent crossover operator. It is assumed that w_a, w_b and w_c are all positive and sum to 1. Essentially this equation defines the weighted, multi-parent crossover as two crossovers, the

---

Algorithm 1 The parallel implementation of the GPSO on GEP *k*-expressions .

---
   allocate and initialise enough space for n candidate programs

   allocate space for random deviates
   while termination criteria not met **do**
      call CURAND to fill the random number array with uniform deviates in the range [0,1)

      copy candidates and candidate bests to device
      CUDA: compute_argument_maps()
      CUDA: interpret/execute programs
      CUDA: update food locations/fitness
      copy back to host

      if end-of-generation then **then**
         CUDA: update candidate bests
         CUDA: recombine and mutate programs
         replace old programs with new ones
      end if
      visualise the result
   end while

---

first being between a and b, where weights are re-normalised to sum to 1, and the second is a crossover with c. Togelius, De Nardi and Moraglio provide more details on this using convex set theory [36]. See also [27], [24], [25], [36] for more details on the rationale and mathematical aspect of this procedure.

As can be seen in Alg. 1 we parallelise the majority of computations. In order to interpret the programs, we also need to compute an "argument map" so as to allow the interpreter to determine which arguments belong to which functions in program strings.

Now we have determined the mechanism by which we will ensure that crossover still maintains its geometric properties as much as possible. What remains to be determined is precisely how this will be done on the linear *k*-expressions . Ferreira [4] defines one-point crossover as choosing a crossover site or "pivot", and then exchanging symbols about this point to obtain two new candidates. In order to ensure that this crossover is geometric in the sense that we can compute a multi-parent one-point crossover and still be able to bias the result towards one parent candidate or the other, we must ensure that it is *weighted*.

Our method for accomplishing this recombination is by using the ω, ϕ_g and ϕ_p parameters as the weights (w_a, w_b and w_c) in Eq. 3. We further define the candidate a as the current candidate under consideration, b as the corresponding personal best of a, and c as the global best candidate discovered so far. The fitness values of these are not used in the crossover process. Notice also that unlike GP, we do

not require selection, other than simply $P(crossover)$, a probability defined by the user, as in GP. GEP crossover defines a "donor" and a "recipient" tree, which are chosen randomly also.

Mutation is simple in comparison. Traditionally, mutation is derived from initialisation methods such as [39]:

1) Grow method
2) Full method
3) Ramped half-and-half

Typically, mutation is simply a replacement of a subtree by regrowing it using one of these methods. Point mutation is not the only space exploration operator, but it is the one given the most consideration [39] since the work of Koza [17].

Point mutation is simple for *k*-expressions , apart from the only restriction being that a tail-section symbol may not be swapped for a function symbol. This ensures that the result of the mutation operator is always a valid candidate. It is worth noting that a symbol in the head section may be changed into any other function symbol, regardless of arity [4]. The size of the head and tail are left as a configuration parameter, but can be computed to ensure the head is maximised. From Ferreira's work [4] the head and tail section sizes must satisfy the equation shown in Eq. 4:

$$t = h(n-1) + 1 \tag{4}$$

The symbols in the equation represent the tail length t, head length h and the maximum arity possible in the function set n.

Having described our algorithm we now turn our attention towards a suitable test platform. The Santa Fe Ant Trail is a classic problem used for evaluating genetic programming-based algorithms. Essentially the problem demands an appropriate combination of two function symbols (`IfFoodAhead` arity 2, `ProgN2` arity 2) and three terminal symbols (`Move`, `Right` and `Left`) for pursuing food particles in a spatial 2D environment. The `IfFoodAhead` function executes its first argument if there is food straight ahead of the particle, and the second argument if not. The `ProgN2` function simply executes both of its arguments in order. In our case, we have elected to use a 3D version of the problem. Apart from having `Left` and `Right` terminals, we introduce `Up` and `Down` terminals.

To the best of our knowledge, there has been no previous effort to parallelise the GEP (or indeed a variant of this) algorithm on GPUs. However, to assist in comparison, we have compared results from this algorithm against an implementation of Genetic Programming (GP) with tournament selection and *karva*-expressions as program representation.

Our analysis of this algorithm involves two aspects. Firstly, its ability to converge upon a good solution (preferably the global optimum), and secondly its ability to utilise the parallel architecture of a GPU; hence its *wall-clock* efficiency.
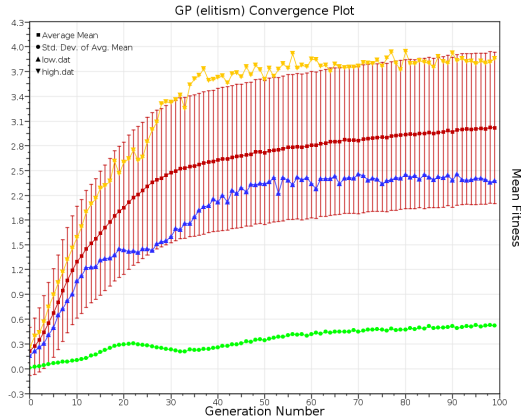
Fig. 3: Convergence results for the GP, with elitism. The graph shows the average mean value of each generation, from 100 independent runs. The error bars represent the average standard deviation of the 100 runs in each generation.
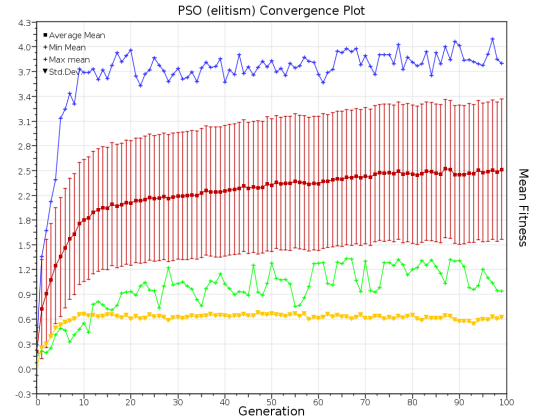

Fig. 4: Convergence results for the GPSO with elitism. Each data point has been averaged 100 times in independent runs, and the error bars represent the average standard deviation of each generation.

In our experiments we compare against the aforementioned GP implementation with k-expressions in terms of convergence and speed. The parameters we used for the GP were: P(Crossover) = 0.8, P(Mutate) = 0.1. We use the same crossover and mutation rates for the modified GPSO, and for the PSO-specific settings, we used: $\omega = 0.1$, $\phi_p = 0.6$, $\phi_g = 0.3$. As for the simulation itself, we restrict angular velocities to 0.1 units, and initial velocities are initialised to between -0.16 and 0.16. In order to use a higher mutation rate, Togelius et al recommend using Elitism, whereby the best candidate is replicated verbatim into the new population following the genetic operators. This is a common technique used in EAs to bias the population in a particular direction. We make use of elitism in both the GP and the GPSO.

4. Algorithm Convergence Results

Fig. 5 shows the convergence results for the modified GPSO and the GP. Each data point in all the plots shown have been averaged 100 times in independent runs. It is therefore conclusive that the PSO is indeed more able to find a good solution faster, but if computing fitness for more than about 23 frames is viable, then the GP is more appropriate.

Figs. 3 and 4 show the convergence results for the GP and GPSO respectively. We experimented with elitism, where the best individual is copied verbatim into the next generation, Fig. 5 shows conclusively that elitism allows the algorithms to perform better, albeit marginally.

Each data point of the Average Mean has been represents an average of 100 means from the same generation number. From these plots, it is clear that the GPSO has more spread per generation than the GP, which is not very desirable. The minimum and maximum values are also shown.

Finally, Fig. 6 shows the average compute time, by generation, for both the GP and the GPSO. The fitness evaluation consisted of computing 300 frames of the candidate

programs and gathering fitness results from this. Therefore, each data point represents the average frame compute time across each of the 300 frames, and then averaged 100 times by independent runs. The generation compute times are also shown, although they are somewhat hidden. While the first observation seems that the GPSO is faster than the GP, this is somewhat misleading. Essentially, the plots in Fig. 6 would be completely linear, if all the terminal and function symbols were of the same complexity.

The average new-generation population compute time for the GP was $420\mu sec$, and for the GPSO it was $440\mu sec$. Even though this is not comparable to the fitness evaluation ($340,000\mu sec$), it was still worth the effort, as this must happen in serial following the fitness evaluation phase. The function `IfFoodAhead` has a rough complexity of $\mathcal{O}(f)$, where f is the number of food particles, which would approach $\mathcal{O}(fN)$, should all candidates have one of these symbols in its program. Of course, the worst case here is that every candidate consists only of these functions and enough terminals to satisfy the k-expression's head and tail sections. Hypothetically, given a maximum expression length $l = 8$, and a head length $h = 3$ (hence a tail length of 5), then the maximum number of `IfFoodAhead` functions would be 3. Extrapolating from this, assume all N particles were formed like this, then evaluation would be of complexity $\mathcal{O}(3fN)$, which could very well exceed $\mathcal{O}(N^2)$.

Therefore, following from this argument, we could perhaps conjecture that at generation 20, the GP increased its use of the `IfFoodAhead` function, while the PSO had reached a steady equilibrium of a certain number of these functions. This would seem to agree with our suspicion that the GP is in fact better in preserving population diversity. In Section 5 we explore this in more depth.

Our attempts to improve the GPSO beyond the results we see here was met with disappointment. Our parameter tuning
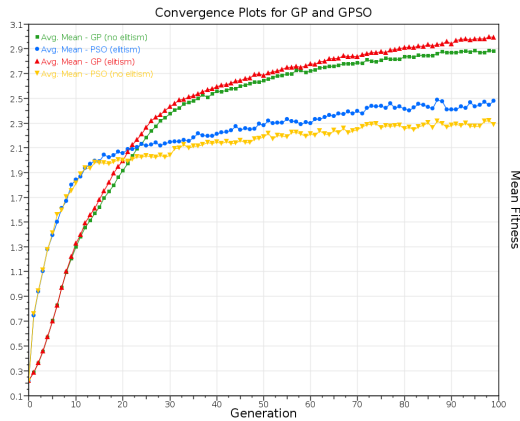
Fig. 5: Convergence results for the GP and GPSO, as well as the use of elitism for both. Each data point has been averaged across 100 independent runs to obtain meaningful statistical data.

Fig. 6: Compute times for fitness evaluation, and generation compute time for GPSO and GP.

effort for phi_g, phi_p and ω included normalised combination of respective scores of particles and also normalised weighted scores, but the best parameters were simply $phi_g = 0.3, phi_p = 0.6, \omega = 0.1$.

Fine-tuning crossover and mutation probabilities had varying effects on convergence. Removing the crossover phase with the global best solution reduced mean scores to 0.2, and similar results were obtained from removing the crossover with the personal best. Randomising slightly the crossover point with hand-tuned parameters to aid in diversity did not improve scores at all.

Our results indicate that, at the very least, that the GPSO operating over *k*-expressions is appropriate for when the fitness evaluation is extremely computationally expensive. Given enough time and compute power, however, the GP operating on *k*-expressions is more suited to the problem.

5. Discussion

Evolutionary Algorithms such as the GPSO and the GP we have compared above frequently involve a very manual parameter-tuning effort in order to ensure an unbiased comparison. We describe a meta-optimiser (also based on the PSO) in [10]. We found that the PSO was suitable as a "super-optimiser" or "meta-optimiser" for fitness evaluations which are of relatively low compute expense. In this case, fitness evaluation was clearly far more expensive, and for a "meta-optimiser" to be successful in obtaining good parameters, it would need to be an optimiser which requires very few frames for a good solution. In this case, we believe that our meta-optimiser could potentially take months to obtain a result comparable to hand-tuning. Meta-optimisers, in general, are notoriously expensive to run.

From our experiments it is not immediately clear why the GPSO is not as effective as the GP over more than approximately 23 generations. We believe that this may be
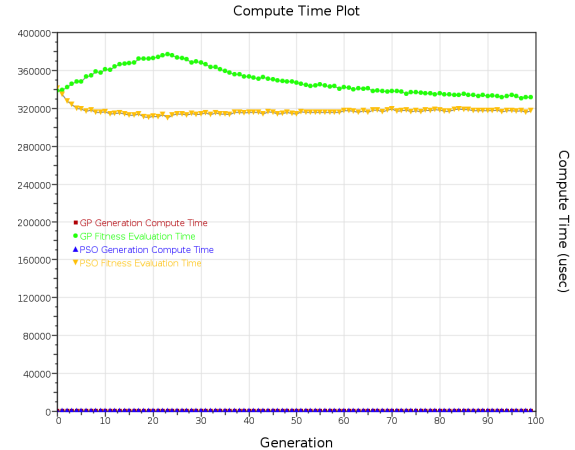
due to how diversity in population is managed between the GPSO and GP. Consider the following as the rationale for this: Fig. 5 shows a clear change in average mean fitness from approximately generation 10 for the GPSO. Whereas, for the GP, a very slight decrease in average mean fitness is shown. This is reminiscent of local minima in parametric optimisers. This may also be indicative of the inability of the GPSO to use extensive diversity to its advantage in escaping and reaching the global optimum. Consider also the artifact shown in Fig. 3 in the standard deviation at generation number 20. The same position in Fig. 4 is fully linear.

As for the performance data we present, it would be unwise to favour the GPSO from the observation in Fig. 6 that the frame compute time is lower. The slightly higher compute time does, after all, translate into a higher success rate as shown in Figs. 3 and 5.

6. Conclusions

We have presented a modified Geometric Particle Swarm Optimiser (GPSO) searching through the space of Ferreira's *k*-expressions. We have also compared this against a Geometric version of the canonical Genetic Programming method for evolutionary optimisation in the space of *k*-expressions. Our results show that the GPSO is not clearly superior over the GP, however, it is able to attain an acceptable solution faster, more consistently. This could be a desirable attribute, especially when the fitness function is inordinately expensive to compute.

We have also shown that Geometric algorithms such as the GPSO can be parallelised effectively in both the fitness evaluation phase, and the genetic operator phase (mutation, recombination). CUDA is particularly effective in this case, as evolutionary algorithms lend themselves well to data-parallelism.

There is scope for other Evolutionary Algorithms with geometric modification and parallelisation to be investigated using similar GPU/Hybrid techniques to those we have presented.

References

[1] Markus Brameier. *On Linear Genetic Programming*. PhD thesis, University of Dortmund, 2004.

[2] Mark Crosbie and Eugene H. Spafford. Applying genetic programming to intrusion detection. Technical report, Department of Computer Sciences, Purdue University, West Lafayette, 1995. AAAI Technical Report FS-95-01.

[3] R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proc. Sixth Int. Symp. on Micromachine and Human Science*, pages 39–43, Nagoya, Japan, 1995.

[4] Cândida Ferreira. Gene expression programming: A new adaptive algorithm for solving problems. *Complex Systems*, 13(2):87–129, 2001.

[5] G.E.Uhlenbeck and L.S.Ornstein. On the theory of the Brownian motion. *Phys.Rev.*, 36:823–841, Sep 1930.

[6] J. H. Holland. *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press, 1975.

[7] Gang Huang, Yuanming Long, and Jinhang Li. Levy flight search patterns in particle swarm optimization. In *Seventh International Conference on Natural Computation*, 2011.

[8] A. V. Husselmann and K. A. Hawick. Spatial agent-based modelling and simulations - a review. Technical Report CSTN-153, Computer Science, Massey University, Albany, North Shore,102-904, Auckland, New Zealand, October 2011. In Proc. IIMS Postgraduate Student Conference, October 2011.

[9] A. V. Husselmann and K. A. Hawick. Levy flights for particle swarm optimisation algorithms on graphical processing units. Technical report, Computer Science, Massey University, 2012. Submitted to J. Parallel and Cloud Computing.

[10] A. V. Husselmann and K. A. Hawick. Particle swarm-based meta-optimising on graphical processing units. Technical report, Computer Science, Massey University, 2012. Submitted to AsiaMIC, Phuket, Thailand 2013.

[11] A. V. Husselmann and K. A. Hawick. Spatial data structures, sorting and gpu parallelism for situated-agent simulation and visualisation. In *Proc. Int. Conf. on Modelling, Simulation and Visualization Methods (MSV'12)*, pages 14–20, Las Vegas, USA, 16-19 July 2012. CSREA.

[12] A. V. Husselmann and K. A. Hawick. Random flights for particle swarm optimisers. In *Proc. 12th IASTED Int. Conf. on Artificial Intelligence and Applications*, Innsbruck, Austria, 11-13 February 2013. IASTED.

[13] Alwyn V. Husselmann and K. A. Hawick. Parallel parametric optimisation with firefly algorithms on graphical processing units. In *Proc. Int. Conf. on Genetic and Evolutionary Methods (GEM'12)*, number CSTN-141, pages 77–83, Las Vegas, USA, 16-19 July 2012. CSREA.

[14] A.V. Husselmann and K.A. Hawick. Simulating species interactions and complex emergence in multiple flocks of boids with gpus. In T. Gonzalez, editor, *Proc. IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2011)*, pages 100–107, Dallas, USA, 14-16 Dec 2011. IASTED.

[15] Kennedy and Eberhart. Particle swarm optimization. *Proc. IEEE Int. Conf. on Neural Networks*, 4:1942–1948, 1995.

[16] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In *Proc. IEEE Int. Conf. on Neural Networks*, pages 1942–1948, Piscataway, NJ. USA, 1995.

[17] John R. Koza. Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, 4(2):87–112, June 1994.

[18] W. B. Langdon. A many-threaded cuda interpreter for genetic programming. In Ana Isabel Esparcia-Alcazar, Aniko Ekart, Sara Silva, Stephen Dignum, and A. Sima Uyar, editors, *Proceedings of the 13th European Conference on Genetic Programming, EuroGP*, pages 146–158. Springer, April 2010.

[19] Arno Leist, Daniel P. Playne, and K. A. Hawick. Exploiting Graphical Processing Units for Data-Parallel Scientific Applications. *Concurrency and Computation: Practice and Experience*, 21(18):2400–2437, 25 December 2009. CSTN-065.

[20] Sean Luke. Genetic programming produced competitive soccer softbot teams for robocup97. In J. R. Koza, W. Banzhaf, K. Chellapilla, D. Kumar, K. Deb, M. Dorigo, D.B. Fogel, M.H. Garzon, D.E. Goldberg, H. Iba, and R. Riolo, editors, *Genetic Programming 1998: Proceedings of the 3rd annual conference*, pages 214–222. Morgan Kaufmann, San Mateo, California, 1998.

[21] Sean Luke, Charles Hohn, Jonathan Farris, Gary Jackson, and James Hendler. Co-evolving soccer softbot team coordination with genetic programming. *Robocup-97: Robot soccer world cup I*, 1:398–411, 1998.

[22] Steven M. Manson. Agent-based modeling and genetic programming for modeling land change in the southern yucatán peninsular region of mexico. *Agriculture Ecosystems & Environment*, 111:47–62, 2005.

[23] Julian F. Miller and Stephen L. Smith. Redundancy and computational efficiency in cartesian genetic programming. *IEEE Transactions on Evolutionary Computation*, 10(2):167–174, 2006.

[24] A. Moraglio. *Towards a Geometric Unification of Evolutionary Algorithms*. PhD thesis, Computer Science and Electronic Engineering, University of Essex, 2007.

[25] A. Moraglio, C. Di Chio, and R. Poli. Geometric particle swarm optimization. In M. Eber et al, editor, *Proceedings of the European conference on genetic programming (EuroGP)*, volume 4445 of *Lecture notes in computer science*, pages 125–136, Berlin, 2007. Springer.

[26] A. Moraglio and S. Silva. Geometric differential evolution on the space of genetic programs. *Genetic Programming*, 6021:171–183, 2010.

[27] A. Moraglio and J. Togelius. Geometric differential evolution. In *Proceedings of GECCO-2009*, pages 1705–1712. ACM Press, 2009.

[28] Michael O'Neill, Leonardo Vanneschi, Steven Gustafson, and Wolfgang Banzhaf. Open issues in genetic programming. *Genetic Programming and Evolvable Machines*, 11:339–363, 2010.

[29] Panait and Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11:387–434, 2005.

[30] R. Poli, W.B. Langdon, and N.F. McPhee. *A field guide to genetic programming*. lulu.com, 2008.

[31] Riccardo Poli and Stefano Cagnoni. Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement. In *Genetic Programming 1997: Proceedings of the 2nd Annual Conference*, pages 269–277. Morgan Kaufmann, 1997.

[32] Riccardo Poli and Nicholas F. McPhee. Exact schema theory for gp and variable-length gas with homologous crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, 2001.

[33] Riccardo Poli, Leonardo Vanneschi, William B. Langdon, and Nicholas Freitag McPhee. Theoretical results in genetic programming: the next ten years? *Genetic Programming and Evolvable Machines*, 11:285–320, 2010.

[34] Toby J. Richer. The levy particle swarm. In *IEEE Congress on Evolutionary Computation*, 2006.

[35] Yuhui Shi and Russel Eberhart. A modified particle swarm optimizer. In *Evolutionary Computation Proceedings*, 1998.

[36] Julian Togelius, Renzo De Nardi, and Alberto Moraglio. Geometric pso + gp = particle swarm programming. In *2008 IEEE Congress on Evolutionary computation (CEC 2008)*, 2008.

[37] Sjors van Berkel. Automatic discovery of distributed algorithms for large-scale systems. Master's thesis, Delft University of Technology, 2012.

[38] Frans van den Bergh. *An Analysis of Particle Swarm Optimizers*. PhD thesis, University of Pretoria, 2001.

[39] Matthew Walker. Introduction to genetic programming. Downloaded from http://www.cs.montana.edu/ bwall/cs580/introduction_to_gp.pdf February 5, 2013.

[40] Chi Zhou, Weimin Xiao, Thomas M. Tirpak, and Peter C. Nelson. Evolving accurate and compact classification rules with gene expression programming. *IEEE Transactions on Evolutionary Computation*, 7:519–531, December 2003.

HybridMiner: Effective Rule Mining based on a Hybrid Dynamic Swarm Method

Veenu Mangat[1] and Dr. Renu Vig[2]
[1] Assistant Professor, UIET, Panjab University, Chandigarh, India
[2] Director-Professor, UIET, Panjab University, Chandigarh, India

Abstract - *An enormous amount of information pertaining to medicine is produced and recorded continuously. Discovering useful knowledge and assisting decision makers in the diagnosis and treatment of diseases from this vast data has become imperative. The preferred data mining functionality is association rule mining as rules are simple to understand and infer. These rules can be used to classify patients based on recorded characteristics. This paper applies latest improvements in swarm intelligence to devise a novel strategy for rule mining that exhibits high predictive accuracy and comprehensibility. It has been applied to classify Indian patients as having liver disease or not. Section 1 introduces the problem of rule mining and swarm intelligence. Section 2 surveys the current techniques for rule mining in the medical domain and identifies gaps in research. Section 3 describes our proposed algorithm which includes a novel rule discovery procedure and a novel rule list selection criterion. Section 4 provides comparison of results of the proposed algorithm with the other best known approaches. Section 5 concludes the paper and paves the way for future work.*

Keywords: Association rules, Particle Swarm Optimisation, Ant Colony Optimisation, Quality

1   Introduction

Data Mining is an analytic process designed to explore large amounts of data in search of consistent patterns and/or systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data. The discovered knowledge should be accurate and comprehensible. Knowledge comprehensibility can be achieved by using high-level knowledge representations. Association rules are the most intuitive choice. Association rule mining aims to extract interesting correlations, frequent patterns, associations or causal structures among sets of attributes or characteristics in the data repositories. An association rule is an implication in the form of X->Y, where X, Y are attribute-value pairs and $X \cap Y = \phi$. X is called antecedent while Y is called consequent, the rule means X implies Y. The rules can be used for classification if Y is a single attribute-value pair. There are two important basic measures for association rules, support and confidence. Support(s) of an association rule is defined as the percentage/fraction of records that contain X Y to the total number of records in the database. Confidence of an association rule is defined as the percentage of the number of transactions that contain X Y to the total number of records that contain X. Association rule mining problem is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those attribute-value combinations whose occurrences exceed a predefined threshold in the database; these are called frequent terms. The second problem is to generate association rules from these frequent terms by checking their combinations for constraints of minimal confidence. The first sub-problem can be further divided into two sub-problems: candidate frequent terms generation process and frequent terms generation. Those terms that are expected or have the hope to be frequent are called candidate terms. As the size of modern medical datasets is very large and constantly growing, most algorithms generate an extremely large number of association rules, often in thousands or even millions. Further, the association rules are sometimes very large containing multiple terms. It is nearly impossible for the end users to comprehend or validate such large number of complex association rules, thereby limiting the usefulness of the data mining results.

Evolutionary methods of soft computing paradigm can perform well in problems with vast search spaces and produce near optimal solutions. Soft computing is a paradigm that deals with imprecision, uncertainty, partial truth, and approximation to achieve tractability, robustness and low solution cost. An evolutionary algorithm (EA) is a subset of evolutionary computation; a generic population-based metaheuristic optimization algorithm. An EA uses some mechanisms inspired by biological evolution: reproduction, mutation, recombination, and selection. Swarm Intelligence is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from biological examples. These techniques can be used in data mining to find a target solution to the problem when other methods are difficult to implement.

2   Conventional Rule Mining Methods

2.1 Review of Literature

The AIS algorithm was the first algorithm proposed for mining association rules [1]. Subsequently, various algorithms based on Apriori [2], FP-Tree [3], Tree Projection, Bit Matrix, etc were proposed. These have been discussed extensively in various review papers.

The application of evolutionary approaches to rule mining started in [4] wherein the authors proposed an algorithm called AntMiner for extracting rules from a

medical database as a decision aid. They showed that the proposed algorithm achieved good predictive accuracy and a reduced number of rules at the same time. The main drawback that was cited was the computational cost.

Many variants of the Ant-Miner in terms of heuristic information, pheromone update, rule construction and pruning procedures have been reviewed in [5].

Mapping of medical data to transaction format was discussed in [6]. The combinatorial nature of association rules was discussed and methods to preprocess medical data were explored. The suitability of association rules for decision making support was further reinforced in [7] wherein rule mining was used to identify diseases that occurred together commonly, or less commonly than their individual frequencies in the population would predict.

The use of association rule mining with constraints and neural network technique based on back propagation to detect tumor was explored in [8]. Authors reported that neural network technique was less sensitive to database imbalance but had high training costs. ARM obtained better results on balanced database.

The role of evolutionary algorithms for rule mining and knowledge discovery was elaborated in [9]. The use of genetic algorithms for rule mining was done and it was found that GAs cope well with attribute interaction. GA has been used as a wrapper to select attributes for a constructive neural network. Genetic Programming also has the potential to create derived attributes with greater predictive power.

[10] was a seminal paper that proposed the use of Particle Swarm Optimization as a new tool for data mining. Three different Particle Swarm Data Mining Algorithms were implemented and tested against a Genetic Algorithm and a Tree Induction Algorithm (J48).From the obtained results, Particle Swarm Optimizers proved to be a suitable candidate for rule discovery tasks. The results obtained in these domains seem to indicate that Particle Swarm Data Mining Algorithms are competitive, not only with other evolutionary techniques, but also with industry standard algorithms such as the J48 and c4.5 algorithm, and can be successfully applied to more demanding problem domains.

Another method for rule mining based on an evolutionary (GA) approach –EGAR was described in [11] and compared with FP tree method. It was found that FPtree works well for discrete attributes whereas EGAR performs better with mix of discrete and continuous attributes.

An advanced swarm intelligence data mining algorithm was proposed in [12]. The method addressed issues such as missing value management and interactive rule extraction. The method was applied to select candidates for surgery for temporal lobe epilepsy. Four algorithms: decision tree, ant colony miner, PSO miner and the proposed hybrid PSO, were compared. It was found that C4.5 gave higher accuracy but at the cost of a more complex rule set and lower generalization. PSO shows a good convergence speed and is faster than ACO while having same performance. Convergence of C4.5 is very fast, but not recommended for small databases. The hybrid PSO is faster than other evolutionary variants but has slightly more memory usage. It uses a combination of support vector machines and radial basis functions in conjunction with PSO.

In [13], search constraints are introduced to find only medically significant association rules and make search more efficient. Association rules are compared to predictive rules mined with decision trees. Results provide evidence that decision trees are less effective than constrained association rules to predict disease with several related target attributes, due to low confidence factors (i.e. low reliability), slight over fitting, rule complexity for unrestricted trees (i.e. long rules) and data set fragmentation (i.e. small data subsets).

A hybrid particle swarm optimization/ant colony optimization (PSO/ACO) algorithm for the discovery of classification rules was proposed in [14]. Unlike a conventional PSO algorithm, this hybrid algorithm can directly cope with nominal attributes, without converting nominal values into binary numbers in a preprocessing phase. Authors compared the algorithm to an industry standard algorithm PART and its reduced version, coping only with continuous data, to the new classification algorithm for continuous data based on differential evolution. The results show that proposed algorithm is very competitive in terms of accuracy to PART and that it produces significantly simpler (smaller) rule sets, a desirable result in data mining—where the goal is to discover knowledge that is not only accurate but also comprehensible to the user. The results also show that the reduced PSO version for continuous attributes provides a slight increase in accuracy when compared to the differential evolution variant.

Association rule mining has been applied for discovering hyperlipidemia form biochemistry blood parameters in [15]. PSO/ACO approach to knowledge discovery was successfully applied in a pharmacovigilance context in [16]. The approach showed high accuracy to detect presence of previously undetected causal relationships between therapeutics, patient characteristics and adverse events.

Several additions to the standard GA intended to enhance the ability for an evolutionary system to perform rule discovery in data mining have been elaborated in [17]. Another evolutionary method for quantitative association rule mining is mainly motivated by (1) partition of quantitative attribute is not easy for every attribute and every user, (2) users, and even experts, usually feel difficult to specify the minimum-support, (3) the search space might be very large when we face quantitative attributes, and (4) the rules returned might be too many to deal with. The developed EARMGA algorithm can mine high quality rules without the user specifying minimum support or confidence threshold levels [18].

Large and dense databases with a huge amount of attributes can be mined through the combination of conventional GNP based mining method and a specially designed genetic algorithm (GA). The strategy consists of the division of a large and dense database into many small databases. These small databases are considered as individuals and form a population. Then the conventional GNP based mining method is applied to extract association rules for each of these individuals. Finally, the population is evolved through several generations using GA with special genetic operators. The results show that this method allows extracting association rules from large and dense databases directly and more efficiently than the conventional GNP method [19].

One of the major problems in pattern mining is the explosion of the number of results. The problem can be solved by using the Minimum Description Length principle (MDL): the best set of patterns is that set that compresses the database best. Using the Krimp algorithm for frequent itemset mining and classification, a dramatic reduction in the number of returned frequent item sets is obtained. These selections, called code tables, are of high quality. This is shown with compression ratios, swap-randomization, and the accuracies of the code table-based Krimp classifier, all obtained on a wide range of datasets. The algorithm shows high stability w.r.t. different candidate sets and it is parameter-free. There are many data mining tasks for which it can be used eg: frequent itemset mining while preserving privacy [20].

A data mining tool (DIFACONN-miner) was used for generating accurate classification rules for classifying causes of defects [21]. DIFACONNminer uses differential evolution (DE) algorithm for training ANNs and touring ant colony optimization (TACO) algorithm for generating classification rules. Fitness of ANN structure is evaluated according to a multiple objective function which consists of three performance measures namely error of ANN, number of rules and training accuracy. It was proven that DIFACONN-miner is able to produce accurate and effective classification rules.

A new evolutionary image classification algorithm namely Simplified Swarm Optimization (SSO) has been proposed recently. Image classification has faced a problem where the number of possible different combination of variables is very high. The performance of SSO, Particle Swarm Optimization (PSO) and Support Vector Machine (SVM) has been compared and analyzed. Unlike PSO, SSO does not need to use the velocity parameter and inertia weight; it uses only one random number and three predefined parameters to update each of the particle's position. SSO provides better classification accuracy than PSO and SVM. It has a potential to achieve higher classification result with smaller number of particles. Furthermore, PSO needs to allocate more memory than SSO for each particle to achieve better performance [22].

Four techniques based on swarm intelligence were studied and implemented in [23] and they provided accuracy comparable to other non SI based mining approaches. Shuffled frog leaping with new quality function showed very good results.

ABCMiner, an algorithm to mine rules using concepts from Artificial Bees Colony, was suggested in [24]. It showed good results in terms of accuracy for classification task. The new ABC data mining algorithm suggested new approaches in the following aspects: rule format, fitness value function, local search strategy and prediction strategy. This paper has presented an extension to Ant-Miner, named cAnt-Miner, which copes with continuous attributes during the rule construction process. By having the ability to create discrete intervals for continuous attributes "on-the-fly", cAnt-Miner does not require a discretization method in a preprocessing step [25].

A new strategy to discover a list of classification rules, which guides the search performed by the ACO algorithm using the quality of a candidate list of rules, instead of a single rule, has been proposed in [26]. The pheromone matrix used by the ACO algorithm is extended to include a tour identification that indirectly encodes the sequence in which the rules should be created, allowing a more effective search for the best list of rules.

2.2 Gaps in Literature

a) Dirty Data:
   Data in the real world is inherently dirty containing duplicate tuples, missing values, etc. The system needs to be designed to automatically deal with dirty data and avoid returning rules involving missing information.

b) Support for different data types:
   No single efficient scheme that works on all textual data types has been reported in literature. The algorithm should be such that it directly supports binary, nominal, categorical and continuous attributes.

c) High dimensionality small datasets:
   The medical datasets are usually of high dimensionality and typically contain a few thousand records. As the search space becomes larger, the computational feasibility should not get lost. This suggests an evolutionary kind of approach.

d) Interestingness framework:
   Support and confidence framework by itself is not enough to prune out uninteresting rules as medical data needs to find rules with low values of support. The requirement is to reduce the number of false positives. This will require some modification of the fitness function defined in terms of support and confidence.

e) Trivial patterns:
   Certain combinations of attributes may be trivial and should be grouped or eliminated based on previous knowledge of the domain.

f) Explanatory model:
   The discovered knowledge should not be a black box which makes predictions without explaining them. The user may not trust such a system.

g) Predictive accuracy:

Last but not the least, no single approach reported in literature consistently outperformed the other in terms of accuracy, though swarm based approaches gave good results. Results were highly dependent on the datasets. A hybrid approach might help in this regard.

3   Proposed Method: HybridMiner

Hybrid algorithms which combine concepts from ACO and PSO can deal with all types of attributes. In a pilot study conducted in [23], these methods have shown reasonably good accuracies in the range of 91-94% while maintaining the comprehensibility of the rules as measured using size of rule and rule sets. To tailor the system to medical domain, a new fitness function can be embedded into the process. The major advantage of using evolution in the discovery of prediction rules is that they perform a global search based on a greedy approach.

The proposed algorithm based on ACO/PSO uses a sequential covering approach to discover one classification rule at a time.

AlgoCover is :
RSglobal = {}
RSlocal = {}
FOR EACH class C
TS = {All training examples in dataset}
WHILE (Number of uncovered training examples belonging to class C >MaxUncovExampPerClass)
Run the AlgoCore to discover best nominal rule predicting class C called Rule
Run the modified PSO algorithm to add continuous terms to Rule, and return the best discovered rule BestRule
Prune BestRule
RSlocal = RSlocal   BestRule
TS = TS −{training examples covered by discovered rule}
ENDWHILE
END FOR
Prune RSlocal
Order rules in RSlocal by descending Quality
If Quality(RSlocal)>Quality(RSglobal)
then
RSglobal=RSlocal

Nominal attributes are handled by the AlgoCore:
Initialise population
REPEAT for MaxInterations
FOR every particle x
Set Rule Rx = "IF {null} THEN C"
FOR every dimension d in x
Use roulette selection to choose whether state should be set to off or on. If it is on then the corresponding attribute-value pair set in the initialization will be added to Rx; otherwise nothing will be added.
LOOP
Calculate Quality Qx of Rx
P = x's past best state
Qp = P's quality
IF Qx>Qp

Qp = Qx
P = x
END IF
LOOP
FOR every particle x
P = x's past best state
N = the best state ever held by a neighbor of x according to N's quality QN
FOR every dimension d in x
IF Pd = Nd THEN pheromone entry corresponding to the value of Nd in the current xd is increased by Qp
ELSE IF Pd = off AND seeding term for xd ≠ Nd THEN pheromone entry for the off state in xd is increased by Qp
ELSE
pheromone entry corresponding to the value of Nd in the current xd is increased by Qp
END IF
Normalize pheromone entries
LOOP
LOOP
LOOP
RETURN best rule discovered
The rule discovery, fitness evaluation and pruning of individual terms is the same as for ACO/PSO with PF [23].

We suggest a modified PSO to handle the continuous attributes. Diversity is introduced by selecting the exemplar particles from a prespecified region or neighbourhood rather than randomly. All other particles' past best information is used to update a particle's velocity. The velocity updating is done according to a vector from the particle's region only. But the regions are reconstructed at fixed prespecified points of time in the execution of the algorithm. This provides a good mix between the exploration and exploitation properties of the algorithm.

The particle learns simultaneously from its own best known position as well as the global best. We update only one dimension and not all the dimensions of all the velocities of the particles at the same time. We construct a vector for each particle which indicates which other particle's personal best should this particle learn from. Fi=|fi1, fi2,..fid|.

If the fitness of a particle does not increase for a fixed number of iterations(parameter *iter*) then a random number between 0 and 1 is chosen. If this number is greater than Pc, then fid=i. If it is less than Pc, then the particle i learns from some other particle's personal best in the same region as given by vector Fi. Pc is parameter that controls how frequently learning occurs.

Any connected topology can be used to define regions. The length of region controls number of particles in it. The vector to be optimized consists of two dimensions per continuous attribute, one for an upper bound and one for a lower bound. At every particle evaluation, the vector is converted to a set of terms and added to *Rule* produced by the algorithm for fitness evaluation. If two bounds cross over, both terms are omitted from decoded rule, but *Personal Best position* is still updated in those dimensions using (1)

$$vid = \chi \, (vid + c1\varphi1(P_{id}^{fid} \text{-}xid) + c2\varphi2(Pgd \text{-}xid)))$$
$$xid = xid + \, vid \qquad\qquad (1)$$

where vid is the dimension velocity, xid is the particle position, P_{id}^{fid} denotes the corresponding dimension d of the ith particle's own pbest or the exemplar's pbest, Pgd is the best position in the neighborhood, χ is constriction coefficient, $\varphi1$ and $\varphi2$ are random weights, c1 and c2 are constants.

A particle operates within its own region. Each particle's initial position is set to a uniformly distributed position between the value of a randomly chosen seed (from within same region) example's continuous attribute and that value added to the range for that attribute (for upper bound) and at a uniformly distributed position between an example's value and an example's value minus the range for that attribute (for lower bound).

Quality, Q is defined using Precision as given by:

Laplace-corrected Precision = $(1+TP)/(1+TP+FP)$
If TP<MinTP, Q=Laplace-Corrected Precision*0.1,
ELSE Q=Laplace-Corrected Precision   (2)
whereMinTP is the least number of correctly covered examples that a rule has to cover [23].

The second modification is to evaluate the quality of not just the individual rules in rule set, but also evaluate the complete rule set for quality. This is done using function:

Quality(RS) =
$1\text{-} (\sum_{r=1}^{N} (TPn + FPn).U_{CF}(FPn,TPn + FPn))/S$ (3)

where TPn, FPn are the number of true positives and false positives of nth rule, S is number of training examples and U_{CF} is the error rate[26].

4 Experimental Setup

4.1 Database

We have used the Indian Liver Patient Dataset [27] to check the performance of our algorithm. This data set contains 10 variables that are as follows: age of the patient, gender of the patient, Total Bilirubin, Direct Bilirubin, Alkaline Phosphotase, Alamine Aminotransferase, Aspartate Aminotransferase, Total Protiens, Albumin and Albumin and Globulin Ratio. The task is to infer whether a patient has liver disease or not and to use rule based methodology for this classification.

4.2 Parameter Settings

Similar parameter settings have been used as in hybrid ACO/PSO with PF[23]. For ACO, the following parameter values were taken: Number of Ants=500, Minimum number of records per rule=15, maximum number of uncovered records=20(usually 10% of total records of class) and number of rules to test ant convergence=20. For PSO, number of particles=20 and

number of iterations=50. ACO pruning was used if rule has less than 20 terms. The value for minimum number of true positives=15, constriction factor χ=0.729, social and personal learning coefficients, c1=c2=2.05. Maximum number of uncovered examples per class was set to 20. We assume a ring topology where the length of region is taken as 5. The number of iterations for regrouping is set to 7. The learning probability P_c is varied from 0 to 0.5. *Iter* parameter is set to 15.

5 Results and Analysis

A comparison of our proposed HybridMiner algorithm has been done with well known algorithms: c4.5, PART, ACO/PSO with PF and cAnt-Miner. The first criterion used to analyze the performance of the various implemented techniques is predictive accuracy, defined in terms of cross validation accuracy rate, which in turn equals quotient between number of test cases correctly classified and the total number of test cases. A k-fold cross validation was used with value of k=10. This is a standard technique used to evaluate accuracy of data mining techniques. The other two criteria for performance evaluation are the average number of rules in a rule set and the average number of attribute value combinations or conditions per rule. Rules that are shorter in length are more effective since they are easy to comprehend. Also the length of the entire rule set should be less in order for the rules to be usable.

Table I summarizes the results obtained by the c4.5, PART, ACO/PSO with PF, cAnt-Miner and HybridMiner.

TABLE I. COMPARISON BETWEEN c4.5, PART, ACO/PSO WITH PF, cAntMiner, HybridMiner

Algorithm	Accuracy	No. of terms in a rule	No. of rules in ruleset
C4.5	94.13	4.86	16.20
PART	93.84	4.72	15.83
ACO/PSO with PF	96.21	4.69	18.52
cAntMiner	95.13	3.24	8.32
HybridMiner	96.42	2.12	6.86

Results indicate that our proposed algorithm provides improvement in accuracy and also reduces the average rule length and rules set size. This is because each individual rule is pruned based on a quality measure in every iteration. Additionally an individual rule is added to the rule set only if it results in an increased quality of the rule set.

6 Conclusion and Future Scope

This paper proposed an algorithm for mining association rules and using them for classification of liver patients as

fit or unfit. It modified the ACO/PSO algorithm in two major ways. Firstly, better results can be obtained by improvements in the traditional PSO which allow learning of particle in a region of the search space i.e. exploitation, and then reformation of regions to allow for exploration. The search can be better guided this way. Secondly, the quality of the entire set of rules as well as each individual rule in rule set is evaluated. This should contribute to the accuracy of the entire rule list and be able to handle effect of rule interactions within the same list. Proposed algorithm: HybridMiner performs best in terms of simplicity and accuracy over liver dataset. One possible further research direction is to explore the effect of region formation and learning probability parameters on the performance of the algorithm. Also different topologies for the regions may be required for good performance over other datasets.

7   References

[1] Agrawal, R., Imielinski, T., and Swami, A. N., "Mining association rules between sets of items in large databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp 207-216, 1993.

[2] Agrawal, R. and Srikant, R., "Fast algorithms for mining association rules", in Proceedings of 20th International Conference on Very Large Data Bases, pp 487-499, 1994.

[3] Han, J. and Pei, J. , "Mining frequent patterns by pattern-growth: methodology and implications", ACM SIGKDD Explorations Newsletter 2, pp 14-20, 2000.

[4] Parpinelli, R.S., Lopes, H.S., and Freitas, A.A., "Data Mining with an Ant Colony Optimization Algorithm", IEEE Transactions on Evolutionary Computation, special issue on Ant Colony algorithms, pp 321-332, 2002.

[5] D. Martens, B. Baesens, et. al., "Editorial Survey: swarm intelligence for data mining", Machine Learning, vol. 82, No. 1, pp.1-42, 2011.

[6] Carlos Ordonez, Cesar Santana, Levien de Braal, "Discovering interesting association rules in medical data", Proceedings of the ACM SIGMODD workshop on Research Issues in Data Mining and Knowledge Discovery, 2000.

[7] Goldacre, Kurina et al, "Use of large medical databases to study associations between diseases, International Journal Q J Medicine", pp 669-675, 2000.

[8] Maria-Luiza Antonie, Osmar R. Za¨ıane, Alexandru Coman, "Application of Data Mining Techniques for Medical Image Classification", Proceedings of ACM/ SIGKDD, 2001.

[9] Alex A. Frietas, "A survey of evolutionary algorithms for data mining and knowledge discovery", Advances in Evolutionary Computing, Springer-Verlag , pp 819-845, 2003.

[10] Sousa, T., Silva, A., Neves, "A Particle Swarm based Data Mining Algorithms for classification tasks", Journal of Parallel Computing, pp 767–783, 2004.

[11] Halina Kwasnicka and Kajetan Switalski, "Discovery of association rules from medical data - classical and evolutionary approaches", Proceedings of Meeting of Polish Information Processing Society Conference, pp. 163–177, 2005.

[12] M. Ghannad-Rezaie,et al, "Medical Data Mining using Particle Swarm Optimization for Temporal Lobe Epilepsy", in Proceedings of IEEE Congress on Evolutionary Computation, 2006.

[13] Carlos Ordonez, "Comparing Association Rules and Decision Trees for Disease Prediction", Proceedings of ACM HIKM'06, USA, 2006.

[14] Holden, N., Freitas, A., "A Hybrid PSO/ACO Algorithm for Discovering Classification Rules in Data Mining", Journal of Artificial Evolution and Applications (JAEA), 2008.

[15] Sengul Dogan and Ibrahim Turkoglu, "Diagnosing hyperlipidemia using association rules", Mathematical and Computational Applications, Vol. 13, No. 3, pp. 193-202, 2008.

[16] Margarita Sordo, Shawn N. Murphy, "A PSO/ACO Approach to Knowledge Discovery in a Pharmacovigilance Context", Harvard Medical School, Charlestown, MA. USA.

[17] Robert Cattral,et al, "Techniques for Evolutionary Rule Discovery in Data Mining", Proceedings of IEEE Congress on Evolutionary Computation, 2009.

[18] Xiaowei Yan, et al, "Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support", Journal on Expert Systems with Applications, Expert Systems with Applications 36, pp 3066–3076, 2009.

[19] Eloy Gonzales, et al, "Combination of Two Evolutionary Methods for Mining Association Rules in Large and Dense Databases", Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.13 No.5, pp 561-72, 2010.

[20] Jilles Vreeken, Matthijs van Leeuwen, Arno Siebes, "KRIMP: mining itemsets that compress", Data Mining and Knowledge Discovery Journal, No 23, pp 169–214, 2011.

[21] Adil Baykasoglu, Lale Ozbakir, Sinem Kulluk, "Classifying defect factors in fabric production via

DIFACONN-miner: A case study", Elsevier Journal on Expert Systems with Applications 38, pp. 11321–11328, 2011.

[22] Noorhaniza Wahid, "A Novel Image Classification Algorithm Using Swarm-Based Technique for Image Database", Ubiquitous Computing and Multimedia Applications, Springer-Verlag Journal, Part II, Vol 151, pp. 460–470, 2011.

[23] Veenu Mangat, "Natural Intelligence based knowledge discovery for medical practitioners", Intelligent Automation and Systems Engineering, Lecture Notes in Electrical Engineering, Vol 103, pp. 53-64, 2011.

[24] Mohd Afizi Mohd Shukran et al, "Artificial Bee Colony Based Data Mining Algorithms for Classification Tasks", Modern Applied Science, Vol 5, No 4, 2011.

[25] Fernando E. B. Otero, Alex A. Freitas and Colin G. Johnson, "cAnt-Miner: An Ant Colony Classification Algorithm to Cope with Continuous Attributes", Ant Colony Optimisation and Swarm Intelligence, Lecture Notes in Computer Science, Vol 5217, pp 48-59, 2008.

[26] Fernando E. B. Otero, Alex A. Freitas and Colin G. Johnson, "A New Sequential Covering Strategy for Inducing Classification Rules with Ant Colony Algorithms", IEEE Transactions on Evolutionary Computation, 2012.

[27] UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science

Improved Multi-objective PSO for Semi-desirable Facility Location Problem

Bing Qi[1], Fangyang Shen[2], Heping Liu[3] and Terry House [4]

[1]Department of Computer Science, Methodist University, Fayetteville, NC, USA
[2]Department of Computer Science and technology, New York city College of Technology, Brooklyn, NY, USA
[3]Department of industrial Engineering, Auburn University, Auburn, NC, USA
[4]Department of Computer Science, Methodist University, Fayetteville, NC, USA

Abstract— *Evolutionary optimization algorithms have been used to solve multiple objective problems. However, most of these methods have focused on search a sufficient Pareto front, and no efforts are made to explore the diverse Pareto optimal solutions corresponding to a Pareto front. Note that in semi-obnoxious facility location problems, diversifying Pareto optimal solutions is important. The paper therefore suggests an improved multi-objective particle swarm optimization algorithm (MOPSO) to find diversified Pareto optimal solutions in the parameter space for semi-obnoxious facility location problems while achieving a similar Pareto front in the objective space. The improvement of MOPSO is obtained by introducing a new mechanism based on distances among Pareto optimal solutions. Three semi-obnoxious facility location problems from the literature are used to evaluate the performance of the improved MOPSO. The results indicate that the approach is promising, being able to expand the diversity of non-dominated/Pareto Optimal solutions while acquiring a similar Pareto front.*

Keywords: Pareto front, Multi-objective particle swarm optimization, semi-obnoxious facility location, non-dominated optimal solutions.

1. Introduction

The semi-obnoxious facility problems aim to determine an optimal location for a new facility within a bounded region [1]. The optimal facility location needs to minimize undesirable effects produced by the new facility itself, and at the same time minimize total transportation costs to other demanded sites [1]–[3]. It is clear that semi-obnoxious facility problems are bi (multi)-objective optimization problems. Traditionally, mathematical programming techniques are used to solve multi-objective optimization problems (MOPs). These techniques, however, are sensitive to the shape of Pareto front. When a Pareto front is concave or disconnected, they may not work well. However, a typical semi-obnoxious facility problem usually has two objective functions with several constraints, and usually it is not concave, which brings more challenges to search an optimal solution.

To solve any MOPs like the semi-desirable facility location problem, A theoretically ideal solution is to find the entire Pareto optimal solution set that are non-dominated with each other. However, it is impossible to identify all theoretically existing Pareto optimal solutions, since the number of Pareto optimal solutions is infinite in real-life multi-objective problems. Hence, current practical methods normally focused on generating a sufficient Pareto front that is uniformly distributed over the true Pareto front [4], [5]. For a given computational limit, no much effort has been made to expand the spectrum of the corresponding Pareto optimal solutions. However, the diversity of the Pareto optimal solution is equally important in some decision-making problems, especially in semi-desirable facility location problems, since diverse Pareto optimal solutions can bring more choices of facility location to decision makers in order to enhance the robustness of their decision.

In this paper, under the condition of gaining the same Pareto front in the objective space, we improved the current Multi-Objective Particle Swarm Optimization (MOPSO) by extending the diversity of the corresponding Pareto optimal solutions in the parameter space. To achieve this goal, instead of randomly deleting a non-dominated solution in the most crowded cube when the external archive was full, we incorporated a new mechanism to compute the physical distance between optimal solutions in parameter space and maintained those that are mostly away from the crowding populations. We compared our new algorithm with the original MOPSO on a suit of semi-desirable facility location problems, and the result showed that the new mechanism is capable of expanding the diversity of the solutions in the parameter space and meanwhile maintaining a similar Pareto front in the objective space.

The remainder of the paper is organized as follows. In the second section, evolutionary algorithms for multi-objective optimization are introduced. In the third section, the improvement of algorithm is provided. In the fourth section, the computational experiments from three cases are presented. Finally, some remarks along with future research directions are given to conclude the paper.

2. Evolutionary Multi-objective Optimization

2.1 Foundation of Multi-objective Optimization Problems

Multi-objective optimization involves the process of simultaneously optimizing two or more conflicting objectives subject to certain constraints. Considering the following typical multi-objective problem which aims to optimize n objectives:

$$\begin{cases} \text{minimize} f(x) = \{f_1(x), f_2(x), ...f_n(x)\} \\ \text{subject to } g_i(x) \geq 0 \text{ i} = 1, 2, ...n \\ \text{where } x = (x_1, x_2, ...x_m) \in R \end{cases} \quad (1)$$

Where x is an m dimensional vector, $x_i(i = 1, ..., m, m >= 1)$ is in the parameter/solution space R. $f(x)$ is the objective vector and it is composed of n different objective functions $f_i(i = 1, ..., n)$. These objective functions could be either linear or non-linear functions. $g_i(x)$ is the constraint for the corresponding objective functions. In many real-life problems, objectives under consideration may compromise with each other. Thus it is hard to find a "perfect" solution to minimize all objectives. Thus, to solve such multiple objective problems, a feasible solution would be the best balance among these objectives, which satisfies all the objectives at certain level and is not dominated by any other solutions at the same time [6]. This kind of solution is considered as a Pareto optimal solution. The corresponding objective function values are called the Pareto front [7]. (MOPSO) [4], [5].Currently, many different algorithms have been proposed to deal with MOPs, such as the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [8], Strength Pareto Evolutionary Algorithm (SPEA) [9] and Multi-Objective Particle Swarm Optimization(MOPSO).

2.2 Evolutionary Multi-objective Optimization

Particle Swarm Optimization (PSO) is a stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995 and is inspired by social behavior of bird flocking or fish schooling [10]–[13]. It has been used to solve many different nonlinear and multi-modal problems. Although PSO is originally designed to solve single objective problems, it has been widely extended to deal with multi-objective optimization problems due to its relative simplicity and population-based techniques [14]–[17]. Multi-objective Particle Swarm Optimization (MOPSO) is one of the most popular algorithms that is based on PSO and has been proved highly competitive compared with other evolutionary multi-objective optimizat ion techniques [4]. For any single objective problem, it is obvious to identify the *pbest* and *gbest* because there is only one objective to be evaluated. However, for any multiple objective problem, which has more than one objective to be optimized, a new approach

needs to be defined to compare alternative solutions against each other. Thus, the major modifications for the MOPSO are the selection process of *pbest* and *gbest* to guide all particles in the right direction.

Currently, the MOPSO algorithm uses Pareto preference to select *pbest* and *gbest*. To be more detail, a solution in multi-objective problem is referred to as a non-dominant or Pareto optimal solution. This Pareto optimal solution is not dominated by any other solution in the parameter space and it has better or at least no worse performances in all objectives. Hence, a Pareto optimal solution means the improvement of one objective could only be achieved at the expense of worsening at least another objective. The corresponding objective function values regarding to such Pareto optimal solutions are called Pareto front. Therefore, to solve any multi-objective problem, we not only search for a single solution but rather a set of solutions or non-dominated archives.

With the original MOPSO algorithm, each particle needs to keep tracking the best solution it has found so far, which acts as a *pbest*. At the same time, an external memory or repository is generated to store the non-dominant solutions identified by all the particles. On every flight cycle, each particle independently selects a non-dominant solution from the repository as its current *gbest* to guide its next direction. At the end of the computation, the non-dominant solutions or Pareto front stored in the repository are considered as the final solutions. Thus, it is significant to store the most informative or important non-dominant solutions into the repository. Since the size of the repository is limited, when some new non-dominant solutions are generated after one flight cycle but the repository is full, the less informative non-dominant solutions need to be eliminated in order to accommodate the more informative new solutions. To achieve this goal, the MOPSO uses a geographically based method. It divides the current Pareto front space into cubes and gives preference to eliminate those solutions that are located in more populated cubes. After the MOPSO has identified the cube with the most crowded particles based on the Pareto front values, it randomly evicts one non-dominant solution from the specific cube. As a result, the MOPSO is able to keep the diversity of the Pareto front by eliminating the Parteo front's "redundant" solutions.

2.3 An Improved Evolutionary Multi-objective Optimization Algorithm

However, MOPSO only focuses on the diversity of the Pareto front without considering the corresponding diversity of the Pareto optimal solutions. In real world, the diversity of Pareto optimal solutions is also or even more important. For example, for any multi-objective problem, under the condition of obtaining the similar objectives (the similar Pareto front), it will be more meaningful for any multi-objective problem if more diverse options (more Pareto
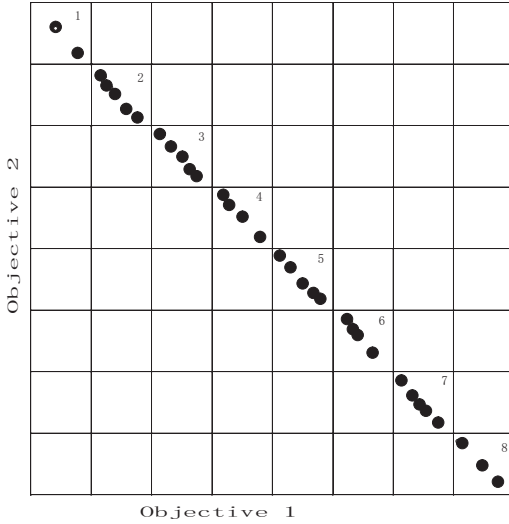
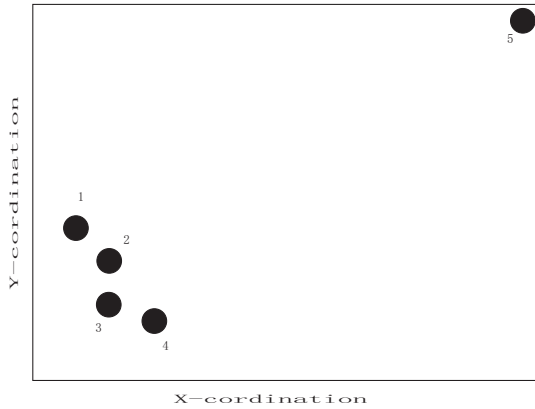Fig. 1: Geographically-based Pareto Front Distribution

Fig. 2: Corresponding Non-dominated Solution in the Parameter Space

optimal solutions) can be located. Suppose Figure 1 indicates the current non-dominant Pareto front in the repository. In this figure, the whole objective solution space is divided into 8×8 cubes based on the Pareto front value. In case one more informative non-dominant solution needs to be inserted in but the current repository is full, the MOPSO will first select one of the most populated cubes, which are cube 2,3 5 and 7 since they all contain the same number of non-dominant Pareto front solutions. Let's assume the MOPSO select cube 3. Based on Pareto front values, those 5 solutions within cube 3 have no much difference since they are locating in the same small cube and very close to each other. As a result, by randomly selecting one solution to eliminate, the MOPSO can still keep the diversity of Pareto front. However, the corresponding Pareto optimal solutions may not be close to each other even if their objective values are almost same. Suppose Figure 2 indicates the corresponding Pareto op-

timal solutions regarding to the Pareto front solution in cube 3. Apparently, one of the Pareto optimal solutions (solution 5) is very far away from the others. To achieve one similar specific objective value, it means that this specific Pareto optimal solution can give the decision maker a very different option compared with other solutions in the same cube. Therefore, solution 5 is a more informative non-dominant solution compared with other solutions. (a) Divide the current Pareto front space into tiny cubes based on the geographical representation of the objective values.

(b) Measure the number of non-dominant Pareto front solutions in each cube and randomly select the most crowded cube (with the most population in it).

(c) For the given cube, the corresponding Pareto optimal solutions are plotted based on the parameter/solution space.

(e) For any one non-dominant solution in the cube, the physical distances between this solution and the other solutions are measured.

(f) Finally identify the non-dominant optimal solution that has the minimal distance from the others. And this solution will be considered as a redundant solution both in Pareto front space and Pareto optimal solution space. It will be eliminated in order to accommodate other more informative solution.

3. Semi-desirable Location Problem

The semi-obnoxious facility problem is a popular realistic problem. It intends to identify an ideal new facility location that can minimize the undesirable effects produced by the new facility itself as well as minimize the total transportation costs to other demanded points [1]–[3]. For this semi-obnoxious facility problem, it is extremely necessary to generate as diverse Pareto optimal solutions in the parameter space as possible regarding to a similar Pareto front. A semi-obnoxious facility problem can be modeled as a bi-objective optimization problem. which are modeled as the following [18]:

(a) **Objective 1: Minimization of transportation costs**

$$\text{Minimize } F_1(x) = \sum_{i=1}^{n} w_{i1} d(x, a_i) \qquad (2)$$

Where i is the index for the fixed points $i = 1, 2...n$. w_{i1} is the unit weight associated with transportation costs to the fixed points a_i. And $d(x, a_i)$ is the Euclidean distance between the facility location x to the fixed point a_i.

(b) **Objective 2: Minimization of undesirable effects**

From the literature [18], three models have been proposed to represent this problem.

• The first model has the following form:

$$\text{Minimize } F_2(x) = \sum_{i=1}^{n} w_{i2} [d(x, a_i)]^{-1} \qquad (3)$$

Table 1: Parameters For All Test Problems

	i	1	2	3	4	5	6	7
Test 1 and 2	a_i	(5,20)	(18,8)	(22,16)	(14,17)	(7,2)	(5,15)	(12,4)
	w_1	5	7	2	3	6	1	5
	w_2	1	1	1	1	1	1	1
Test 3	a_i	(4,4)	(8,7)	(11,10)	(13,4)			
	w_1	2	4	3	2			
	w_2	1	1	1	1			

Table 2: Extra Data for Test Problem 2

Parameters	Value
M	200
m	1
d_1	10
d_2	30

As the definition for objective 1, i is the index of fixed points a_i. w_{i2} is the unit weight regarding to the undesirable effects to fix point a_i.

- The second model uses the following criterion to calculate the obnoxious effects (1) the undesirable effects of the facility do not decrease much upon a small increase in the distance when the facility is very close to one certain fixed point; (2) the undesirable effects of the facility do not increase much upon a small decrease in the distance when the facility is far from a population center; (3) the undesirable effects are constant within a certain distance. It is defined as the following:

$$F_2(x) = \sum_{i=1}^{n} \begin{cases} M \text{ if } d(x, a_i) \leq d_1 \\ M - m(d(x, a_i)) \text{ if } d_1 < d(x, a_i) < d_2 \\ 0 \text{ if } d_2 \leq d(x, a_i) \end{cases}$$

(4)

Like before, i is the index of fixed points $i = 1, 2...n$ and $d(x, a_i)$ is the distance between the fixed point a_i to the facility location point x. M and m are assigned to different values depending on different facility location problems. Similarly, d_1 and d_2 also can be set to different thresholds with respect to the specifics of the real-life problem.

- The third model applies a maximum function to minimize the undesirable effects of the fixed points, which can be represented as follows:

$$\text{Maximize } F_2(x) = \min_{1 \leq i \leq n} (w_{i2} d(X, a_i)) \qquad (5)$$

4. Computational Experiments

In the paper, the original MOPSO and the improved MOPSO algorithm are applied independently to solve the facility location models using three test problems from the literature. These three test problems are all bi-objective optimization problems, in which the first objective function are all based on the Objective 1's model showed on equation 2. Test problem 1, 2 are adopted from [19] directly, where the second objective function in test problem 1 is based on the equation 3 and test problem 2's second objective function

model is based on the equation 4. Test problem 3 utilizes a weighted minimum function to minimize the transportation costs and a maximum function to model the obnoxious effects of the facility over the fixed points which is adopted directly from [2], its second objective function relies on the model showed on the equation 5. All the parameters and the distance metrics used for these test problems are set exactly same as in their corresponding references. Table 1 shows the parameters for these 3 test problems. The extra parameters used by test problem 2 are set as in Table 2. In the experiments, both MOPSO and improved MOPSO algorithms use 100 particles, a repository size of 100 particles and a mutation rate of 0.1. Meanwhile, we divide the Pareto front space into 30×30 cubes and select the global best from the top 10% sorted repository. For both algorithms, we randomly generate the initial population to restrict the influence of random effects and repeat thirty times for each test problem. Our experimental results indicate that each individual experiment produced very similar result. Thus, we randomly select one experiment to plot the result.

Since the objective of this paper is to improve the original MOPSO by extending the diversity of Pareto Optimal solutions with the guarantee of obtaining the similar Pareto front, we plot both Pareto front and Pareto optimal solution to do the comparison for each test problem.

Figure 3, Figure 4 and Figure 5 show the experimental results for test problem 1,2 and 3, respectively. In all figures, the line marked with $\triangle$ indicated the results obtained from the original MOPSO and the one marked with $\bullet$ represented the results generated by our improved MOPSO algorithm (iMOPSO). For each test problems, the sub-figure displayed on the left means the Pareto front value in the objective space and the right sub-figure shows the corresponding Pareto Optimal solutions in the parameter space. As shown in these figures, only test 2 has discontinuities in the Pareto front which is due to the three-piecewise structure of the second objective model/function. As expected based on the Pareto front sub-figures, both MOPSO and iMOPSO are able to obtain a very similar Pareto front set that distributed uniformly over their true Pareto front for all three test problems. This is because iMOPSO uses the same method to select the top 10% non-dominant solutions from the repository as *gbest* for each particle to guide its flight. Therefore, we concluded that the iMOPSO algorithm is as effective as the original MOPSO regarding of finding the Pareto front as well as keeping the diversity of Pareto front.

Regarding to the corresponding Pareto optimal solutions in their parameter spaces, we found that iMOPSO can always cover the Parteo optimal solutions that identified by MOPSO for all test problems. In addition to that, we can see from the Figure 3 that iMOPSO find much more diversified Pareto optimal solutions especially in the area when x locate in $-50 < x < 0$. Most Pareto optimal solution identified by MOPSO is crowded in some certain
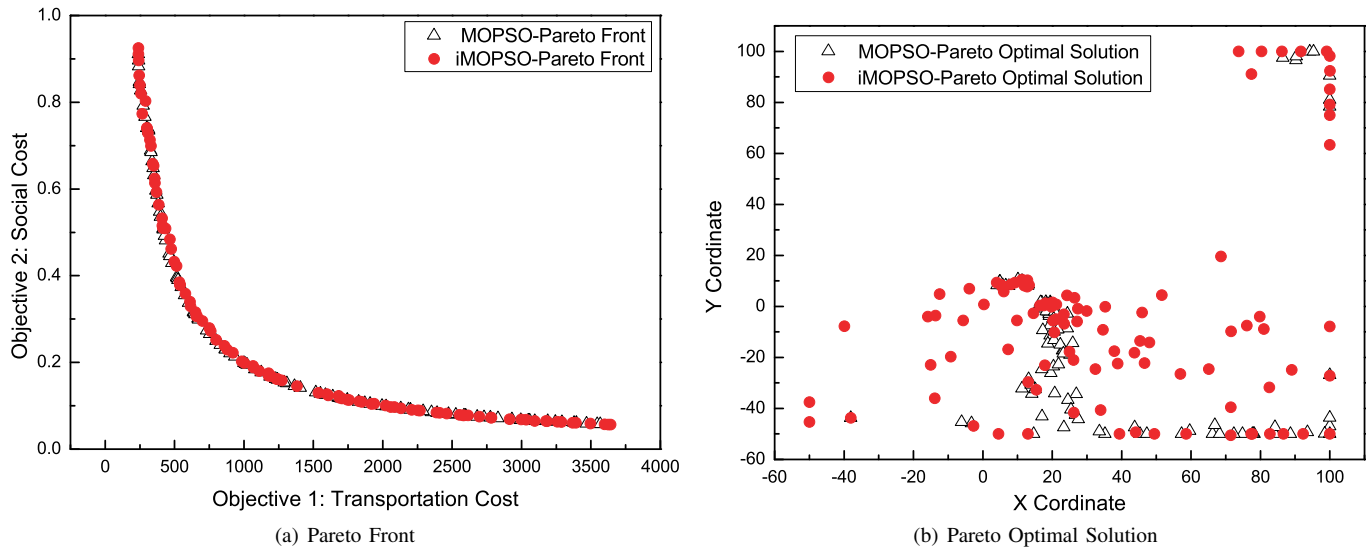
(a) Pareto Front

(b) Pareto Optimal Solution

Fig. 3: Test Problem 1

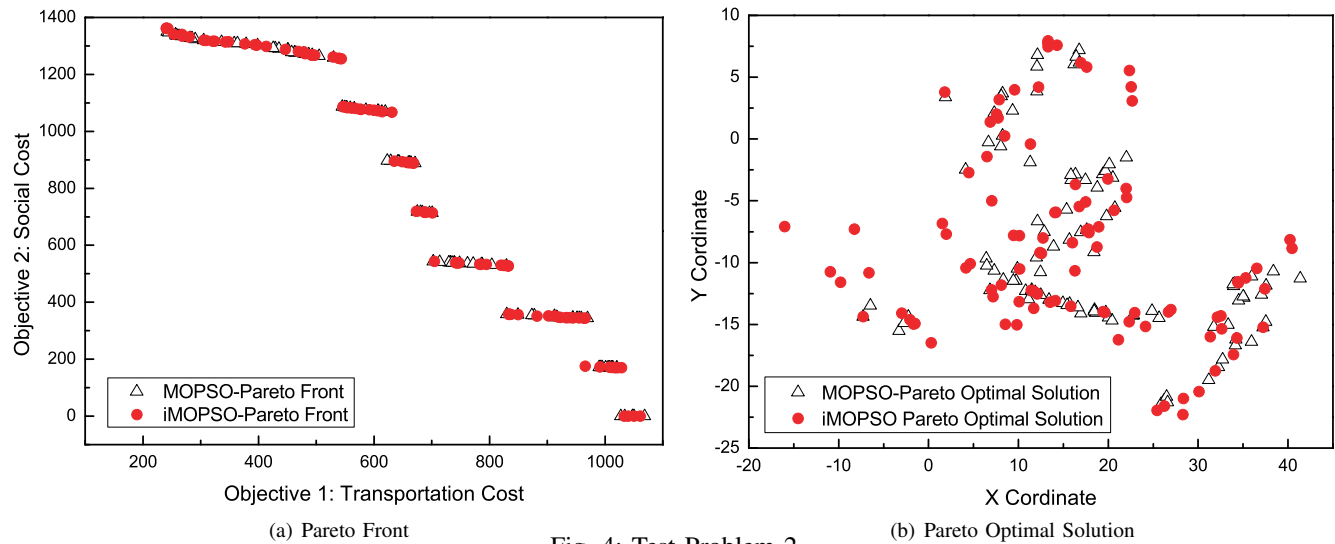(a) Pareto Front

(b) Pareto Optimal Solution

Fig. 4: Test Problem 2

places because MOPSO only depends on Pareto front value to update its repository. Similarly, the Figure 4 confirms that the iMOPSO is able to explore a wider solution space and obtain a more distributed population along the Pareto optimal solution than the MOPSO. However, for test problem 3 indicated in Figure 5, both algorithms have obtained a very similar Pareto optimal solutions.

5. Conclusions

In this paper, we propose an improved Multi-objective Particle Swarm Optimization method based on the original MOPSO algorithm. To verify the effectiveness of our method, we apply the improved algorithm to solve the semi-desirable facility location problem model by using three different test cases from literature. We compare extensively the performance against the original MOPSO and the experimental results demonstrate that our new algorithm outperforms MOPSO by being able to explore more Pareto optimal solutions and find more diverse Pareto optimal solutions in the parameter space corresponding to the similar Pareto front in the objective space.

References

[1] E. Erkut and S. Neuman, "Analytical models for locating undesirable facilities," *European Journal of Operational Research*, vol. 40, pp. 275–295, 1989.

[2] E. Melachrinoudis and Z. Xanthopulos, "Semi-obnoxious single facility location in euclidean space," *Comput. Oper. Res.*, vol. 30, no. 14, pp. 2191–2209, 2003.

[3] E. Melachrinoudis, "Bicriteria location of a semi-obnoxious facility," Tarrytown, NY, USA, 1999.

(a) Pareto Front

(b) Pareto Optimal Solution
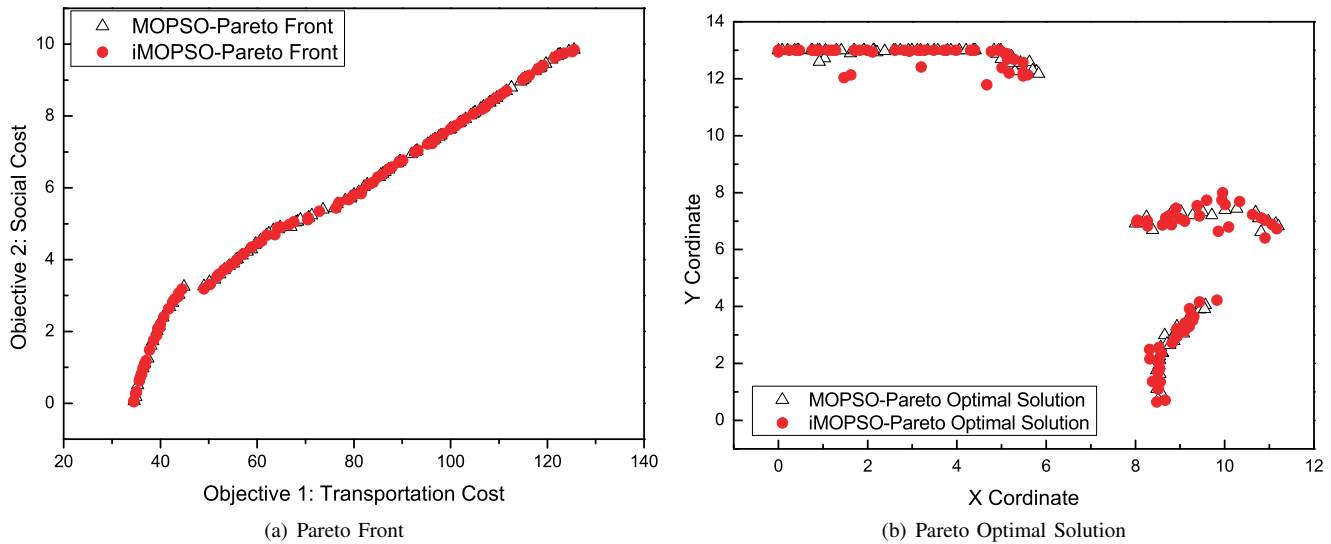
Fig. 5: Test Problem 3

[4] C. A. C. Coello and M. S. Lechuga, "Mopso : A proposal for multiple objective particle swarm," *Evolutionary Computation*, vol. 2, pp. 1051 – 1056, 2002.

[5] M. L. CAC Coello Coello, GT Pulido, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, 2004.

[6] A. E. S. Abdullah Konak, David W. Coit, "Multi-objective optimization using genetic algorithms: A tutorial," *Reliability Engineering & System Safety In Special Issue - Genetic Algorithms and Reliability,*, vol. 91, pp. 992–1007, 2006.

[7] K. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley and Sons, 2001.

[8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, 2002.

[9] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, vol. 2, pp. 95–100, 2001.

[10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, 1995, pp. 1942–1948.

[11] J. K. Russell C. Eberhart, Yuhui Shi, *Swarm Intelligence (The Morgan Kaufmann Series in Evolutionary Computation)*. Morgan Kaufmann, 2001.

[12] R. C. Eberhart and Y. Shi, "Comparison between genetic algorithms and particle swarm optimization," in *EP '98: Proceedings of the 7th International Conference on Evolutionary Programming VII*. London, UK: Springer-Verlag, 1998, pp. 611–616.

[13] J. Kennedy, "The particle swarm: social adaptation of knowledge," *Evolutionary Computation,*, pp. 303–308, 1997.

[14] J. E. Fieldsend, E. Q. Uk, and S. Singh, "A multi-objective algorithm based upon particle swarm optimisation, an efficient data structure and turbulence." 2002.

[15] J. E. Alvarez-benitez, R. M. Everson, and J. E. Fieldsend, "A mopso algorithm based exclusively on pareto dominance concepts," in *Third International Conference on Evolutionary MultiCriterion Optimization, EMO 2005*. SpringerVerlag, 2005, pp. 459–473.

[16] X. Hu and R. Eberhart, "Multiobjective optimization using dynamic neighborhood particle swarm optimization," in *CEC '02: Proceedings of the Evolutionary Computation on 2002. CEC '02. Proceedings of the 2002 Congress*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 1677–1681.

[17] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method in multiobjective problems," in *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*. New York, NY, USA: ACM, 2002, pp. 603–607.

[18] H. Yapicioglu, A. E. Smith, and G. Dozier, "Solving the semi-desirable facility location problem using bi-objective particle swarm," pp. 733–749, March 2007. [Online]. Available: http://ideas.repec.org/a/eee/ejores/v177y2007i2p733-749.html

[19] J. Brimberg and H. Juel, "A bicriteria model for locating a semi-desirable facility in the plane," pp. 144–151, April 1998. [Online]. Available: http://ideas.repec.org/a/eee/ejores/v106y1998i1p144-151.html

A New Sequential VB-VB Approach for Economic Dispatch of Thermal Generation Units with Convex/Nonconvex Cost Characteristics

[1]Azzam ul Asar, Tahir Nadeem Malik[2], Intisar Ali Sajjad[2]
[1]Dept. of Electrical Engineering, CECOS University, Peshawar, Pakistan.
[2]Dept. of Electrical Engineering, University of Engineering & Technology, Taxila, Pakistan.

Abstract - *Economic dispatch (ED) is the generation allocation problem subjected to various constraints. Actual behaviour of thermal generators is nonconvex but cost characteristics are approximated to convex ones for analysis. So the results obtained through optimization techniques are inaccurate. Convex ED is solved using mathematical approaches however dynamic programming has been used to solve nonconvex ED but it suffers from curse of dimensionality. Nonconvex ED problem is generally solved by using AI tools and hybrid approaches. Virtual Bees (VB) algorithm has been used for the solution of complex optimization problems. This paper presents a New Sequential VB-VB algorithm to solve convex as well nonconvex ED problem. Standard IEEE test systems have been tested with convex and nonconvex ED problems for the validation of this approach. Numerical results obtained indicate the validity of proposed algorithm in comparison with other state of the art techniques.*

Keywords: Virtual Bees Algorithm, Economic Dispatch, Nonconvex Cost Function, Local Search, Global Solution, Valve–Point Effect.

1 Introduction

Economic dispatch (ED) problem is considered essential and a critical step in power system operation. If the supply of the load is total and in a most economic way, in addition to being exposed to the inequality and equality constraints, then it is the generation allocation problem and is defined as the process of calculating the generation of the generating units [2]. In power system operation, generators are represented by input–output curves. The precise economic dispatch depends mainly upon the correct representation of these characteristic curves. The characteristics for input–output are physically nonlinear and nonconvex due to valve-point effect, multiple fuels and operational constraints such as prohibited operating zones. Economic dispatch problem is generally classified as convex and nonconvex. In convex economic dispatch input–output characteristics are taken as piecewise linear and of monotonically increasing nature and the optimization algorithms based on mathematical programming e.g. λ iteration, base point participation method, as well as Gradient and Newton's methods [3] are used. The nonconvex ED problem represents the absolute,

pragmatic behaviour of generating units having non-smooth (discontinuous, non-differentiable and nonlinear) characteristic. This kind of problem requires a swift, precise and robust solution methodology.

A large number of optimization techniques are presented in the literature for the solution of power ED [4] problem. Dynamic programming has been used but it has dimensionality problem [5]. Generally, learning based heuristic search methods are employed as tools for the solution of composite optimization problems because of their ability to overcome the weaknesses of the established optimization methods [18]. The virtual Bees algorithm (VBA) is a potential heuristic tool for ED problems [6-7]. Bees algorithm is a population based optimization technique which is being inspired by the foraging behaviour of honey bees. This algorithm has the ability to reach the global minimum search space in a short time, but then takes longer time to converge due to randomness. One of the major limitations of this technique is that the bees can't learn from errors and leading to repeatedly foraging a wrong space [15-17]. Proposed new approach is one of the methodologies to get around this problem. This approach based on sequential combination of VB Algorithm with itself and has been used to solve convex and nonconvex problem. Organization of this paper is as follows. Section 2 presents the economic dispatch problem formulation for both convex and non convex problems. Section 3 covers the VBA based economic dispatch and proposed new Sequential VB-VB approach. Section 4 presents the results obtained and finally section 5 gives the concluding remarks.

2 ED Problem Formulations

2.1 Convex Economic Dispatch Problem

Convex ED is actually the approximation of generators input/output characteristics.

$$\text{Minimize:} \quad F(P) = \sum_{i=1}^{n} (a_i + b_i P_i + c_i P_i^2)$$

$$\text{Subject to :} \quad \sum_{i=1}^{n} P_i - P_{load} - P_{loss} = 0$$

$$P_{\min} \leq P_i \leq P_{\max}$$

2.2 Non-Convex ED Problem

Present day large power generating units using multi-valves steam turbines show a noticeable variation in the input–output characteristic functions, thus non-convexity appears in the characteristic curves. The major nonconvex ED problems [8] are ED with piecewise quadratic cost function (EDPQ) and ED with prohibited operating zones (EDPO). The former is nonconvex due to valve-point effect and multiple fuel mix. In general, the valve-point show ripples [9] when every steam valve begins to open. The ED cost objective function considering the valve-point effects, is generally described as superposition of sinusoidal function and quadratic function mathematically, this non- smooth curve is represented as [10]:

$$\text{Minimize:} \quad F(P) = \sum_{i=1}^{n} (a_i + b_i P_i + c_i P_i^2 + |e_i \sin(f_i(P_{i(\min)} - P_i))|)$$

$$\text{Subject to :} \qquad \sum_{i=1}^{n} P_i - P_{load} - P_{loss} = 0$$

$$P_{\min} \leq P_i \leq P_{\max}$$

3 Economic Dispatch Using VB Algorithm
3.1 Honey Bees Foraging

Honey bees foraging behaviour is one of the best examples of large number of different social insects behavioural pattern to perform a variety of complex tasks. This practice is highly organized [19]. Virtual bees algorithm was proposed by Karaboga that simulates the foraging behaviour of honey bees. Virtual bees colony consists of three groups of bees which are employed bees, onlooker bees and scout bees. Onlooker bees and scout bees are also called the unemployed bees. Half of the total number of bees is called the employed bees and other half is called the onlooker bees. Employed bees whose food source is pooped become scout bees [18].

Employed bees find the food source and on returning to hive perform a typical dance called the waggle dance to communicate the information about the location and potential of food source to the other bees (onlooker bees). Onlooker bees follow the information passed by employed bees and travel to get the food. The number of onlooker bees at a specified food source depends on the quantity of food. This process is repeated until the all food is collected [15].

3.2 VB Algorithm Parameters for Economic Dispatch

The elements of the VB Algorithm are stated briefly as follows:

Position
It is an individual solution called virtual bee represented by a k-dimensional real-valued vector, where k is the number of

optimized parameters. For any generated power combination, the ith bee can be described as:

$$S_i = \left[P_i^1, P_i^2, P_i^3, \ldots\ldots P_i^k \right]$$

Where P_i^j is the value of jth generator output in k no. of generators search space.

No of Virtual Bees (n)
It is the size of virtual bees population which is the function of generated power vector P i.e.,

$$S(P) = \left[S_1(P), S_2(P), \ldots\ldots S_n(P) \right]$$

Where $S_i(P)$ is the position vector of ith virtual bee among n virtual bees

Time Steps
It is the number of iterations for which the objective function is to be optimized or the objective function reaches its optimized value which is the stopping criteria.

Randomness Amplitude of Bees (α)
It is the constant value specified for the randomness in order to avoid the convergence at local minima used in population updating i.e.

$$S_{rand} = \alpha \times (Rand(i) - 0.5)$$

Speed of Convergence (β)
It is also a constant value specified for the fast convergence towards the best position among all virtual bees of the population.

Best Position
It is the position of virtual bee for which the objective function value is minimum among all others virtual bees of the population.
Initial location of the patches is defined as
$$T_1 = \lfloor P_{ij} \rfloor$$
Where
   i: the generation unit
   j: the bee
$$P_{ij} = \text{Random value} \times \text{Range}_i + P_{i\,\min}$$
Range is defines as
$$\text{Range}_i = P_{i\,\max} - P_{i\,\min}$$
Initially random value is generated and range is defined using initial value of generation for ith unit is generated. Each bee gives generation value for each generator.
Initial value generated for each unit is compared with the upper and lower limits. If the initial value generated is exceeding the upper limit of generation unit, then it is fixed to its maximum limit value and on other hand for value below the lower limit is fixed to lower limit. Values are stored in the array as:

$$T_1 = \begin{bmatrix} P_{11} & P_{12} & P_{13} & .... \\ P_{21} & P_{22} & P_{23} & .... \\ P_{31} & P_{32} & P_{33} & .... \\ . & .. & .. & .... \end{bmatrix}$$

P_{23} is value generated for unit No. 2 by the 3rd bee. The matrix is transposed and stored

The value given in above equation is generated by the scout bees. This data is given to objective function and cost for each unit for each bee is calculated. Then total cost is calculated for all units for each bee.

Generation values used for total cost calculation is checked for load demand. It is required to generation values to meet the load demand. Constraint is checked by

$$\text{Constraint} = P_d - P_{gi}$$

Minimum value of constrain is selected and corresponding value of generation is stored. Here the scout bees has completed the initial patch evaluation and found most suitable solution.

For moving bees, the initial best patch is available. The moving bees target the best solution and search in the neighbourhood of the solution using

$T_{1new} = [T_{1old} \times (1\text{-beta})] +$
$[T_{1old} \times beta] + [alpha\ (random\ value - 0.5)]$

New set of generation values for each unit is generated. If the value exceeds the limits of unit, then it is fixed to boundary value. Objective function is calculated and constraint of load demand is implemented. For minimum constraint value the minimum cost is selected and best generation is stored.

3.3 New Sequential VB-VB Optimization Algorithm

Despite of the strengths of virtual bees algorithm it has the drawback of not learning ability of artificial bees from errors and resulting into foraging the wrong space repeatedly. VBA is hybridized with itself by overcoming the weakness mentioned above. (Detail)

In proposed algorithm one of the weaknesses of basic VBA is addressed and remedial measures are taken to overcome this drawback. Drawback of basic VBA is that it can't learn from errors and consequently may forging a wrong space at each run of algorithm.

Elite population selection method is used new algorithm to get better results.

At first step, algorithm is called for specific number of runs and at each run a population with best fitness value is selected. These best fittest solutions are used for second step of algorithm. Best selection phase can be expressed mathematically as:

$$best\_cost = cost(best\_converge)$$
$$best\_gen = gen(best\_converge)$$

Elite generations selected in previous phase are used for further optimization process. Employed bees phase is bypassed and on the basis of results obtained using elite population random selection is used in on-looker bees phase. Mathematically this phase can be represented as:

$$Initial\ location = best\_gen(random([1,n]), i)$$

Where *n* is the total no. of elite generations.
Pseudo code for new approach is as follows:

1. *Initialization of parameters and population*
2. *Fitness Evaluation*
3. *Repeat initial VB phase until no. of runs*
   3.1.1   *Onlooker Bees Phase*
   3.1.2   *Scout Bees Phase*
   3.1.3   *Elite Selection*
4. *Repeat Modified VB phase until no. of runs*
   4.1.1   *Random Selection from Elite Population*
   4.1.2   *Onlooker Bees Phase*
   4.1.3   *Scout Bees Phase*
5. *Print Results*
6. *Terminate Program*

This pseudo code is represented using flow chart in Fig. 1
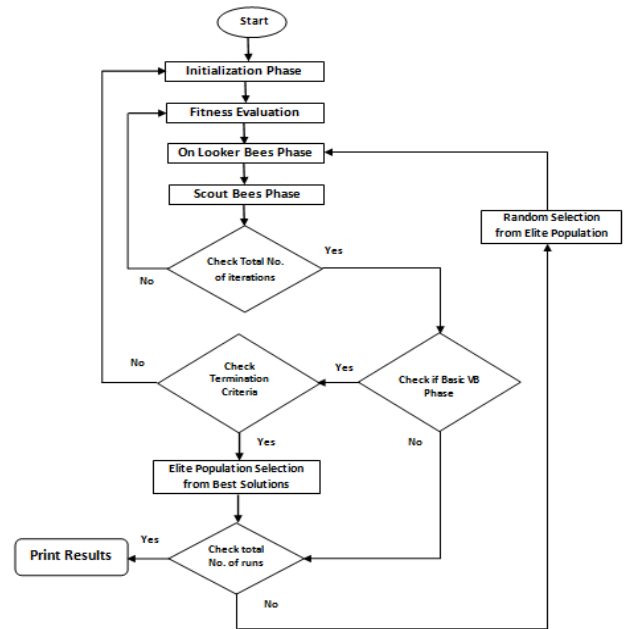

Fig. 1: Flow Chart for New Sequential VB-VB Approach

4   Case Studies and Test Results

To validate the feasibility of new approach it has been applied on different convex as well as nonconvex systems and made comparison with different methods. This algorithm is implemented on MATLAB environment. Following parameter ranges are used for evolutionary model:

Table 1: Evolutionary Model for Case Studies

Algorithm Parameter	Symbol	Range
Population Size	N	40 – 100
Randomness Amplitude	α	1 – 5
Convergence Rate	β	0.1
Probability of Selection	POS	0.5
No. of Iterations	Iter.	500 – 10000
No. of Runs	runs	30

The test systems used in case studies are given in Table 2. Statistical results of different case studies are summarized from Table 3 to Table 8. In case study 1, 2, & 3 generating units are used with convex cost characteristics and load demands are 150 and 850 MW respectively. Case study 3 describes the convex ED problem with 15 generating stations and load demand of 2630 MW. In case study 4, 20 generating units with convex characteristics are taken. Results are shown in Table 5. Case study 5 is related to non-convex ED with 3 generators and load demand of 210 MW. Similarly case study 6 and 7 are summarized in Table 7 and 8 with nonconvex cost characteristics of 6 and 13 generating units and load demand of 283.4 and 1800 MW respectively.

Table 2: Results of Case Study 1

Description	Virtual Bees Algorithm		VB-VB Sequential Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	64.3684	32.6366	58.5284	39.292
P_2	17.3168	49.0421	27.4907	41.561
P_3	68.3295	68.359	63.9417	69.137
$\sum P_i$ (MW)	150.0147	150.0376	149.9607	149.99
Cost ($/h)	1605.9196	1562.4588	1595.1842	1530.208
Tolerance	0.014672	0.0376	0.039289	0.0099957
Iterations	23	4	16	1
Std. Deviation	8.9534		47.5238	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	1574.143 [14]		1530.208	

Table 3: Results of Case Study 2

Description	Virtual Bees Algorithm		Sequential VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	550.9283	385.81	461.1266	487.61
P_2	103.13	264.7555	212.4435	167.57
P_3	195.9295	199.4382	176.4386	194.83
$\sum P_i$ (MW)	849.9877	850.0037	850.0087	850.0033
Cost ($/h)	8342.6904	8085.4104	8835.3167	7951.5681
Tolerance	0.012265	0.0037081	0.0087465	0.0033005
Iterations	2902	1	514	01
Std. Deviation	68.9761		306.4675	
Comparison	Literature – Min.		Seq. VB-VB– Min.	
	8125.734 [13]		7951.5681	

Table 4: Results of Case Study 3

Description	Virtual Bees Algorithm		Seq. VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	453.2169	454.666	289.0821	451.1299
P_2	308.207	313.9854	309.9596	223.3429
P_3	123.6886	128.4887	114.3607	128.5608
P4	127.8605	129.5871	125.0488	129.1468
P5	276.7261	308.1445	344.8359	227.4632
P6	272.288	234.0246	302.1475	321.9228
P7	242.8115	275.539	285.3998	214.1664
P8	162.1758	115.1478	253.6758	299.0171
P9	160.9586	161.7624	142.7643	160.7859
P10	157.6558	158.209	155.7469	158.4664
P11	77.7445	78.6368	73.3211	66.9951
P12	76.3151	79.4656	69.3541	77.78
P13	83.1461	83.7589	72.2647	76.6119
P14	53.8078	54.5998	47.8815	47.1779
P15	53.3534	53.9838	44.1172	47.4326
$\sum P_i$ (MW)	2629.9556	2629.9995	2629.96	2629.9995
Cost ($/h)	32951.4235	30877.899	32993.5572	30602.863
Tolerance	0.044421	0.00047381	0.040039	0.00046487
Iterations	25	25	9828	01
Std. Dev.	541.2254		642.5501	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	32716.87 [14]		30602.863	

Table 5: Results of Case Study 4

Description	VBA		Seq. VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	160.5545	115.38	231.7882	197.5873
P_2	56.5737	83.819	67.8867	67.7121
P_3	178.6741	107.04	110.1198	0.9591
P4	128.8883	113.45	164.011	88.5922
P5	148.7699	107.96	156.2681	136.2317
P6	95.8305	103.28	99.2597	94.7308
P7	52.2137	99.581	36.7768	96.6909
P8	54.7607	97.155	54.9384	70.6483
P9	53.2142	92.374	53.7638	87.7694
P10	112.0122	108.99	91.8574	105.8521
P11	297.9698	104.03	285.8229	297.6054
P12	202.0193	106.03	235.7272	270.0051
P13	158.353	104.25	155.8093	153.4257
P14	128.4283	119.24	120.7035	127.9532
P15	184.2066	110.52	182.9922	169.8062
P16	76.7613	110.71	76.713	78.8253
P17	81.5656	102.37	69.5417	78.9729
P18	117.7405	93.195	113.934	105.5725
P19	115.3835	116.59	92.1607	113.9127
P20	96.0378	97.363	100	77.1476
$\sum P_i$ (MW)	2499.9577	2500.0028	2500.0744	2500.0007
Cost ($/h)	61443.1052	60939.693	61157.1506	58038.363
Tolerance	0.04229	0.0028387	0.074428	0.00072934
Iterations	60	130	1923	1
Std. Deviation	413.0115		876.8359	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	62450.797105 [11]		58038.363	

Table 6: Results of Case Study 5

Description	Virtual Bees Algorithm		Seq. VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	52.0353	52.0995	55.8938	52.817
P_2	40.1013	38.4412	53.688	40.887
P_3	117.7784	119.4839	100.4149	116.28
$\sum P_i$ (MW)	209.915	210.0246	209.9966	209.9874
Cost ($/h)	3198.3797	3167.8814	3210.4691	3145.9095
Tolerance	0.084986	0.024574	0.0033571	.012644
Iterations	730	361	01	01
Std. Dev.	7.4368		24.8993	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	3205.991 [1]		3145.9095	

Table 7: Results of Case Study 6

Description	Virtual Bees Algorithm		Sequential VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	113.2507	137.88	119.7678	148.85
P_2	77.0222	69.043	74.2749	33.005
P_3	30.6264	24.35	19.7065	41.136
P4	28.753	20.871	33.2203	16.273
P5	18.1041	12.522	18.8648	23.45
P6	15.9523	18.731	17.2941	20.687
$\sum P_i$ (MW)	283.7087	283.3959	283.1283	283.4032
Cost ($/h)	1023.087	975.0494	1018.92	931.0837
Tolerance	0.30867	0.004127	0.27167	0.0031623
Iterations	3930	9928	7525	6308
Std. Deviation	27.8546		30.6952	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	984.9365 [1]		931.0837	

Table 8: Results of Case Study 7

Description	Virtual Bees Algorithm		Sequential VB-VB Approach	
	Max. Cost	Min. Cost	Max. Cost	Min. Cost
P_1	318.967	283.32	120.929	104.7929
P_2	128.5898	162.86	213.5013	102.9776
P_3	229.9939	158.55	188.3687	179.6633
P4	115.0251	163.52	147.4577	176.3996
P5	128.8602	88.905	141.6271	157.4011
P6	92.8207	89.65	103.5175	165.1323
P7	83.3552	128.38	90.6097	108.7695
P8	166.0214	176.65	173.5064	173.1821
P9	165.9512	172.44	178.8677	176.1735
P10	110.6996	113.15	109.1779	115.2834
P11	107.3016	79.857	107.1645	114.7839
P12	69.272	80.054	106.8693	114.512
P13	83.6122	102.67	118.4017	110.9121
$\sum P_i$ (MW)	1800.4699	1800.0178	1799.9985	1799.984
Cost ($/h)	19113.212	18720.2009	19565.8036	17917.672
Tolerance	0.46994	0.017762	0.0015301	0.01596
Iterations	4063	2265	173	1
Std. Dev.	134.1676		445.8575	
Comparison	Literature – Min.		Seq. VB-VB – Min.	
	17944.9521 [12]		17917.6726	

Discussion:

Results obtained from different case studies for convex economic dispatch are summarized in Table 9, 10 and Figure 2 to 5. Keeping in view the above results following observations can be made:

In all test cases the total production cost is low as compare to conventional VBA and latest research results available.

Experimental results on convex and nonconvex functions in different dimensions show that the proposed algorithm has remarkable robustness.

Table 9 and figure 2, 4 shows the behaviour of algorithm for the solution of convex optimization problem for economic dispatch. It can also be seen that behaviour is quadratic and have improve results for increase in system size.

Table 10 and figure 3, 5 shows that non convex problems can also be solved using this new technique and this strategy is verified upto 13 machine system.

Table 9: Comparison of Seq. VB-VB Results with Literature Min. For Convex ED

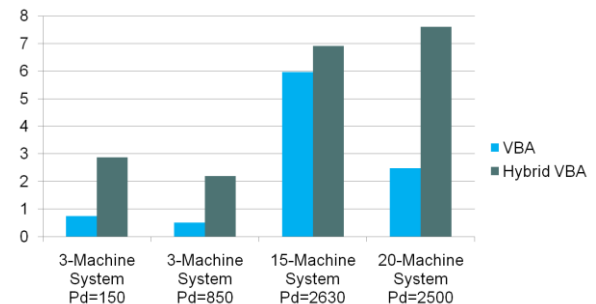| Description	PD MW	Seq. VB-VB Min. Cost	Literature Min. Cost	Seq. VB-VB Percent Reduction
3-Machine (Convex)	150	1530.208	1574.143 [14]	2.87
3-Machine (Convex)	850	7951.5681	8125.734 [13]	2.19
15-Machine (Convex)	2630	30602.863	32716.87 [14]	6.91
20-Machine (Convex)	2500	58038.363	62489.5 [11]	7.60

Fig. 2: Comparison of % reduction in Cost for Convex ED

Table 10: Comparison of Seq. VB-VB Results with Literature Min. For Non-Convex ED

Description	PD (MW)	Seq. VB-VB Min. Cost	Literature Min. Cost	Seq. VB-VB %Reduction
3-Machine (Non Convex)	210	3145.9095	3205.991[1]	**1.91**
6-Machine (Non Convex)	283.4	931.0837	984.9365 [1]	**5.78**
13-Machine (Non Convex)	1800	17917.6726	17944.9521 [12]	**0.15**
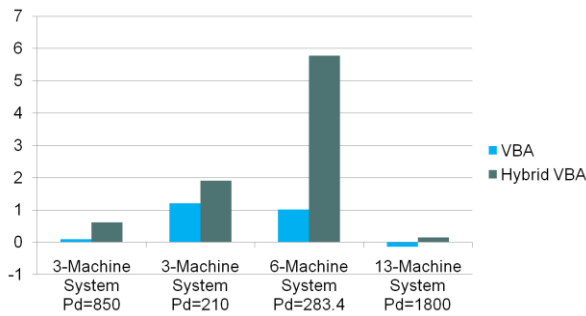
Fig. 3: Comparison of % reduction in Cost for Non-Convex ED
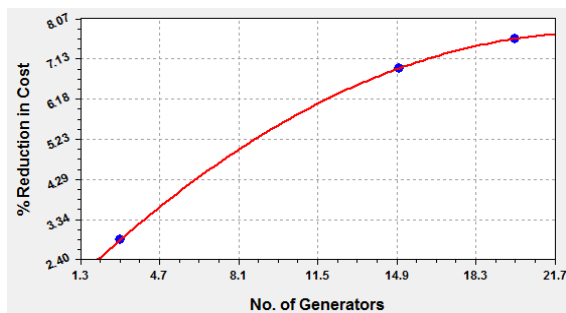
Fig. 4: Sequential VB-VB Trend on % Cost Reduction for Convex ED

Quadratic Fit: $y=a+bx+cx^2$

Where

a = 1.3341176,   b = 0.54701961   c = -0.011686275
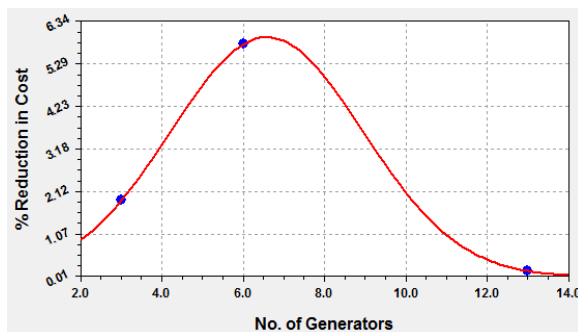
Fig. 5: Sequential VB-VB Trend on % Cost Reduction for Non Convex ED

Gaussian Model: $y=a*exp((-(b-x)^2)/(2*c^2))$

Where

a =5.9508442,   b =6.5718588,   c =2.3692341

5 Conclusions

In this paper a new VB-VB sequential approach is presented for the solution of convex as well as non convex economic dispatch problem. For non convexity of thermal machines valve point effect is considered. VB algorithm is potential heuristic tool for the solution of optimization problems. This algorithm has ability to reach near the global minimum point in a short interval of time but then take somewhat longer time to converge. In addition to this VB has disadvantage that it has no learning ability from errors. New approach is developed to overcome its weakness of learning ability by using the elite selection mechanism. The New VB-VB Sequential Algorithm has been applied on both convex and non convex economic dispatch problems taking valve point effect as main source of non convexity. The results obtained through this new approach show the promise of this technique. Results confirm that this approach is capable to solve both convex and non convex economic dispatch problems with more optimized results.

References

[1] Tahir Nadeem Malik, Azzam ul Asar, Mudasser F. Wyne, Shakil Akhtar, "A New Hybrid Approach for the Solution of Non Convex Economic Dispatch Problem with Valve Point Effects", Electric Power System Research, Elsevier, volume 80, issue 9, April, 2010.

[2] H.H. Happ, "Optimal power dispatch – a comprehensive survey", IEEE Transactions on Power Apparatus and Systems PAS-96 (20 (May/June)) (1977) 3841–3854.

[3] A.J. Wood, B.F. Wollenberg, Power Generation Operation and Control, John Wiley, New York, 1996.

[4] B.H. Chowdhury, S. Rahman, "A review of recent advances in economic dispatch", IEEE Transactions on Power Systems 5 (4) (November 1991) 1248–1259.

[5] Z.X. Liang, J.D. Glover, "A zoom feature for a dynamic programming solution to economic dispatch including transmission losses", IEEE Transactions on Power Systems 7 (2) (1992) 544–550.

[6] G.A. Bakare, U.O. Aliyu, G.K. Venayagamoorthy, Y.K. Shu'aibu, "Genetic algorithms based economic dispatch with application to coordination of Nigerian thermal power plants", IEEE Power Engineering Society, General Meeting (2005) 2886–2891.

[7] C. Chiang, "Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels", IEEE Transactions on Power Systems 20 (4) (2005) 1690–1699.

[8] W.M. Ling, F.S. Cheng, M.T. Tsay, "Nonconvex economic dispatch by integrated artificial intelligence",

IEEE Transactions on Power Systems 16 (2) (2001) 307–311.

[9] J. Park, K. Lee, J. Shin, K.Y. Lee, "A particle swarm optimization for economic dispatch with nonsmooth cost functions", IEEE Transactions on Power Systems 20 (1) (2005) 34–42.

[10] D.C. Walters, G.B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading", IEEE Transactions on Power Systems 8 (3) (1993) 1325–1332.

[11] Mekhamer, S.F.; Abdelaziz, A.Y.; Badr, M.A.L.; Kamh, M.Z." Enhancing the performance of hopfield neural network applied to the Economic Dispatch Problem", IEEE Power Systems Conferenec, MEPCON, 2006.

[12] Nagarjuna Duvvuru, K. S. Swarup, "A Hybrid Interior Point Assisted Differential Evolution Algorithm for Economic", IEEE Transaction on Power system, 2010.

[13] M. F. AlHajri and M. E. El-Hawary, "Pattern Search Optimization Applied to Convex and Non-Convex Economic Dispatch" IEEE International Conference on Systems, Man and Cybernetics, 2007.

[14] Tiacharoen, S.; Potiya, S.; Polratanasuk, P. " Solving Various Types of Economic Dispatch Problems Using Bees Algorithm", International Conference on Electrical Engineering/Electronics Computer Telecommunication and IT (ECTI-CON), May 2010.

[15] D. Karaboga, "An idea based on bee swarm for numerical optimization," Tech. Rep. TR-06, Erciyes University , Engineering Faculty, Computer Engineering Department, 2005.

[16] D. Karaboga and B. Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," J. Global Optim, Vol. 39, pp. 459–471, 2007.

[17] D. Karaboga and B. Basturk. "On the performance of artificial bee colony (ABC) algorithm," Appl. Soft Comput., Vol. 8,2008, pp. 687–697.

[18] Nguyen Tung Linh, Nguyen Quynh Anh, "Application of Artificial Bee Colony Algorithm for reconfiguring Distribution Network", International Conference on Computer Modeling and Simulation (ICCMS), 2010, Vol. 1, pp. 102-106.

[19] Dusan Teodorovic, Panta Lucic, Goran Markovic, Mauro Dell' Orco, "Bee Colony Optimization: Principles and Applications", IEEE 8th Seminar on Neural Network Applications in Electrical Engineering. NEUREL, Serbia, Sep. 2006, pp. 151-156.

SESSION

KNOWLEDGE AND INFORMATION REPRESENTATION, PROCESSING, ENGINEERING, ACQUISITION METHODS AND APPLICATIONS

Chair(s)

TBA

Syllogistic Reasoning for Cardinal Direction Relations

Ah Lian Kor[1] ,and Brandon Bennett[2]
[1]School of Computing, Creative Technologies, and Engineering,
Faculty of Arts, Environment and Technology, Leeds Metropolitan University,
Headingley Campus, LS6 3QS, Leeds, UK.
[2]School of Computing, University of Leeds, Woodhouse Lane, LS2 9JT, Leeds, UK.
A.Kor@leedsmet.ac.uk, Brandon@comp.leeds.ac.uk

Abstract - *Syllogistics is a type of logical reasoning which involves quantifiers such as All, and Some. Here, we explore the use of syllogisms to formalize quantified direction relations by incorporated them into the classical Projection-based Model for cardinal directions [Frank,1992], a two-dimensional Euclidean space relative to an arbitrary single-piece region, a, is partitioned into the following nine tiles: North-West, nw(a); North, n(a); North-East, ne(a); West, w(a); Neutral Zone, o(a); East, e(a); South-West, sw(a); South, s(a); and South-East, se(a). Typically, only these tiles are employed for reasoning about cardinal direction relations [Ligozat,1988; Goyal and Egenhofer, 2000;.Skiadopoulos and Koubarakis, 2001,2004]. Chen et. al (2007) have integrated cardinal direction relations and RCC-5 and RCC-8 to represent directional and topological information. They have investigated the mutual dependencies them, and discussed the composition of the hybrid relations. However, in this paper, we have developed formalisms to facilitate an expressive reasoning mechanism for spatial databases. These formalisms is the outcome of the integration of RCC-5 [Cohn, et.al,1997] for topological relations, the Project-based Model for cardinal direction relations and syllogisms. In the latter part of the paper, we demonstrate how to compute the composition of such hybridized cardinal direction relations.*

Keywords: Syllogistic reasoning, cardinal direction, composition table, formal reasoning, quantifier, knowledge representation

1.Introduction

According to Egenhofer and Mark (1995), Naïve Geography encompasses formal but yet commonsense models which form the basis for the design of Geographic Information Systems. They view qualitative spatial and temporal reasoning as the crux of Naïve Geography because it deals with partial information and helps simplify spatial queries in GIS [ibid; Frank, 1992] that is prevalent in spatial applications. Little work has been done on the formalization and efficient processing of qualitative reasoning for cardinal directions [Theorodis, et. al. 1998] which describe relative positions of regions in large-scale spaces. These relations specify the direction from one region to another in terms of the familiar compass bearings: north, south, east and west. The intermediate directions north-west, north-east, etc. are also often used. The models for reasoning with cardinal directions are the cone-shaped, projection-based models [Frank, 1992], and relative orientation model [Freksa, 1992]. Isli [2004] has developed an expressive model by combining these two models.

Papadias and Theorodis [1997] describe topological and direction relations between regions using their minimum bounding rectangles (MBRs). However, the language used is not expressive enough to describe

direction relations. Additionally, the MBR technique yields erroneous outcome when involving regions that are not rectangular in shape [Goyal and Egenhofer, 2000]. Some work has been done on hybrid direction models. Qualitative orientation has been integrated with distance to obtain positional information [Escrig et.al, 1998; Clementini et.al,1997; Moratz et. al, 2012]. Gerevini and Renz [2002] combined topological and metric size information for spatial reasoning. Sharma and Flewelling [1995] infer spatial relations from integrated topological and cardinal direction relations and Sharma et. al [1994] has developed a prototype known as Qualitative Spatial Reasoner which integrates the two and also approximate distances. Kor and Bennett [2003, 2020, in press] have developed expressive models cardinal direction relations. Liu et. al [2009] analyzed the interaction between topological and directional constraints for extended spatial regions. They have shown that the JSP over basic RCC-8 and basic Rectangle Algebra (RA and note that it is a two-dimensional counterpart of Interval Algebra) networks can be solved in polynomial time while the JSP over basic RCC-8 and basic Cardinal Direction Calculus (CDC) networks is NP-Complete. On the other hand, Li [2006, 2007] combined RCC-5 and RCC-8 with DIR9 (employed to express directional information and is a subclass of RA) respectively. He discussed the Constraints Satisfaction Problem (CSP) and also the reasoning complexity. Chen, et. al [2007] also investigated the mutual dependencies between basic relations of RCC-5 and cardinal direction relations, enhanced the constraint propagation algorithm to enforce path consistency and examining the consistency of the hybrid constraints.

Quantifiers have been applied to temporal reasoning [Pratt and Francez, 1998] but not cardinal directions. Thus, our novel contribution is the integration of topological binary relations in RCC-5, direction relations and syllogisms. Syllogism is a form of deductive reasoning (Bucciarelli & Johnson-Laird, 1999) which comprises a small set of inferences that are based on premises which conatin quantifiers (*All, Some,* and *No*). We have employed the composition technique making the inferences and also used visualization to confirm the validity of the syllogistic reasoning for cardinal directions.

2.Cardinal Direction Relations Model

In the *projection-based model* for cardinal directions, a two-dimensional Euclidean space of an arbitrary single-piece region, *a*, is partitioned into nine *tiles* as shown in Figure 1. They are North-West, nw(*a*); North, n(*a*);

North-East, ne(a); West, w(a); Neutral Zone, o(a); East, e(a); South-West, sw(a); South, s(a); and South-East, se(a).
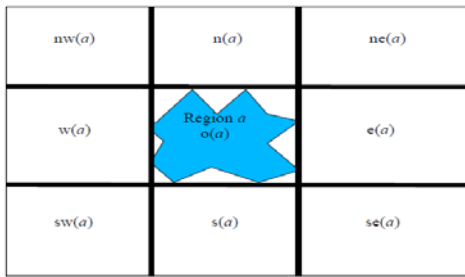

Figure 1. A *Projection-based* Cardinal Direction Model

3. Syllogistic Reasoning

Syllogistic reasoning syllogistic reasoning is viewed as a form of deductive reasoning which involves reasoning with a pair of premises containing quantifiers All, and Some (Bucciarelli and Johnson-Laird, 1999; Johnson-Laird, 1983). Syllogisms were first analysed by Aristotle and the four types of premises are: All x is y (a universal and affirmative assertion); Some x is y (a particular and affirmative assertion); No x is y (a universal and negative assertion); and Some x is not y (a particular and negative assertion). We shall employ the quantifiers All, and Some to extend existing spatial language for expressing direction relations. Johnson-Laird [1983] used Euler Circles to represent the 4 types of assertions diagrammatically (Table 1).
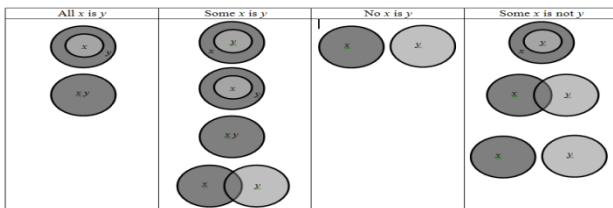
Table 1. Diagrammatic Representation for Assertions with the Quantifiers *All*, and *Some*

4. Formal Representation for topological relations

We shall first begin with the parthood relation, p(x,y) which means x is a part of y. It is the primitive for 'x overlaps y' and most of the RCC-5 [Cohn et.al, 1997] JEPD binary topological relations for regions. A diagrammatic representation of the RCC-5 topological relations is shown in Table 2.

D1. ovr(x,y) $\equiv_{def}$ $\exists z$ [p(z,x) $\wedge$ p(z,y)]
D2. eq(x, y) $\equiv_{def}$ [x=y]
D3. pp(x, y) $\equiv_{def}$ [p(x,y) $\wedge$ ¬eq(x,y)]
D4. ppi(x, y) $\equiv_{def}$ [p(y,x) $\wedge$ ¬eq(x,y)]
D5. po(x, y) $\equiv_{def}$ [ovr(x,y) $\wedge$ ¬p(x,y) $\wedge$ ¬p(y,x)]
D6. dr(x,y) $\equiv_{def}$ [¬ovr(x,y)]
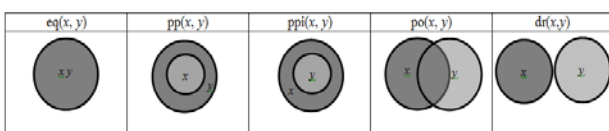
Table 2. Diagrammatic Representation for RCC-5 relations

5. Syllogistic Reasoning Model for Direction Relations

In this paper, we shall demonstrate how the premises shown in Table 1 could be integrated with RCC-5 relations in Table 2 to facilitate reasoning about direction relations.

1. All *b* is in n(a) is represented by:
$$\forall[\text{in}(b, \text{n}(a))] \equiv \text{pp}(b,\text{n}(a)) \vee \text{eq}(b,\text{n}(a))$$
Several possible graphical representations are in Figure 2.

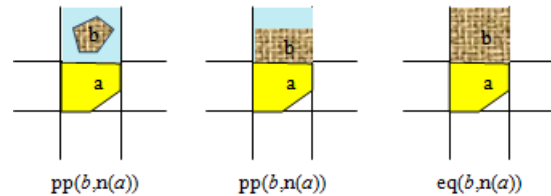pp(b,n(a))   pp(b,n(a))   eq(b,n(a))

Figure 2. Several Graphical Representations for 'All b is in n(a)'
Assume that a region is represented by x and any *tile* of the reference region is y.

All x is in y is represented by: $\forall[\text{in}(x,y)] \equiv \text{pp}(x,y) \vee \text{eq}(x,y)$
2. Some *b* is in n(a) is represented by:
$\exists[\text{in}(b, \text{n}(a))] \equiv \text{eq}(b,\text{n}(a)) \vee \text{pp}(b,\text{n}(a)) \vee \text{po}(b,\text{n}(a)) \vee \text{ppi}(b,\text{n}(a))$
The following graphical representation is not exhaustive.

pp(b,n(a))   pp(b,n(a))   eq(b,n(a))   po(b,n(a))   po(b,n(a))   ppi((b,n(a))
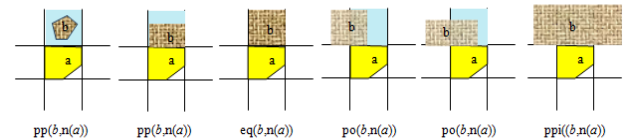
Figure 3. Graphical Representation for 'Some *b* is in n(a)'
Assume that a region is represented by x and any *tile* of the reference region is y.

Some x is in y is represented by:
$$\exists[\text{in}(x,y)] \equiv \text{pp}(x,y) \vee \text{eq}(x,y) \vee \text{po}(x,y) \vee \text{ppi}(x,y)$$
3. No *b* is in n(a) can be represented as:
$\forall[\neg\text{in}(b, \text{n}(a))] \equiv \text{dr}(b,\text{n}(a))$ or $\neg\exists[\text{in}(b, \text{n}(a))] \equiv \text{dr}(b,\text{n}(a))$
The non-exhaustive graphical representation is as below:
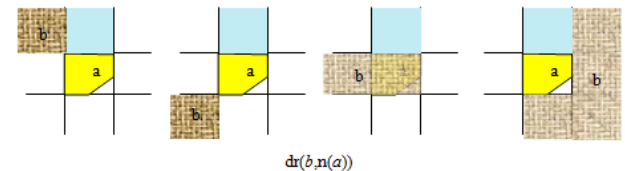

dr(b,n(a))

Figure 4. Graphical Representation for 'No *b* is in n(a)'
No x is in y is represented by: $\forall[\neg\text{in}(x,y)] \equiv \text{dr}(x,y)$ or $\neg\exists[\text{in}(x,y)] \equiv \text{dr}(x,y)$
4. Some n(a) is not in *b* can be defined as:
$\exists[\neg\text{in}(b, \text{n}(a))] \equiv \text{dr}(b,\text{n}(a)) \vee \text{po}(b,\text{n}(a)) \vee \text{ppi}(b,\text{n}(a))$
The non-exhaustive graphical representation is as follows:

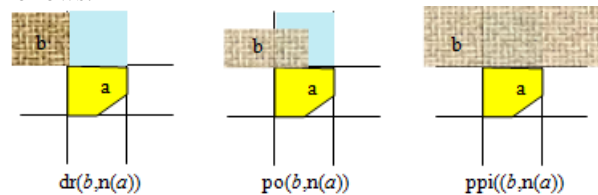dr(b,n(a))   po(b,n(a))   ppi((b,n(a))

Figure 5. Graphical Representation for 'Some n(a) is not in *b* '
Some x is not in y is represented by:
$$\exists[\neg\text{in}(x,y)] \equiv \text{dr}(x,y) \vee \text{po}(x,y) \vee \text{ppi}(x,y)$$
Assume R = {n(a), ne(a), nw(a), w(a), o(a), e(a), s(a), se(a), sw(a)}. The general definitions of the above mentioned assertions are as follows:

D7. All *b* is in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as: ∀[in(*b*, *t*)] ≡ pp(*b*,*t*) ∨ eq(*b*,*t*)

D8. Some *b* is in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as: ∃[in(*b*, *t*)] ≡ pp(*b*,*t*) ∨ eq(*b*,*t*) ∨ po(*b*,*t*) ∨ ppi(*b*,*t*)

D9. No *b* is in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as: ∀[¬in(*b*, *t*)] ≡ dr(*b*,*t*) or ¬∃[in(*b*, *t*)] ≡ dr(*b*,*t*)

D10. Some *b* is not in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as: ∃[¬in(*b*, *t*)] ≡ dr(*b*,*t*) ∨ po(*b*,*t*) ∨ ppi(*b*,*t*)

Assume U is a universal set of the RCC-5 relations between a region *b* and a tile *t* of region *a*. Thus, U={dr(*b*,*t*),pp(*b*,*t*)∨eq(*b*,*t*)∨po(*b*,*t*)∨ppi(*b*,*t*)}. Based on definitions D7 to D10 above,

i. 'Some *b*' or 'No *b*' is in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as:
∃[in(*b*,*t*)]∨ ¬∃[in(*b*,*t*)] ≡ pp(*b*,*t*) ∨ eq(*b*,*t*) ∨ po(*b*,*t*) ∨ ppi(*b*,*t*) ∨ dr(*b*,*t*) ≡ U

ii. 'Some *b* is not' or 'All *b*' is in a tile *t* where *t* ∈ R (a tile of a region *a*) is represented as:
∃[¬in(*b*, *t*)] ∨ ∀[in(*b*, *t*)] ≡ pp(*b*,*t*) ∨ eq(*b*,*t*) ∨ po(*b*,*t*) ∨ ppi(*b*,*t*) ∨ dr(*b*,*t*) ≡ U

Thus, the quantifiers 'Some *b*' is a complement of 'No *b*' and vice versa. On the other hand, 'Some *b* is not' is a complement of 'All *b*' and vice versa.

6. Composition Direction Relations Based on the Syllogistic Reasoning Model

Skiadopoulos and Koubarakis [2004] have provided a formal presentation of the composition of cardinal direction relations based on Goyal and Egenhofer's direction model [2001]. In our previous paper [Kor and Bennett, 2010], we have developed a formula to compute the composition of weak direction relations between 'whole' and 'part' extended regions. Chen et. al [2007] have discussed the composition of RCC-5 topological relations and cardinal direction relations. However, in this paper, we have integrated RCC-5, cardinal direction relations and syllogism. We shall first start by establishing the possible topological relation/s between each individual *tile* of *b* and that particular t*ile* of *a* when given a topological relation between region *b* and a *tile* of region *a*. Such relations have been tabulated in Table 3 (note the table is in three parts).

eq(b,n(a))	eq(b,ne(a))	eq(b,nw(a))	eq(b,w(a))
pp(n(b),n(a)) {po∨dr}(ne(b),n(a)) {po∨dr}(nw(b),n(a)) {po∨dr}(w(b),n(a)) pp(o(b),n(a)) {po∨dr}(e(b),n(a)) {po∨dr}(s(b),n(a)) {po∨dr}(sw(b),n(a)) {po∨dr}(se(b),n(a))	pp(n(b),ne(a)) pp(ne(b),ne(a)) {po∨dr}(nw(b),ne(a)) {po∨dr}(w(b),ne(a)) pp(o(b),ne(a)) pp(e(b),ne(a)) {po∨dr}(s(b),ne(a)) {po∨dr}(sw(b),ne(a)) {po∨dr}(se(b),ne(a))	pp(n(b),nw(a)) {po∨dr}(ne(b),nw(a)) pp(nw(b),nw(a)) pp(w(b),nw(a)) pp(o(b),nw(a)) {po∨dr}(e(b),nw(a)) {po∨dr}(s(b),nw(a)) {po∨dr}(sw(b),nw(a)) {po∨dr}(se(b),nw(a))	{po∨dr}(n(b),w(a)) {po∨dr}(ne(b),w(a)) {po∨dr}(nw(b),w(a)) pp(w(b),w(a)) pp(o(b),w(a)) {po∨dr}(e(b),w(a)) {po∨dr}(s(b),w(a)) {po∨dr}(sw(b),w(a))
eq(b,o(a))	**eq(b,e(a))**	**eq(b,s(a))**	**eq(b,se(a))**
dr(n(b),o(a)) dr(ne(b),o(a)) dr(nw(b),o(a)) dr(w(b),o(a)) eq(o(b),o(a)) dr(e(b),o(a)) dr(s(b),o(a)) dr(sw(b),o(a)) dr(se(b),o(a))	{po∨dr}(n(b),e(a)) {po∨dr}(ne(b),e(a)) {po∨dr}(nw(b),e(a)) {po∨dr}(w(b),e(a)) pp(o(b),e(a)) pp(e(b),e(a)) {po∨dr}(s(b),e(a)) {po∨dr}(sw(b),e(a))	{po∨dr}(n(b),s(a)) {po∨dr}(ne(b),s(a)) {po∨dr}(nw(b),s(a)) {po∨dr}(w(b),s(a)) pp(o(b),s(a)) {po∨dr}(e(b),s(a)) pp(s(b),s(a)) {po∨dr}(sw(b),s(a))	{po∨dr}(n(b),se(a)) {po∨dr}(ne(b),se(a)) {po∨dr}(nw(b),se(a)) {po∨dr}(w(b),se(a)) pp(o(b),se(a)) pp(e(b),se(a)) pp(s(b),se(a)) {po∨dr}(sw(b),se(a)) pp(se(b),se(a))
eq(b,sw(a))	**pp(b,n(a))**	**pp(b,ne(a))**	**pp(b,nw(a))**
{po∨dr}(n(b),sw(a)) {po∨dr}(ne(b),sw(a)) {po∨dr}(nw(b),sw(a)) pp(w(b),sw(a)) pp(o(b),sw(a)) {po∨dr}(e(b),sw(a)) pp(s(b),sw(a)) pp(sw(b),sw(a)) {po∨dr}(se(b),sw(a))	pp(n(b),n(a)) {po∨dr}(ne(b),n(a)) {po∨dr}(nw(b),n(a)) {po∨dr}(w(b),n(a)) pp(o(b),n(a)) {po∨dr}(e(b),n(a)) {po∨dr}(s(b),n(a)) {po∨dr}(sw(b),n(a)) {po∨dr}(se(b),n(a))	pp(n(b),ne(a)) pp(ne(b),ne(a)) {po∨dr}(nw(b),ne(a)) {po∨dr}(w(b),ne(a)) pp(o(b),ne(a)) pp(e(b),ne(a)) {po∨dr}(s(b),ne(a)) {po∨dr}(sw(b),ne(a)) {po∨dr}(se(b),ne(a))	pp(n(b),nw(a)) {po∨dr}(ne(b),nw(a)) pp(nw(b),nw(a)) pp(w(b),nw(a)) pp(o(b),nw(a)) {po∨dr}(e(b),nw(a)) {po∨dr}(s(b),nw(a)) {po∨dr}(sw(b),nw(a)) {po∨dr}(se(b),nw(a))
pp(b,w(a))	**pp(b,o(a))**	**pp(b,e(a))**	**pp(b,s(a))**
{po∨dr}(n(b),w(a)) {po∨dr}(ne(b),w(a)) {po∨dr}(nw(b),w(a)) pp(w(b),w(a)) pp(o(b),w(a)) {po∨dr}(e(b),w(a)) {po∨dr}(s(b),w(a)) {po∨dr}(sw(b),w(a)) {po∨dr}(se(b),w(a))	{po∨dr}(n(b),o(a)) {po∨dr}(ne(b),o(a)) {po∨dr}(nw(b),o(a)) {po∨dr}(w(b),o(a)) pp(o(b),o(a)) {po∨dr}(e(b),o(a)) {po∨dr}(s(b),o(a)) {po∨dr}(sw(b),o(a)) {po∨dr}(se(b),o(a))	{po∨dr}(n(b),e(a)) {po∨dr}(ne(b),e(a)) {po∨dr}(nw(b),e(a)) {po∨dr}(w(b),e(a)) pp(o(b),e(a)) pp(e(b),e(a)) {po∨dr}(s(b),e(a)) {po∨dr}(sw(b),e(a)) {po∨dr}(se(b),e(a))	{po∨dr}(n(b),s(a)) {po∨dr}(ne(b),s(a)) {po∨dr}(nw(b),s(a)) {po∨dr}(w(b),s(a)) pp(o(b),s(a)) {po∨dr}(e(b),s(a)) pp(s(b),s(a)) {po∨dr}(sw(b),s(a)) {po∨dr}(se(b),s(a))

pp(b,se(a))	pp(b,sw(a))	po(b,n(a))	po(b,ne(a))
{po∨dr}(n(b),se(a))	{po∨dr}(n(b),sw(a))	{pp∨po}(n(b),n(a))	{pp∨po}(n(b),ne(a))
{po∨dr}(ne(b),se(a))	{po∨dr}(ne(b),sw(a))	{po∨dr}(ne(b),n(a))	pp(ne(b),ne(a))
{po∨dr}(nw(b),se(a))	{po∨dr}(nw(b),sw(a))	{po∨dr}(nw(b),n(a))	{po∨dr}(nw(b),ne(a))
{po∨dr}(w(b),se(a))	pp(w(b),sw(a))	{po∨dr}(w(b),n(a))	{po∨dr}(w(b),ne(a))
pp(o(b),se(a))	pp(o(b),sw(a))	po(o(b),n(a))	po(o(b),ne(a))
pp(e(b),se(a))	{po∨dr}(e(b),sw(a))	{po∨dr}(e(b),n(a))	{pp∨po}(e(b),ne(a))
pp(s(b),se(a))	pp(s(b),sw(a))	{po∨dr}(s(b),n(a))	{po∨dr}(s(b),ne(a))
{po∨dr}(sw(b),se(a))	pp(sw(b),sw(a))	{po∨dr}(sw(b),n(a))	dr(sw(b),ne(a))
pp(se(b),se(a))	{po∨dr}(se(b),sw(a))	{po∨dr}(se(b),n(a))	{po∨dr}(se(b),ne(a))

po(b,nw(a))	po(b,w(a))	po(b,o(a))	po(b,e(a))
{pp∨po}(n(b),nw(a))	{po∨dr}(n(b),w(a))	{po∨dr}(n(b),o(a))	{po∨dr}(n(b),e(a))
{po∨dr}(ne(b),nw(a))	{po∨dr}(ne(b),w(a))	{po∨dr}(ne(b),o(a))	{po∨dr}(ne(b),e(a))
pp(nw(b),nw(a))	{po∨dr}(nw(b),w(a))	{po∨dr}(nw(b),o(a))	{po∨dr}(nw(b),e(a))
{pp∨po}(w(b),nw(a))	{pp∨po}(w(b),w(a))	{po∨dr}(w(b),o(a))	{po∨dr}(w(b),e(a))
po(o(b),nw(a))	po(o(b),w(a))	po(o(b),o(a))	po(o(b),e(a))
{po∨dr}(e(b),nw(a))	{po∨dr}(e(b),w(a))	{po∨dr}(e(b),o(a))	{pp∨po}(e(b),e(a))
{po∨dr}(s(b),nw(a))	{po∨dr}(s(b),w(a))	{po∨dr}(s(b),o(a))	{po∨dr}(s(b),e(a))
{po∨dr}(sw(b),nw(a))	{po∨dr}(sw(b),w(a))	{po∨dr}(sw(b),o(a))	{po∨dr}(sw(b),e(a))
dr(se(b),nw(a))	{po∨dr}(se(b),w(a))	{po∨dr}(se(b),o(a))	{po∨dr}(se(b),e(a))

po(b,s(a))	po(b,se(a))	po(b,sw(a))	dr(b,n(a))
{po∨dr}(n(b),s(a))	{po∨dr}(n(b),se(a))	{po∨dr}(n(b),sw(a))	{pp∨po∨dr∨eq∨ppi}(n(b),n(a))
{po∨dr}(ne(b),s(a))	{po∨dr}(ne(b),se(a))	dr(ne(b),sw(a))	{po∨dr∨ppi}(ne(b),n(a))
{po∨dr}(nw(b),s(a))	dr(nw(b),se(a))	{po∨dr}(nw(b),sw(a))	{po∨dr∨ppi}(nw(b),n(a))
{po∨dr}(w(b),s(a))	{po∨dr}(w(b),se(a))	{pp∨po}(w(b),sw(a))	{po∨dr∨ppi}(w(b),n(a))
po(o(b),s(a))	po(o(b),se(a))	po(o(b),sw(a))	{po∨dr∨ppi}(o(b),n(a))
{po∨dr}(e(b),s(a))	{pp∨po}(e(b),se(a))	{po∨dr}(e(b),sw(a))	{po∨dr∨ppi}(e(b),n(a))
{pp∨po}(s(b),s(a))	{pp∨po}(s(b),se(a))	{pp∨po}(s(b),sw(a))	dr(s(b),n(a))
{po∨dr}(sw(b),s(a))	{po∨dr}(sw(b),se(a))	pp(sw(b),sw(a))	{po∨dr}(sw(b),n(a))
{po∨dr}(se(b),s(a))	pp(se(b),se(a))	{po∨dr}(se(b),sw(a))	{po∨dr}(se(b),n(a))

dr(b,ne(a))	dr(b,nw(a))	dr(b,w(a))	dr(b,o(a))
{pp∨po∨dr∨eq∨ppi}(n(b),ne(a))	{pp∨po∨dr∨eq∨ppi}(n(b),nw(a))	{po∨dr∨ppi}(n(b),w(a))	{po∨dr∨ppi}(n(b),o(a))
{pp∨po∨dr∨eq∨ppi}(ne(b),ne(a))	{po∨dr}(ne(b),nw(a))	{po∨dr}(ne(b),w(a))	{po∨dr∨ppi}(ne(b),o(a))
{po∨dr}(nw(b),ne(a))	{pp∨po∨dr∨eq∨ppi }(nw(b),nw(a))	{po∨dr∨ppi}(nw(b),w(a))	{po∨dr∨ppi}(nw(b),o(a))
dr(w(b),ne(a))	{pp∨po∨dr∨eq∨ppi}(w(b),nw(a))	{pp∨po∨dr∨eq∨ppi}(w(b),w(a))	{po∨dr∨ppi}(w(b),o(a))
{po∨dr∨ppi}(o(b),ne(a))	{po∨dr∨ppi}(o(b),nw(a))	{po∨dr∨ppi }(o(b),w(a))	{po∨dr∨ppi}(o(b),o(a))
{pp∨po∨dr∨eq∨ppi}(e(b),ne(a))	{po∨dr}(e(b),nw(a))	dr(e(b),w(a))	{po∨dr∨ppi}(e(b),o(a))
dr(s(b),ne(a))	dr(s(b),nw(a))	{po∨dr∨ppi}(s(b),w(a))	{po∨dr∨ppi}(s(b),o(a))
dr(sw(b),ne(a))	{po∨dr}(sw(b),nw(a))	{po∨dr∨ppi }(sw(b),w(a))	{po∨dr∨ppi}(sw(b),o(a))
{po∨dr}(se(b),ne(a))	dr(se(b),nw(a))	{po∨dr}(se(b),w(a))	{po∨dr∨ppi}(se(b),o(a))

dr(b,e(a))	dr(b,s(a))	dr(b,se(a))	dr(b,sw(a))
{po∨dr∨ppi}(n(b),e(a))	dr(n(b),s(a))	dr(n(b),se(a))	dr(n(b),sw(a))
{po∨dr∨ppi}(ne(b),e(a))	{po∨dr}ne(b),s(a))	{po∨dr}(ne(b),se(a))	dr(ne(b),sw(a))
{po∨dr}(nw(b),e(a))	{po∨dr}(nw(b),s(a))	dr(nw(b),se(a))	{po∨dr}(nw(b),sw(a))
dr(w(b),e(a))	{po∨dr∨ppi}(w(b),s(a))	dr(w(b),se(a))	{pp∨po∨dr∨eq∨ppi}(w(b),sw(a))
{po∨dr∨ppi}(o(b),e(a))	{po∨dr∨ppi}(o(b),s(a))	{po∨dr∨ppi}(o(b),se(a))	{po∨dr∨ppi}(o(b),sw(a))
{pp∨po∨dr∨eq∨ppi}(e(b),e(a))	{po∨dr∨ppi}(e(b),s(a))	{pp∨po∨dr∨eq∨ppi}(e(b),se(a))	dr(e(b),sw(a))
{po∨dr∨ppi}(s(b),e(a))	{pp∨po∨dr∨eq∨ppi}(s(b),s(a))	{pp∨po∨dr∨eq∨ppi}(s(b),se(a))	{pp∨po∨dr∨eq∨ppi}(s(b),sw(a))
{po∨dr}(sw(b),e(a))	{po∨dr∨ppi}(sw(b),s(a))	{po∨dr}(sw(b),se(a))	{pp∨po∨dr∨eq∨ppi}(sw(b),sw(a))
{po∨dr∨ppi}(se(b),e(a)	{po∨dr∨ppi}(se(b),s(a))	{pp∨po∨dr∨eq∨ppi}(se(b),se(a))	{po∨dr}(se(b),sw(a))

ppi(b,n(a))	ppi(b,ne(a))	ppi(b,nw(a))	ppi(b,w(a))
dr(n(b),n(a))	dr(n(b),ne(a))	dr(n(b),nw(a))	dr(n(b),w(a))
dr(ne(b),n(a))	dr(ne(b),ne(a))	dr(ne(b),nw(a))	dr(ne(b),w(a))
dr(nw(b),n(a))	dr(nw(b),ne(a))	dr(nw(b),nw(a))	dr(nw(b),w(a))
dr(w(b),n(a))	dr(w(b),ne(a))	dr(w(b),nw(a))	dr(w(b),w(a))
ppi(o(b),n(a))	ppi(o(b),ne(a))	ppi(o(b),nw(a))	ppi(o(b),w(a))
dr(s(b),n(a))	dr(e(b),ne(a))	dr(e(b),nw(a))	dr(e(b),w(a))
dr(sw(b),n(a))	dr(s(b),ne(a))	dr(s(b),nw(a))	dr(s(b),w(a))
dr(se(b),n(a))	dr(sw(b),ne(a))	dr(sw(b),nw(a))	dr(sw(b),w(a))
	dr(se(b),ne(a))	dr(se(b),nw(a))	dr(se(b),w(a))

ppi(b,o(a))	ppi(b,e(a))	ppi(b,s(a))	ppi(b,se(a))
dr(n(b),o(a))	dr(n(b),e(a))	dr(n(b),s(a))	dr(n(b),se(a))
dr(ne(b),o(a))	dr(ne(b),e(a))	dr(ne(b),s(a))	dr(ne(b),se(a))
dr(nw(b),o(a))	dr(nw(b),e(a))	dr(nw(b),s(a))	dr(nw(b),se(a))
dr(w(b),o(a))	dr(w(b),e(a))	dr(w(b),s(a))	dr(w(b),se(a))
ppi(o(b),o(a))	ppi(o(b),e(a))	ppi(o(b),s(a))	ppi(o(b),se(a))
dr(e(b),o(a))	dr(e(b),e(a))	dr(e(b),s(a))	dr(e(b),se(a))
dr(s(b),o(a))	dr(s(b),e(a))	dr(s(b),s(a))	dr(s(b),se(a))
dr(sw(b),o(a))	dr(sw(b),e(a))	dr(sw(b),s(a))	dr(sw(b),se(a))
dr(se(b),o(a))	dr(se(b),e(a))	dr(se(b),s(a))	dr(se(b),se(a))

ppi(b,sw(a))
dr(n(b),sw(a))
dr(ne(b),sw(a))
dr(nw(b),sw(a))
dr(w(b),sw(a))
ppi(o(b),sw(a))
dr(e(b),sw(a))
dr(s(b),sw(a))
dr(sw(b),sw(a))
dr(se(b),sw(a))

Table 3. Topological Relations between the *Tiles* of Both Regions *a* and *b*.

Bucciarelli et.al [1999] has composed quantified relations (known as syllogistic reasoning in Cognitive Science) between two general entities. However, the procedure for composing two quantified direction relations is quite different. We use the generated topological relations between the *tiles* of regions *a* and *b* in Table 3 and the composition table (Table 4) for the RCC-5 relations [Bennett, 1994], to compute the composition of quantified cardinal direction relations. We shall verify the outcome of the composition diagrammatically.

	dr(b,c)	po(b,c)	eq(b,c)	pp(b,c)	ppi(b,c)
dr(a,b)	U	dr∨po∨pp	dr	dr∨po∨pp	dr
po(a,b)	dr∨po∨ppi	U	po	po∨pp	dr∨po∨ppi
eq(a,b)	dr	po	eq	pp	ppi
pp(a,b)	dr	dr∨po∨pp	pp	pp	U
ppi(a,b)	dr∨po∨ppi	po∨ppi	ppi	ovr	ppi

Note: U means dr∨po∨eq∨pp∨ppi

Table 4. Composition Table for RCC-5 Relations [Bennett, 1994]

6.1 Examples of Syllogistics Reasoning about Hybrid Cardinal Direction Relations

Example 1. All *b* are in ne(*a*) and All *c* are in ne(*b*)
Use **D7**, and the boolean algebraic expression for the above composition is as follows:
$\forall[in(b, ne(a))] \land \forall[in(c, ne(b))] \equiv$
$[pp(b, ne(a)) \lor eq(b, ne(a))] \land [pp(c, ne(b)) \lor eq(c, ne(b))]$ (1)

Given the relations pp(*b*, ne(*a*)) and eq(*b*, ne(*a*)), use Table 3 to establish the relationship between ne(*b*) and ne(*a*) in order to make the appropriate transitive inferences. Based on the information in Table 3, substitute the following into Equation (1): pp(*b*, ne(*a*))

with pp(ne(b),ne(a)); and eq(*b*,ne(*a*)) with pp(ne(b),ne(a)). After substitution, Equation (1) will be as follows:
$[pp(ne(b),ne(a)) \lor pp(ne(b),ne(a))] \land [pp(c,ne(b)) \lor eq(c,ne(b))]$
(1.1)
Use Idempotent Law for union, and Equation (1.1) becomes:
$[pp(ne(b),ne(a))] \land [pp(c, ne(b)) \lor eq(c, ne(b))]$ (1.2)
Use the Distributive Law and Equation (1.2) is expanded as follows:
$[pp(ne(b),ne(a))] \land [pp(c,ne(b)) \lor eq(c,ne(b))] \equiv$
$[pp(ne(b),ne(a)) \land pp(c,ne(b))] \lor [pp(ne(b),ne(a)) \land eq(c,ne(b))]$
(1.3)
Use the composition table in Table 4 and Equation (1.3) becomes:
$[pp(c, ne(b))] \lor [pp(c, ne(b))]$ (1.4)
$\equiv [pp(c, ne(b))]$ (1.5)
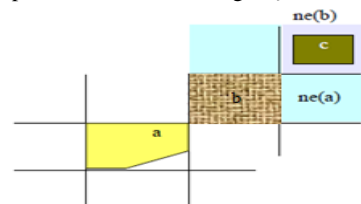{note: use Idempotent Law for union again}

Figure 6: All *b* is in ne(*a*) and All *c* is in ne(*b*)

Compare the relation in Equation (1.5) with the aforementioned assertion **D7** and we obtain this quantified relation: All *c* is in the ne(*a*) tile. When we compare the relation with assertion **D8**, then we have the following conclusion: Some *c* is in the ne(a) tile. This is confirmed by the diagrammatic representation in Figure 6.

Example 2. All b are in se(a) and All c are in n(b)

Use **D7**, and the boolean algebraic expression for the above composition is as follows:

$\forall[\text{in}(b, \text{se}(a))] \wedge \forall[\text{in}(c, \text{n}(b))] \equiv [\text{pp}(b, \text{se}(a)) \vee \text{eq}(b, \text{se}(a))] \wedge$
$[\text{pp}(c, \text{n}(b)) \vee \text{eq}(c,\text{n}(b))]$ (2)

Given the relations pp(b, se(a)) and eq(b, se(a)), use Table 3 to establish the relationship between n(b) and se(a) in order to make the appropriate transitive inferences. Based on the information in Table 3, substitute the following into Equation (2): pp(b, se(a)) with {po$\vee$ dr}(n(b),se(a)); and eq(b, se(a)) with {po$\vee$ dr}(n(b),se(a)). After substitution, Equation (2) will be as follows:

$[\{\text{po}\vee\text{dr}\}(\text{n}(b),\text{se}(a))\vee\{\text{po}\vee\text{dr}\}(\text{n}(b),\text{se}(a))]\wedge[\text{pp}(c,\text{n}(b))\vee$
$\text{eq}(c,\text{n}(b))]$ (2.1)

Use Idempotent Law for union, and Equation (2.1) becomes:

$[\{\text{po}\vee \text{dr}\}(\text{n}(b),\text{se}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]$ (2.2)
$\equiv[\text{po}(\text{n}(b),\text{se}(a)) \vee \text{dr}(\text{n}(b),\text{se}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]$

Use the Distributive Law and Equation (2.2) is expanded as follows:

$\{[\text{po}(\text{n}(b),\text{se}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]\} \vee$
$\{[\text{dr}(\text{n}(b),\text{se}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]\}$
$\equiv\{[\text{po}(\text{n}(b),\text{se}(a)) \wedge\text{pp}(c,\text{n}(b))] \vee [\text{po}(\text{n}(b),\text{se}(a)) \wedge \text{eq}(c, \text{n}(b))]\}\vee$
$\{[\text{dr}(\text{n}(b),\text{se}(a)) \wedge\text{pp}(c,\text{n}(b))]\vee[\text{dr}(\text{n}(b),\text{se}(a)) \wedge \text{eq}(c, \text{n}(b))]\}$
                                                                                         (2.3)

Use the composition table in Table 4 and Equation (2.3) becomes:

$\{\text{po}\vee\text{pp}\}(c,\text{se}(a))\vee\text{po}(c,\text{se}(a))\vee\{\text{dr}\vee\text{po}\vee\text{pp}\}(c,\text{se}(a))\vee\text{dr}(c,\text{se}(a))$
                                                                                         (2.4)
$\equiv\{\text{dr}\vee\text{po}\vee\text{pp}\}(c,\text{se}(a))$ (2.5)

{note: use Idempotent Law for union again}

The outcome of the composition is $\{(\text{dr}\vee \text{po}\vee \text{pp})\}(c,\text{se}(a))$. Compare the relation with the aforementioned assertions (**D7 to D10**) and the following conclusions hold: All c is in se(a); Some c is in se(a); No c is in se(a); Some c is in not in se(a);. This is confirmed by the diagrammatic representation in Figure 7 (note: the representation for each category is not exhaustive).

pp(c, se(a))   po(c, se(a))   dr(c, se(a))
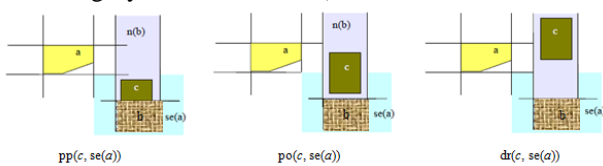
Figure 7. All b is in se(a) and All c is in n(b)

Example 3. No b are in w(a) and All c are in n(b)

Use **D7** and **D9**, and the boolean algebraic expression for the above composition is as follows:

$\forall[\neg\text{in}(b, \text{w}(a))] \wedge \forall[\text{in}(c, \text{n}(b))] \equiv [\text{dr}(c, \text{w}(b))] \wedge [\text{pp}(c, \text{n}(b)) \vee$
$\text{eq}(c, \text{n}(b))]$ (3)

Given the relation dr(b, w(a)), use Table 3 to establish the relationship between n(b) and w(a) in order to make the appropriate transitive inferences. Based on the information in Table 3, substitute the following into Equation (3): dr(b, w(a)) with {po$\vee$ dr$\vee$ppi}(n(b),w(a)). After substitution, Equation (3) will be as follows:

$[\{\text{po}\vee \text{dr}\vee\text{ppi}\}(\text{n}(b),\text{w}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]$ (3.1)
$\equiv[\text{po}(\text{n}(b),\text{w}(a)) \vee \text{dr}(\text{n}(b),\text{w}(a)) \vee \text{ppi}(\text{n}(b),\text{w}(a))] \wedge [\text{pp}(c, \text{n}(b))$
$\vee \text{eq}(c, \text{n}(b))]$ (3.2)

Use the Distributive Law and Equation (3.2) is expanded as follows:

$[\text{po}(\text{n}(b),\text{w}(a))] \wedge [\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))] \vee [\text{dr}(\text{n}(b),\text{w}(a))] \wedge$
$[\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))] \vee [\text{ppi}(\text{n}(b),\text{w}(a))] \wedge$
$[\text{pp}(c, \text{n}(b)) \vee \text{eq}(c, \text{n}(b))]$ (3.3)
$\equiv\{[[\text{po}(\text{n}(b),\text{w}(a))]\wedge[\text{pp}(c,\text{n}(b))]]\vee[[\text{po}(\text{n}(b),\text{w}(a))]\wedge[\text{eq}(c,\text{n}(b))]]\}$
$\vee\{[[\text{dr}(\text{n}(b),\text{w}(a))]\wedge[\text{pp}(c,\text{n}(b))]]\vee[[\text{dr}(\text{n}(b),\text{w}(a))]\wedge[\text{eq}(c,\text{n}(b))]]\}$
$\vee\{[[\text{ppi}(\text{n}(b),\text{w}(a))]\wedge[\text{pp}(c,\text{n}(b))]]$
$\vee[[\text{ppi}(\text{n}(b),\text{w}(a))]\wedge[\text{eq}(c,\text{n}(b))]]\}$ (3.4)

Use the composition table in Table 3 and Equation (3.4) becomes:

$\{\text{po}\vee\text{pp}\}(c, \text{w}(a)) \vee \text{po}(c, \text{w}(a)) \vee \{\text{dr}\vee\text{po}\vee\text{pp}\}(c, \text{w}(a))$
$\vee \text{dr}(c, \text{w}(a)) \vee \text{ovr}(c, \text{w}(a)) \vee \text{ppi}(c, \text{w}(a))$ (3.5)

Use Idempotent Law for union, and Equation (3.5) becomes:

$\{\text{po}\vee\text{pp}\vee\text{dr}\vee\text{ppi}\}(c, \text{w}(a)) \vee \text{ovr}(c, \text{w}(a))$ (3.6)

Based on **D.1**, the topological relationship ovr(c, w(a)) $\equiv \{\text{po}\vee\text{pp}\vee\text{eq}\vee\text{ppi}\}(c, \text{w}(a))$. Substitute this into Equation (3.6), use Idempotent Law for union again, and Equation (3.6) becomes: $\{\text{po}\vee\text{pp}\vee\text{eq}\vee\text{dr}\vee\text{ppi}\}(c, \text{w}(a))$ {note this is the universal set of relations for RCC-5}

Compare the relation with **D7 to D10** and we obtain the following quantified relations: union of **D7** and **D10** – Some c is not or All c is in w(a) tile; union of **D8** and **D9** – Some c is or No c is in w(a) tile. This concurs with the diagrammatic representation in Figure 8 (note: the representation for each category is not exhaustive).
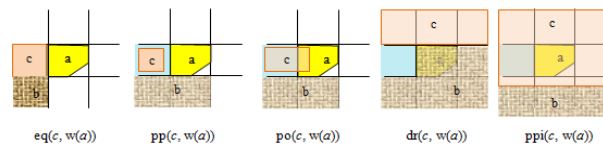

eq(c, w(a))   pp(c, w(a))   po(c, w(a))   dr(c, w(a))   ppi(c, w(a))

Figure 8. No b is in w(a) and All c is in n(b)

7. Conclusions

In this paper, we have employed RCC-5 and quantifiers (*All* and *Some*) to formalize quantified direction relations. We have developed a procedure for computing the composition of such quantified direction relations. The correctness of the outcome of the composition is further verified diagrammatically. However, based on the worked out examples, the conclusions derived from the inferences made are rather weak. Thus, further research will be necessary to investigate ways to enhance the reasoning with RCC-5, cardinal directions and syllogisms.

8. Bibliography

Allen, J. F. (1983). Maintaining Knowledge about interval intervals. ACM Communications, pp.832-843.

Balbiani, P., Condotta, J., & del Cerro, L. F. A. (1998). Model for Reasoning about Bidemnsional Temporal Relations. Proceedings of KRR.

Bennett, B. (1994). Spatial Reasoning with Propositional Logics. Proceedings of KRR.

Bucciarelli, M., & Johnson-Laird, P. N. (1999). Strategies in Syllogistic Reasoning. Cognitive Science, Vol. 23(3), pp.247-303.

Cao, H., Wang, X., and Chen, J. (2008). Design of an extended qualitative cardinal direction relation model based on ontology, In Proceedings of the Geoinformatics 2008 and Joint Conference on GIS and Built Environment: Advanced Spatial Data Models and Analyses (November 2008).

Chen, J., Liu, D., Zhang, C., and Xie, Q. (2007). Combinative Reasoning with RCC5 and Cardinal Direction Relations, Knowledge Science, Engineering and Management, Lecture

Notes in Computer Science, 2007, Volume 4798/2007, 92-102, DOI: 10.1007/978-3-540-76719-0_13.

Chen, J., Jia, H. Y., Liu, D. Y., and Zhang, C. H. (2010a). Composing Cardinal Direction Relations Based on Interval Algebra, International Journal of Software Informatics, Vol.4, No.3, September 2010, pp. 291-303, ISSN 1673-7288.

Chen, J., Schneider, M., Viswanathan, G., and Yuan, W. J. (2010b). The Objects Interaction Matrix for Modeling

Cardinal Directions in Spatial Databases, Database Systems for Advanced Applications , Lecture Notes in Computer Science Volume 5981, 2010, pp 218-232.

Cicerone, S. and Di Felice, P. (2004). Cardinal Directions between Spatial Objects: The Pairwise Consistency Problem. Information Sciences – Informatics and Computer Science: An International Journal Vol. 164, Issue 1-4, pp.165-188.

Clementini, E., and Di Felice, P. (1996). A Model for Representing Topological Relationships between Complex Geometric Features in Spatial Databases, Information Sciences, 90, pp.121-136.

Clementini, E., Di Felice, P. and Hernandez. (1997). Qualitative Representation and Positional Information. Artificial Intelligence, 95, pp.315-356.

Cohn, A.G., Bennett, B., Gooday, J., & Gotts, N.M. (1997). Qualitative Spatial Representation and Reasoning with the Region Connection Calculus.

Egenhofer, M. J. and Sharma, J. (1993). Assessing the Consistency of Complete and Incomplete Topological Information. Geographical Systems, 1(1), pp.47-68.

Egenhofer, M. J., and Mark, D. M. (1995). Naïve Geography, Technical Report 95-8, National Center for Geographic Information and Analysis.

Escrig, M.T. and Toledo, F. (1998).A framework based on CLP extended with CHRS for reasoning with qualitative orientation and positional information. JVLC, 9, pp.81-101.

Frank, A (1992). Qualitative Spatial Reasoning with Cardinal Directions. JVLC, Vol(3), pp.343-371.

Freksa, C. (1992). Using orientation information for qualitative spatial reasoning. Proceedings of International Conference GIS – From space to territory, Theories and methods of spatio-temporal reasoning in Geographic space, pp.162-178.

Gabelaia, D., Kontchakov, R., Kurucz, A., Wolter, F., and Zakharyaschev, M. (2003). On the Computational Complexity of Spatio-Temporal Logics, FLAIRS 2003, American Association for Artificial Intelligence.

Gerevini, A., and Renz, J. (2002). Combining topological and size information for spatial reasoning, Artificial Intelligence, 137, pp. 1-42.

Goyal, R. and Egenhofer, M (2000). Consistent Queries over Cardinal Directions across Different Levels of Detail. 11th International Workshop on Database and Expert Systems Applications, Greenwich, UK, 2000.

Goyal, R. & Egenhofer, M. J. (2001). Cardinal Directions between Extended Spatial Objects. IEEE Transactions on Knowledge and Data Engineering

Isli, A. (2004). Combining cardinal direction relations and other orientation relations in QSR∗†, Proceedings of 8th International Symposium on Artificial Intelligence and Mathematics (AIEM '04).

Johnson-Laird, P. N. (1983). Mental Models. Cambrdge: Combridge University Press.

Knauff, M., Rauh, R., Schlieder, C., & Strube, G. Mental Models in Spatial Reasoning.

Kor, A. L. & Bennett, B. (2013). A Hybrid Reasoning Model for "Whole and Part" Cardinal Direction Relations, Journal of Advances in Artificial Intelligence, Vol. 2013, Hindawi Publishing Corporation.

Kor, A.L. & Bennett, B. (2010). Reasoning Mechanism for Cardinal Direction Relations, In In Dicheva, D., and Dochev, D. (eds.). AIMSA 2010, LNAI 6304, pp. 32-41, Springer, Heidelberg, 2010.

Kor, A.L. & Bennett, B. (2003). Composition for Cardinal Directions by Decomposing Horizontal and Vertical Constraints. Proceedings of AAAI 2003 Spring Symposium, March 24-26.

Li, S. J. (2006). Combining Topological and Directional Information: First Results. In: Lang, J., Lin, F., Wang, J. (eds.) KSEM 2006. LNCS (LNAI), vol. 4092, pp. 252-264. Springer, Heidelberg.

Li, S. J. (2007). Combining topological and directional information for spatial reasoning, Proceedings of IJCAI'07 Proceedings of the 20th international joint conference on Artifical intelligence, pp. 435-440.

Ligozat, G. (1988). Reasoning about Cardinal Directions. JVLC, 9, pp.23-44.

Liu, W. M., Li, S. J., and Renz, J. (2009). Combining RCC-8 with Qualitative Direction Calculi: Algorithms and Complexity, Proceedings of the Twenty-First IJCAI-09.

Moratz, R., and Wallgrun, J. O. (2012). Spatial reasoning with augmented points: Extending cardinal directions with local distances, Journal Of Spatial Information Science, Number 5, pp. 1–30.

Papadias, D. and Theodoridis, Y. (1997). Spatial relations, minimum bounding rectangles, and spatial data structures.Technical Report KDBSLAB-TR-94-04.

Pratt, I., and Francez, N. (1998). Temporal Prepositions and Temporal Generalized Quantifiers.

Randell, D., Cui, Z.,& Coh, A. G. (1992). A Spatial Logic based on Regions and Connection. Proceedings of the Third International Conference on Knowledge Representation and Reasoning.

Schneider, M., Chen, T., Viswanathan, G., And Yuan, W. J. (2012). Cardinal Directions between Complex Regions, ACM Transactions on Database Systems, Vol. 37(2).

Sharma, J., and Flewelling, D. (1995). Inferences from combined knowledge about topology and directions. Advances in Spatial Databses, 4th International Symposium, Portland, Maine, pp.271-291.

Sharma, J., Flewelling, D., and Egenhofer, M. (1994). A Qualitative Spatial Reasoner, proceedings of the Sixth International Symposium on Spatial Data Handling, Edinburgh, Scotland, pp. 665-681.

Skiadopoulos, S. and Koubarakis, M (2001). Composing Cardinal Direction Relations. Proceedings of SSTD-01, Redondo Beach, CA, USA.

Skiadopoulos, S. and Koubarakis, M (2004). Composing Cardinal Direction Relations. Artificial Intelligence, Vol.152(2), pp.143-171.

Theodoridis, Y.; Papadias, D.; Stefanakis, E.; and Sellis, T. (1998). Direction relations and two-dimensional range queries: optimisation techniques, Data and Knowledge Engineering,Vol.27(3), pp. 313-336.

Varzi, A.C. (1996). Parts, Wholes, and Part-Whole Relations: The prospects of Mereotopology, Data and Knowledge Engineering, 20, pp.259-286.

Yao, X. B., and Thill, J. (2006). Spatial queries with qualitative locations in spatial information systems, Computers, Environment and Urban Systems, Volume 30(4), Issue 4, July 2006,pp. 485–502.

Zhang, X. T., Liu, W.M., Sanjiang Li, S.J., and Ying, M. S. (2008).Reasoning with Cardinal Directions: An Efficient Algorithm, Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence.

An Ontology for Semantic Representation of an Urban Virtual Environment

K. Harkouken Saiah[1], N. Sabouret[2], and J-Y. Donnart[3]

[1]LIP6, UPMC, Paris, 75005, France
[2]LIMSI, CNRS, Orsay, 91403, France
[3]THALES Training & Simulation / Cergy-Pontoise, 95525, France

Abstract— *We present in this paper a model of semantic representation of a dynamic virtual environment. Our model is embodied into a simulation architecture of a virtual city. The objective is to enable agents to make action decision consistent with the semantic state of their environment. The idea is to represent the services offered by the environment into an ontology of services and deduce the available object that has the best quality of service at any given time of the simulation. For this, we defined a layer of the ontology containing the general concepts of representation of the environment. This representation allows us to apply a unified model of reasoning to infer elements of knowledge useful for decision-making agents.*

Keywords: Knowledge Representation and Reasoning, Semantic Virtual Environments, Ontology, Urban Simulation

1. Introduction

Knowledge representation for virtual environments (VE) became an important issue with increasing interest in the last decade [1] [2]. Indeed, the agents operating in VE need semantic informations on it to adapt their behavior to identifiable elements of that environment and making a choice of action [3]. Semantic Virtual Environments (SVE) [4] have been proposed to address this lack of semantic informations (environmental knowledge) necessary to support intelligent interactions between agents and their environment. In fact, SVE provide a coherent representation of (i) the simulated world and the behavior of its entities, (ii) the interactions and tasks that users and agents can perform in the environment, and (iii) elements of knowledge that the agents can use to make decisions [5].

The work carried out for the representation of VE are divided into two categories corresponding to two levels of semantics.

The first category of these works aims at an explicit representation of the contents of the environment to reproduce the scenes. Therefore, the agents have direct access to the definitions of entities of the environment. These representations are mainly geometric and topological information of the environment. Several approaches have been proposed for this representation. Most of them offer an ontological representation [6] containing different types of environmental information for different types of application.

Otto [4] uses domain ontologies to create virtual environments in a platform he called *"SeVEn"*. The goal is to find a representation of the environment independent from the software description to create virtual objects that can be reused in several virtual environments. This reuse is possible by adding information relevant to the tasks of an object in the field of virtual environments generation. Following the same principle, [7], [8] propose to generate a virtual world from an ontology. It contains a description of the objects and their properties defined by domain experts.

Kalogerakis et al. [9] have defined an ontology *"X3DOntology"* associated with entities defined in description files of 3D virtual worlds. This provide a semantic representation of the scene to infer its content. We can find the same approach in [10] but at a higher level, making the correspondence between the objects of the virtual environment and the concepts of the ontology.

These works give agents direct access to descriptions of VE entities. However, this definition of the world doesn't allow them to deduce information for decision support to identify (i) the object that allows them to perform an action, (ii) what actions can be performed on an object (iii) the relationship between two objects, (iv) and the relationship between actions.

The second category of works on semantic representation of VE has been made to strengthen the agent-object interactions that include not only information related to the geometry of the scene, but also knowledge about the possible interactions with the objects in the environment [11] (what can be done? how this can be done? why and how it should be done?).

In [12] and [13], the authors have extended the principle of using ontologies to represent SVE incorporating the semantic description related to the function of objects so that agents can design their own animation procedures like path planning.

Kallmann & Thalmann [14] have proposed *Smart Object* whose idea is to include in the description of the object all the required information to describe how to interact with it. They distinguish information on: (i) the properties of the object (semantic and physical), (ii) how to interact with

the object (actions, positions, gestures), (iii) the behavior of the object in response to an action and (iv) the behavior of actors to achieve interaction. Following the same philosophy, Badawi [15] proposes Synoptic Objects *STAR FISH*. The idea is to define a minimum set of primitive actions which are used to build complex actions.

In a lower level of representation [2] they developed a meta-model to represent the semantic of the VE and its structural properties, geometrical, topological, the behavior of agents and interactions between agents and users.

These works propose a semantic representation of VE which offers data for decision-making of agents. This give them the ability to interact with the VE and avoid having aberrant behaviors (that never occur in real life). But these solutions do not give agents the elements of knowledge to make *the best choice* of action for a given situation. They can provide the utility and function of an object of the environment but agents should make the connection with their actions plan to decide if it suits them or not.

Instead, we propose in this paper a method that avoids agents to find which element of the environment will allow them to carry out their plan. The idea is to provide agents with a sorted list of objects according to their relevance regarding their actions plans. Our goal is not to influence the decisions of agents or decide for them, but we aim to provide them with knowledge they need to make a decision consistent with the context. The goal of our approach is to decentralize and capitalize calculations made by agents by strengthening communication links between the decision-making of the agents and the representation of their environment. Our choice is justified by the complexity of the calculations for the decision and the large number of agents contained in the SVE.

Published works with similar objectives, propose a representation of the environment that integrates the description of possible actions that agents can perform. In this case, the interactions between the elements of the simulated world and the agents are managed by their environment and are related to what is defined in it. However, the heterogeneity of SVE (different categories of its elements) increases the size of the representation and affects the calculations cost to be performed by agents.

Our first contribution is to overcome this problem by proposing a unified representation model of the environment which allows us to do a generic calcul of decision support elements (measure of relevance between the action of the agent and function given by an element of the environment). The idea is to have simple calculations and applicable to all elements of the environment by treating uniformly requests of agents. Our study focuses on modeling the semantics of a virtual urban environment which has the distinction of being complex and heterogeneous, ie, it is composed of several types of objects with different features. The objective is to propose a model to represent the semantic of the

environment allowing us to apply a unified treatment for all its components.

Our second contribution is to represent the types of objects that make up the environment by services (services offered by objects of the environment), where each service of the environment will be evaluated according to the type of object that proposed it. We were inspired by the techniques of semantic web [16] [17] by defining the elements of the environment in a *Ontology of Services*. We consider that each object type offers one or more *services* with some *quality*. The Quality of Service is derived from the unified reasoning we have applied on our representation model and it will be used by agents to make their decision.

In the remainder of this article, we present the architecture on which we are working, and then we describe the representation model we proposed. Finally, we discuss the first results of the implementation of this model and perspectives of our work.

2. Architecture

Our representation module of the environment is part of a platform of the project *Terra Dynamica*[1], which is a simulation project of a virtual dynamical city. The architecture of the project consists of several modules interacting with each other (semantic representation of the environment, decision-making, affective, path-finding and patrol).

As shown in figure 1, the environment semantic representation module interacts with the decision-making module of the agents. These interactions are summarized in an exchange of requests / responses between the two modules. The decision-making module queries the semantic representation module so that it offers him the environment entities enabling to define an action plan for the agents or to choose the best element of the environment allowing them to make an action. For the latter case, we distinguish two situations (i) agents that want to achieve immediate action (which they consider very important) and (ii) agents that want to perform a task plan in an opportunistic manner (when a opportunistic situation occurs).
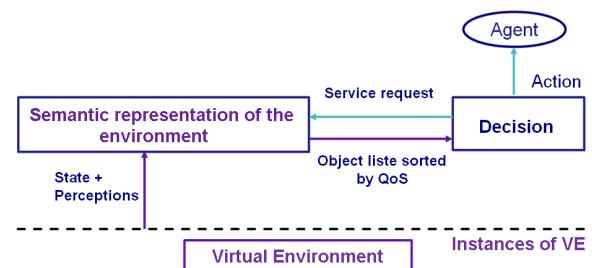
Fig. 1: System architecture: interactions between semantic module representation of the VE and the decision-making module of the agents.

[1]http://www.terradynamica.com/

We therefore considered three types of interactions between the semantic representation module and the decision-making module :

- **Interrogative mode:** during the planning phase the semantic representation module is requested to provide information on the environment to make a plan of action. The semantic module responds to these requests by providing the elements of the environment allowing to achieve the action plan in line with the context of the environment (Query: what allows me to eat, Answer: a restaurant, sandwich, etc.).
- **Reactive mode:** it occurs during the carrying out of the agents plan. In this case, the agents know what element of the environment they want to use (defined during the planning step). When they are preparing to use a service provided by this element, a request is submitted to the semantic module to return the quality of service in relation with the context on one hand and action on the other hand.
- **Pro-active mode:** agents can submit an application of interest for a given service. The semantic module takes account for propose the elements of the environment providing the service when they arise.

We also distinguish two types of queries sent by the decision-making modules of the agents:

- Request to obtain a given service (for interrogative mode and pro-active mode),
- Request on the consistency to use an element of the environment for obtaining a given service (for reactive mode).

We will see in section 4 how these queries are implemented.

3. Model

Our representation model of the environment is based on a two-level ontology that provides support for a generic mechanism for reasoning about the dynamic and functioning of the simulated environment (see Fig. 2).

3.1 Ontology of Services

In our model, an ontology is a pair $\mathcal{O} = <\mathcal{C}, \mathcal{R}>$, where $\mathcal{C}$ is the set of concepts and $\mathcal{R}$ is the set of binary relations. The ontology of services is not limited to relations of hierarchy (isa) or meronymy $(partof)$, but we also defined specific relationships with our goal of representation.

The generic level of our ontology of services contains general knowledge of environmental concepts $\mathcal{C}$ which consists of three subsets $\mathcal{C} = S \cup O \cup C_r$ where:

- S the set of all services,
- O the set of all types of objects in the environment,
- C_r collects the evaluation criteria of the quality of service rendered by an object type.
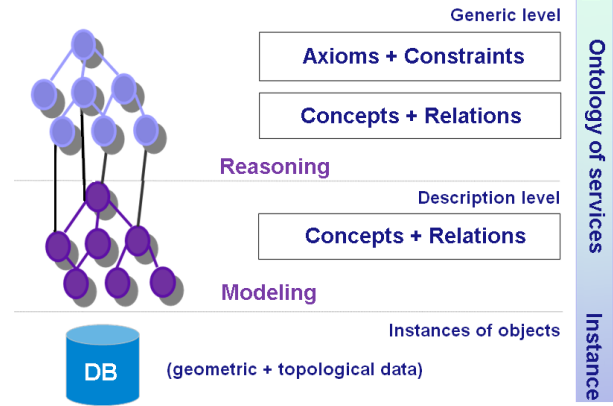
Fig. 2: Representation model of the environment: Ontology of services

Definition of object types

An object type is an element of the environment providing one or more services. Each object type provides a service with a certain quality (rendering service).

Definition of services

A service is a tuple $\langle name, C_r(s) \subset C_r, O(s) \subset O \rangle$ such that $name$ is the service name, $C_r(s)$ the set of criteria for evaluating service s and $O(s)$ the set of object types providing this service.

We have defined the relationship $offers(o, s) \in O \times S$ between services and object types. It allows us to say that one object type o offers a service $s \in S$ such that $o \in O(s)$. For example, the service *Eat* can be offered by several types of objects $O(Eat) = \{Restaurant, Fast Food, Cafeteria, etc\}$, this service will be evaluated according to the criteria $C_r(Eat) = \{Cost, Time, Quantity, Quality, Ambiance\}$.

We have defined two hierarchy levels of service S_a and S_b, as we can see in Fig. 3.

- Behavior service S_b : set of services allowing to have a behavior and to respond to a motivation,
- Action service S_a : set of services allowing to make an action.

Such as $S = S_b \cup S_a$ and $S_b \cap S_a = \phi$.

This two types of services are related by the relation $Implements(s, s') \in S_a \times S_b$, which means that a behavior service can be obtained (implemented) by one or more action service. For example, the service $s = Eat \in S_b$ is a behavior service which can be achieved by different ways (implemented with different action services) $s_1 = have\ a\ snack$, $s_2 = Nibble$, etc, with $s_1, s_2 \in S_a$.

For each behavior service there is at least one action service that implements it $\forall s \in S_b, \exists s' \in S_a$ such that $implements(s', s)$, where a service action is related to a single service behavior $\forall s'' \in S_a, \exists! s \in S_b$ such that $implements(s'', s)$.
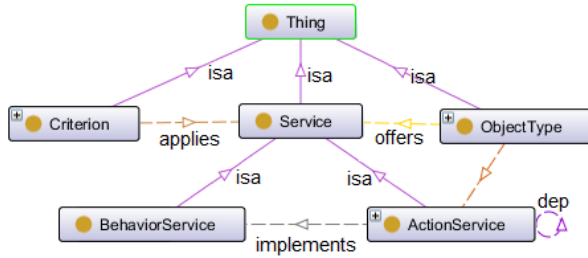
Fig. 3: General concepts of the ontology of services.

Note that the objects types only offer actions services $\forall o \in O, \forall s \in S, o \in O(s) \Rightarrow s \in S_a$.

We have defined a non-hierarchical relation *dependency* $dep(s, s')$ between two services s and s' proposed by the same object type, such as, $dep(s, s') \in 2^S$, with $o \in O(s)$ and $o \in O(s')$.

For example, the service *Drink* proposed by the *Restaurant* depends on the service *eat* proposed by the same object *Restaurant* because we can't use the service *Drink* of the restaurant if we don't intend to *eat*.

We use the dependency relationship between the services in calculating the quality of service (section 3.2.1).
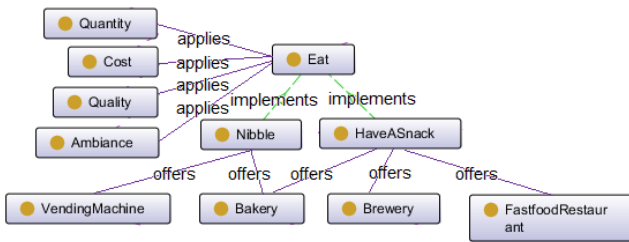
Fig. 4: Ontology of services : example of service "Eat"

Definition of criteria for evaluating services C_r

Services are evaluated according to a set of criteria. A criterion can be shared by multiple services, such as *Cost*, or it can be specific to a service, such as *quality of food* which is reserved for catering services.

Let C_r the set of criteria. Each service $s \in S$ will be assessed according to a subset of C_r which we denote $C_r(s) \subset C_r$.

The set of criteria $C_r(s)$ of an action service $s \in S_a$ is a subset of criteria $C_r(s')$ of behavior service $s' \in S_b$ it implements ($\forall s \in S_a, s' \in S_b, implements(s, s') \Rightarrow C_r(s) \subseteq C_r(s')$).

We have defined a non-hierarchical relation *applies* between services and criteria, such that $applies(c, s) \in C_r \times S$, which means that the criterion c is *applied* to evaluate the service s (see Fig. 4).

We have defined for each criterion $c \in C_r$ a set of values denoted $V(c)$. This set contains all possible values of a given

criterion c. We have for example the values V *(cost)* = {*expensive, not cher, free*}.

3.2 Quality of Service

Quality of Service QoS is calculated thanks to the following parameters:

- **Effectiveness**, to assess whether an environmental element meets the needs of the agent and allows it to perform its task.
- **Difficulty**, calculate the induced cost by additional constraints to access the desired service.
- **Proximity**, measure the accessibility of a service based on the abundance of elements in the environment that offer the desired service.

Note that the QoS is independent of agents preferences (we have set one type of calculation for any object type or service).

We note that the decisional module of agents take into account other information about the environment and the state of the simulation in addition to the QoS for making a decision (geometric and topological data extracted from the DB, distance from an object, the presence of a queue, etc.).

3.2.1 Parameters for calculating the QoS

Effectiveness

The effectiveness of an object type $o \in O$ offering a service $s \in S_a$ searched by an agent is the ratio between the number of criteria satisfied with the service provided by the object type and the total number of evaluation criteria of service s.

To do this, we define the following sets:

Let $QC(o, s) \subseteq C_r(s)$ the set of criteria that apply to assess the service s proposed by an object type $o \in O$ such that $s \in S_a, o \in O(s)$ and $\exists offers(o, s)$. For example, the service $s = $ *have a snack* proposed by the object type $o_1 = $ *Vending machine* have the set criteria as $QC(o_1, s) = \{$ *Cost, Time, Quality, Quantity* $\}$ and the type of object $o_2 = $ *Fast food restaurant* offering the same service s have the set criteria as $QC(o_2, s) = \{$ *Cost, Time, Quality, Quantity, Ambiance* $\}$.

We have defined a set of values on the criteria of the service s we noted $V_q(s, c) \subseteq V(c)$ such that $c \in C_r(s)$. The set $V_q(s, c)$ contains all possible values that can take a service s for one of its criteria c. For example, the criterion $c = $ *Quantity* for the service $s = $ *have a snack* will have a value in the set { *Medium, Small* }.

We have defined a set $EQ(s)$ *(Estimated Quality)* containing all possible combinations of pairs (criterion, value) for a service s with $EQ(s) = \{(c, v) \in C_r(s) \times V_q(s, c)\}$. In the previous example, $EQ($ *have a snack*) contains all { *(Quantity, Small), (Quantity, Medium)* }.

Let $EQ(o, s)$ the set of effects expected by the use of an object type $o \in O$ to obtain a given service $s \in S_a$ with $EQ(o, s) = \{(c, v)/c \in QC(o, s) \land v \in V_q(s, c)\}$.

For example, the service $s = $ *have a snack* proposed by the object $o = $ *Vending machine* will result in $EQ(o,s) = $ *{(Cost, Not expensive), (Time, Speed), (Quantity, Small)}*.

Efficiency $Eff(o,s)$ of the object type o relative to the requested service s is therefore calculated as follows:

$$Eff(o,s) = \frac{|EQ(s) \cap EQ(o,s)|_2}{|QC(o,s)|}$$

Difficulty

The difficulty parameter is based on additional conditions to be added to obtain a service. We consider that an object type offers a service with a difficulty when it depends on another service offered by the same object type. The difficulty $Diff(o,s)$ of the service s offered by the object type o and having a relationship of dependence $(dep(s,s'))$ with another service s' is the rate of criteria in addition to consider (criteria of s') for obtain service s.

$$Diff(o,s) = \frac{|C_r(s')| - |QC(o,s) \cap QC(o,s')|}{|C_r(s')|}$$

The difficulty is not taken into account in the case where the agent had planned an action corresponding to the service s' in its original plan. In this case, the solution to have both services will be considered an optimized solution.

Proximity

We measured the proximity of a service based on the attendance of the object type that offer this service. Our goal is to enable agents to tell the difference between one object type they can find very often in the environment, making it more accessible and less difficult to find, and an object type not very common in the environment and may be the only way for getting given service.

Let $o \in O$ an object type providing the service sought s. We defined the set $O_p(o)$ containing the instances of objects with the object type o. Let $O_s(s)$ the set of instances of the object types $o \in O$ offering the service s.

The proximity $Prox(o,s)$ of an object type o offering the service s will be calculated as the ratio between the number of its instances, and the number of instances of object types that offer same service s:

$$Prox(o,s) = \frac{|O_p(o)|}{|O_s(s)|}$$

3.2.2 calculating the QoS

The quality of service offered by an object type of the environment takes into account the three parameters we mentioned above, with a variant on the weight of their importance. For example, an agent may prefer to have an object type which provides a service with maximum efficiency

---

regardless of the means to do so (difficulty parameter) or difficult to find (proximity parameter).

We calculated the QoS by a weighted sum of three parameters as follows:

$$QoS(o,s) = \frac{p_1 Eff(o,s) + p_2 Diff(o,s) + p_3 Prox(o,s)}{\sum_i^3 p_i}$$

In the implementation of our model (next section), we assumed that all parameters have the same importance weight of $(p_1 = p_2 = p_3 = 1)$.

4. Implementation and first results

We have implemented our model in the platform development of the *Terra Dynamica* project (see Fig. 5), developed in C++. We built our ontology using the Protégé [3]. Access to the ontology is done by sending SPARQL queries [4]. These requests are processed according to the three modes of interaction defined above (interrogative, reactive, pro-active).

Requests sent by the decisional module contain the name of the service desired by the agent. These are transformed into SPARQL queries to return a set of object types offering this service. The principle of the research of object types offering a service is to explore the hierarchy of downwards services if we have a behavior service as an input, this will allow us to define the set of services that implements it (knowing that an object type is connected with an action service). Once all the services that can satisfy the agent are found, we will ask our ontology to infer all object types that allow us to have these services.

All object types that returned from querying our ontology will be sorted according to the QoS we have calculated for each object type. Finaly, the decision module selects an instance of one of these object types with the best QoS.

Fig. 5: Screenshot of the *Terra Dynamica* simulator.

The first results show that the actors of the simulation exhibit a good reactivity to the context. We have seen

---

[2]We denote by $|X|$ the number of elements of the set X

[3]http://protege.stanford.edu/
[4]http://www.w3.org/TR/rdf-sparql-query/

actors who were heading breweries and bakeries to satisfy their hunger motivation (interrogative mode).We can also see actors throw something in the garbage and continue their activity (pro-active mode). And finally, we found that actors used the objects services when it was possible for the reactive mode (no queue on the same object while another object type is available a little further).

5. Conclusion

We presented in this paper a model for representing the semantic of a complex and heterogeneous virtual environment (virtual city). We propose an ontology of services which defines the semantic of the elements of the environment through the services they offer. Our model of representation allows us to deduce in an unified manner the quality of service (QoS) provided by an object type of environment. This will be used for decision support of the agents. Our goal is to allow agents to make the best choice among all object types in the environment that enable them to achieve their action. To do this, we proposed a generic level in the ontology to represent the definition of concepts that allow us to make our calculation of QoS. The semantic of the environment is used at key moments in the evolution of agents (during the planning and execution of the plan) to accompany the agent in these behaviors by allowing it to react to what happens in its environment in a consistent manner and opportunistic.

We have implemented our model in the simulation platform project where we got the first results. The numerical evaluation of the results is not the purpose of this article but we plan to evaluate our model on its performance especially for a large number of heterogeneous elements of the environment and a large number of agents.

In the near future, we are considering the addition of a fourth parameter to assess the QoS that defines the availability of the element of environment. As it is now, the agents observe the availability of an object in the environment when they can see it. But this can be avoided by taking into account knowledge of the environment that will be used to filter out inconsistent results (eg, the banking services offered by the bank are unavailable after 6pm).

Acknowledgment

References

[1] R. Aylett and M. Cavazza, "Intelligent virtual environments: a state-of-the-art report," in *Eurographics 2001, STAR Reports volume*, 2001, pp. 87–109.

[2] P. Chevaillier, T.-h. Trinh, M. Barange, J. Soler, P. De Loor, and R. Querrec, "Semantic modelling of virtual environments using mascaret," in *Proceedings of the Fourth Workshop on Software Engineering and Architectures for Realtime Interactive Systems SEARIS, IEEE VR 2011*, 2011.

[3] T. Tutenel, R. Bidarra, R. M. Smelik, and K. J. D. Kraker, "The role of semantics in games and simulations," *Comput. Entertain*, vol. 6, pp. 1–35, 2008.

[4] K. Otto, "Towards semantic virtual environments," in *Workshop Towards Semantic Virtual Environments*, 2005.

[5] K. M. Stanney, *Handbook of virtual environments: design, implementation, and applications*, K. M. Stanney, Ed. Lawrence Erlbaum Associates, 2002.

[6] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," in *Formal ontology in conceptual analysis and knowledge representation*. Kluwer Academic Publishers, 1993.

[7] B. Pellens, O. D. Troyer, W. Bille, and F. Kleinermann, "Conceptual modeling of object behavior in a virtual environment," V. C. 2005, Ed. Springer-Verlag, 2005.

[8] O. D. Troyer, W. Bille, R. Romero, and P. Stuer, "On generating virtual worlds from domain ontologies," in *Proceedings of the 9th International Conference on Multi-Media Modeling*, 2003.

[9] E. Kalogerakis, S. Christodoulakis, and N. Moumoutzis, "Coupling ontologies with graphics content for knowledge driven visualization," in *Proceedings of the IEEE conference on Virtual Reality*. IEEE Computer Society, 2006, pp. 43–50.

[10] R. C. Vanacken Lode, De Boeck Joan and C. Karin, "Using relations between concepts during interaction modelling for virtual environments," vol. Software Engineering and architectures for Realtime Interactive Systems, 2008.

[11] L. Edward, K. Amokrane, D. Lourdeaux, and J.-P. A. Barthès, "Knowledge representation - an ontology for managing a virtual environment," in *ICAART*, 2010.

[12] F. Grimaldo, M. Lozano, F. Barber, and G. Vigueras, "Simulating socially intelligent agents in semantic virtual environments," *Knowl. Eng. Rev.*, vol. 23, no. 4, pp. 369–388, 2008.

[13] Y.-L. Chu and T.-Y. Li, "Using pluggable procedures and ontology to realize semantic virtual environments 2.0," 2008.

[14] M. Kallmann and D. Thalmann, "Direct 3d interaction with smart objects," in *Proceedings of ACM VRST'99*, 1999.

[15] S. D. Marwan Badawi, "Autonomous agents interacting with their virtual environment through synoptic objects," in *The 17th Conference on Computer Animation and Social Agents*, 2004.

[16] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, pp. 34–43, 2001.

[17] D. Martin, M. Burstein, J. Hobbs, O. Lassila, D. McDermott, S. McIlraith, S. Narayanan, M. Paolucci, B. Parsia, T. Payne, E. Sirin, N. Srinivasan, and K. Sycara, "Owl-s: Semantic markup for web services," in *W3C*, ser. http://www.w3.org/Submission/OWL-S/, 2004.

Extracting and Clustering Blog Texts to Investigate Experiences of Tourists

Masato Tokuhisa[1]**, Takumi Yamamoto**[2]**, Takafumi Fukui**[2]**, Masaki Murata**[1]**, and Jin'ichi Murakami**[1]
[1]Department of Information and Electronics, Graduate School of Engineering, Tottori University
Tottori, Japan
[2]Department of Information and Knowledge Engineering, Faculty of Engineering, Tottori University
Tottori, Japan

Abstract – *In this paper, a method to support the investigation of tourist experiences and the investigation results are presented. Tour services would like to learn from a compilation of their tourists' experiences in order to improve their tours (e.g. blueberry picking tour). Since there are many accounts of tourists' experiences in blog entries, tour services investigate these accounts using those entries. The problem is the amount of text. Our proposal comprises a method for text extracting, text clustering, keyword extracting from the clusters, and emotion reasoning from such accounts. In the experiments, an analyst from the investigation uses this method to select clusters by scanning their keywords and read the texts describing positive experiences. The analyst successfully acquired a compilation of tourist experience accounts in 3 hours from 15,328 sentences found in blog entries.*

Keywords: tourism information, text mining, sentiment analysis, method to support analysts

1 Introduction

Taking a specialized tour is a good way to enjoy a vacation. Fruit picking is an example of such day tours. Since there are four seasons in Japan, tourists[1] who go fruit picking can pick seasonal fruit: blueberries in summer, pears in autumn, and strawberries in spring. However, a fruit picking day tour is so simple that tour services and fruit farms need to come up with new ideas to attract tourists.

Since there are many accounts of tourists' experiences in blog entries, those entries are worth analyzing. But the entries comprise so many sentences that the analyst cannot hope to summarize each account unaided.

In related research [1], with focusing on review documents, the combination method of sentiment analysis, text clustering, and summarization are presented. This method is capable of deriving sentiment aspects from example sentences. Such detailed information is very important for managers of hotels, restaurants, and other facilities because their services can be improved by reading these accounts.

---

[1] In this paper, the meaning of *tourist* includes *tour goer*.

For our target domain, *specialized day tours*, this method cannot be applied directly, since the reviews of day tours are less than reviews on facilities. For example, tourists enjoy not only picking fruit but also getting to and from the farm. To use blueberry picking as an example, after returning home, tourists will often make blueberry jam or give blueberries to their friends as souvenirs. The analyst should observe such behavior and use those ideas as services for the farm to offer (e.g. giving out jam recipes, packing the berries in a cute box, and so on). But, in general, the reviews on the venue or the hotel do not mention any other related activities. On the other hand, blog entries do describe those related activities. However, in blog entries, the polarity is less certain than in reviews. Furthermore, blog entries also contain sentences that do not describe the experience.

Therefore, this paper does not address reviews, but rather blog entries, and then details a method of extracting meaningful experience accounts from the entries, which infers emotions by causality and finds sentences that describe the experience.

This paper is organized as follows. Section 2 proposes our support method. Section 3 explains the analyst's role in investigating tourist experiences using our method. Section 4 presents the results as an objective evaluation. Section 5 presents the summarized experiences as a subjective evaluation of the method. Section 6 discusses the feasibility of the analysis by a beginner analyst with a slightly refined method, and Section 7 concludes.

2 Proposal of support method

2.1 Overview

To solve the problems mentioned in Section 1, this paper proposes a combination method that comprises
(a) extracting accounts of experiences,
(b) making clusters containing similar experience descriptions,
(c) making keywords to describe the cluster, and
(d) selecting experiences by the polarity inferred from the cluster (from indirect affective expressions).
Processes (a) and (c) help the analyst to remove unrelated or meaningless sentences. Processes (b) and (c) help the analyst to skim the text. Process (d) helps the analyst notice ideas for

services, like jam recipes. The analyst can come up with ideas to improve services using this method. Important sentences that stimulate ideas are manually noted and summarized as a compilation of experiences.

Figure 1 shows a work flow diagram of the method and the analyst's actions. Using general search engines, blog entries describing specialized day tours are collected. A set of experience account texts is extracted from the entries. Here, an *experience account text* consists of three sentences whose second sentence includes *experience verb* and the first/third sentences are the two sentences before and after the second (middle) sentence. Then the set of experience account texts is distributed by k-means clustering [2]. The lemmatized verbs in the text are used for the vectors. A cluster first consists of a set of experience account texts. Next, for each account text, emotions are inferred from sentences and polarity (positive/mutual/negative) is assigned to each text. For each cluster, keywords are extracted from the set of experience account texts, and keywords are assigned to each cluster.

An analyst uses these results to scan the texts for ideas that will attract tourists. At first, the analyst reads the cluster keywords. If they are promising, the analyst selects the cluster. Then the analyst scans the cluster text for positive polarity, and writes down important experience descriptions as ideas to consider for tour improvement.
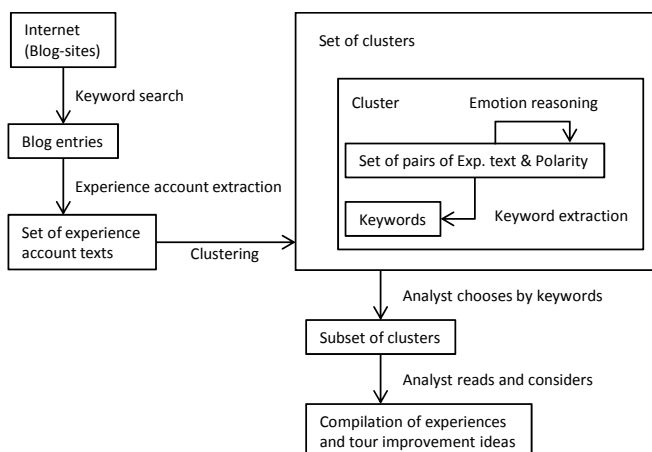
Figure 1   Work flow diagram to help analysts to find experience accounts in blog entries and to consider ideas for tour improvement.

2.2   Important processes

2.2.1   Experience account extraction

This method uses two strategies to extract experience accounts from the blog entries. One is to split an entry by experience key phrases. Another is to focus on verbs in the entry.

In a blog entry, the author of the blog sometimes writes greetings to his/her friends or a brief exposition (i.e. "I'm finished with my exams. So I…"). The purpose of this analysis is to collect the results of the experiences. Therefore,

we decided to remove the exposition[2]. Here, we divide an entry into two parts using the key phrase "blueberry picking[3]." The first part comprises the sentences from the first sentence of the entry to the two sentences preceding the sentence that contains the key phrase. The second part comprises the sentences from the sentence preceding the sentence that contains the key phrase, to the end of the entry. The first part is discarded, leaving the second part.

Next, we extract experience account texts from the second parts in the entries. We search for *experience verbs* [3] there, and then extract the sentence that contains the verb and the two sentences before and after it. Here the experience verbs have been collected by pattern matching from general blog entries during a preliminary stage. The patterns are "used to *V*," "tried to *V*," and "challenged to *V*[4]," where *V* is a variable corresponding to a verb. We collected 432 verbs from 590,000 sentences in general blogs under the condition of making 20 or more appearances there.

2.2.2   Emotion Reasoning

Emotions are inferred based on a valency pattern dictionary [4]. This dictionary consists of 14,800 records. One record contains a valency pattern and emotional attributes. A valency pattern represents some cases and one verb/adjective (as a predicate). For example, the Japanese pattern "*N*1 (the lottery/a prize) *ga N*2 (human) *ni ataru*" means "*N*2 (human) win *N*1 (the lottery/a prize)" and the Japanese pattern "*N*1 (human) *ga N*2 (a fish) *ni ataru*" means "*N*1 (human) be poisoned by *N*2 (a fish)," where *N* is a variable corresponding to a noun. Valency patterns can be used for word sense disambiguation or machine translation.

Emotional attributes are slot-value pairs in terms of emotional information. For example, "*N*2 (human) win *N*1 (the lottery/a prize)" is assigned such attributes as "feeler: *N*2, feel-to/about: *N*1, emotion: gladness, causality: acquirement" and "*N*1 (human) be poisoned by *N*2 (a fish)" is assigned such attributes as "feeler: *N*1, feel-to/about: *N*2, emotion: sadness, causality: physiological displeasure." Thus, emotions can be inferred based on the causality of emotion. This reasoning does not need direct affective expressions like "good" or "bad."

Since an experience account text contains clauses, multiple emotions can be inferred using those. The polarity of the text is calculated by averaging the emotions. This dictionary uses 9 emotions. Here, a score of *gladness*, *like*, and *expectancy* is +1, a score of *sadness*, *dislike*, *fear* and *anger* is −1, and a score of *surprise* and *calm* (*no emotion*) is 0.

---

[2] If the purpose is to motivate consumers or to phrase advertisements, the exposition might be important. Analyzing the exposition allows for improved recommendations (i.e. "If you're finished with your exams, come to a blueberry farm!").

[3] Since this research applies to Japanese blog entries, the key phrase is simple. The exact Japanese phrase was "blueberry *gari* or *tsumi*."

[4] We used "*V temiru*" or "*V temita*" (a Japanese phrase used when describing an activity tried out for the first time). The meaning of this pattern is similar to the above three patterns.

2.2.3   Keyword Extraction

The keywords of each cluster help the analyst decide whether or not the cluster is meaningful. The keywords are extracted by KeyGraph [5]. KeyGraph can extract main words and their supporting words, which represent intention of text, and represent them in a word graph. We assumed that a cluster includes some intention because the cluster consists of similar experience account texts. In addition, we already know that experience is approximately represented by verb-noun pairs [6].

Therefore, we used KeyGraph with the following style: While KeyGraph counts word frequency for each sentence, we give only nouns and verbs to KeyGraph to enhance experience-related findings. Since KeyGraph extracts important nouns and verbs from them, we expect that the analyst can imagine the experience of each cluster by reading the extracted noun-verb pairs.

3   Experience account investigation

This section explains what the analyst should do at the last two steps in Figure 1. The process is as follows.

1) The analyst chooses clusters by keywords. Experience account texts categorized as positive will appear.
2) If the cluster's keywords are extremely promising, the analyst reads the texts carefully. But if the keywords are only somewhat promising, the analyst can skim them. If the keywords are not at all promising, the analyst can discard the cluster immediately.
3) While carefully reading, if the analyst feels the cluster is not promising, the analyst can decide to skim the text. Conversely, while skimming, if the cluster seems to be promising after all, the analyst can decide to read carefully.
4) If the analyst discovers important experience accounts or is stimulated by good ideas, he/she writes them down.
5) The analyst manually sorts the important experience accounts while considering the perspective. The perspective will become the category experience accounts are sorted into. For example, place, time and persons of the experience can become perspective.

Objective evaluation of these operations is difficult, because it depends on the skill of the analyst. For objective evaluation, the next section mainly evaluates operation time and the number of sentences read by the analyst. Section 5 will then introduce the results of the compiled tourist experience accounts for subjective evaluation.

4   Experiment

4.1   Conditions

Blog entries were collected from three Japanese blog sites in 2010. The search terms were "blueberry *gari*/*tsumi* (= picking)." 642 entries (15,328 sentences) were found. The number of clusters for k-means was 24. It can be decided by the number of texts per cluster or by the clarity of keywords

which can be obtained in a practical situation. The analyst of this experiment is a linguistic expert accustomed to distinguishing sentence meaning.

4.2   Results

Table 1 shows the number of sentences after each process step and the ratio of eliminated redundant sentences. After extracting experience accounts, the number of sentences was 4,146, and this process eliminated 73.0% of redundant sentences (Step #1). After clustering and emotion reasoning, the experience account texts (3,204 sentences) remained. These texts were associated with positive emotions (Step #2). Steps #3 and #4 relied on the analyst's decisions. Since there was only one cluster with no promising keywords, the difference between Step #2 and Step #3 is small. At Step #4, the sentences skimmed by the analyst were not counted. Ultimately, the analyst carefully read 1,242 sentences. 91.9% of redundant sentences were discarded.

Table 1   Number of sentences remaining after each step and ratio of eliminated redundant sentences.

| # | Steps | Effect | |
		Sentences	Eliminated Ratio
0	Keyword search	15,328	–
1	Experience account extraction	4,146	73.0%
2	Filter by positive emotions	3,204	79.1%
3	Filter by promising keywords	3,126	79.6%
4	Filter by extremely promising keywords	1,242	91.9%

Table 2 shows the details of reading the clusters; that is, Steps #3 and #4 in Table 1. For the analyst, 1 cluster was not promising, 11 clusters were extremely promising, and 12 clusters were somewhat promising. Since cluster #1 was not promising, the analyst did not read the sentences in this cluster and its 78 sentences were discarded. 10 clusters labeled "C," comprising 1,242 sentences, were carefully read by the analyst.

The reliability of the keywords can be evaluated from the label pair of *promisingness* and *decision of how to read* in Table 2. If such a pair as (N, N), (So, Sk) and (E, C) exists, the keywords of the cluster are reliable; otherwise they are not. 21 clusters have reliable pairs as shown in the table. Therefore, the reliability of keyword extraction is 88% (= 21 / 24).

Here, some clusters are explained in detail. Cluster #1 was not promising for the analyst, because the keywords were not related to blueberry or farm, for example, "(dig, {affair, interest, shovel, sand, ...}), (put on, {plate}), ...". This cluster contained 26 texts (= 78 sentences). For example, the following sentence was one of them: "CLICK HERE to play Live Movie..." As stated above, all 26 texts (78 sentences) in cluster #1 were discarded.

Cluster #2 was extremely promising, because the keywords were farm-related; for example, "(do, {producer, famer, melon, food, blueberry, barbecue...}), ..." All 9 texts were carefully read by the analyst. The analyst extracted 4 important texts from them, noticing that tourists visited the

top of a mountain or a park near a dam and enjoyed breathing clean air while listening to the sounds of nature (e.g. cicadas chirping reminds Japanese people of summer). Guide maps may be required for visitors to improve their tours.

Cluster #4 was somewhat promising, because the keywords were similar to those of cluster #3, including "blueberry, jam." For instance, the analyst did not think this cluster contained new experience accounts. At first, the analyst skimmed the sentences of cluster #4, but then found some promising sentences and decided to read them carefully.

The clusters from #10 to #17 were skimmed. At that time, the analyst began to imagine almost whole experience. When reading cluster #23, the analyst found some new important texts to write down.

The analyst noted 219 important sentences which contained ideas for attracting blueberry picking tour goers. It took 2 hours to read and write down the texts.

Table 2 Results of reading the clusters.

#	Promisingness	Decision of How to Read	Sentences Contained	Sentences Important
1	N	N	78	0
2	E	C	27	12
3	E	C	72	21
4	So	C	378	66
5	E	C	111	21
6	E	C	129	21
7	So	Sk	93	3
8	E	C	162	21
9	E	C	78	9
10	So	Sk	375	6
11	So	Sk	51	0
12	So	Sk	363	3
13	So	Sk	285	6
14	So	Sk	90	3
15	So	Sk	111	0
16	So	Sk	81	3
17	So	Sk	93	0
18	E	C	108	3
19	E	Sk	84	0
20	So	Sk	63	0
21	E	Sk	57	3
22	So	Sk	201	3
23	E	C	48	12
24	E	C	66	3
		Total	3,204	219
	(sum if decision of how to read = C		1,242	)

N...not promising or no desire to read, E...extremely promising, So...somewhat promising, C...carefully reading, Sk...skimming

5 Compiled tourist experience accounts

5.1 Details

The analyst summarizes the 219 important sentences acquired in the previous section to create a compilation of tourists' accounts of their experiences blueberry picking. The analyst summarized 10 categories and comments as follows:

A. Companion
- cousin, grandparent, children, family

B. Transportation
- motorcycle tour
- bike
- car

C. During blueberry picking
- putting four or five blueberries into mouth at the same time to taste the mixture of sour and sweet
- selecting especially sweet berries and eating them
- tasting the difference between the various kinds of blueberries on the farm
- learning about growing blueberries, and observing the differences between berries

D. At the blueberry farm before/after picking berries
- eating blueberry ice cream
- listening to live jazz
- picking raspberries, zucchini, ears of corn, tomatoes, figs, etc.
- observing and learning about harvesting and cooking vegetables from a vegetable sommelier

E. Near the blueberry farm before/after picking berries
- making sausages
- petting animals like sheep
- letting his/her dog run free at a dog run
- holding a party at the farm
- holding a barbecue party
- having a picnic lunch under shade trees
- eating fried noodles
- eating at a restaurant
- stopping at a beer garden
- eating sweets right after lunch
- having cake at a shop connected to the farm because it was closed due to bad weather
- stopping at a fancy shop on the way home

F. Physical condition after fruit picking
- bathing at hot spring baths to wash off sweat
- sleepy

G. General experiences on the way to/from the farm
- enjoying clean air
- stopping at a mountain observatory
- stopping at a park around a dam
- seeing waterfalls
- seeing fireflies
- playing at a river
- canoeing/kayaking
- horse-riding

H. Other activities surrounding blueberry picking
- camping trip
- picnic
- staying in a cabin
- golf
- overnight training camp
- visiting parents' home

I. After returning home
- making blueberry jam, pie, tart, bread, or other pastry/dessert

- making a blueberry sour cocktail
- making a blueberry ratafia (liqueur)
- making sorbet
- freezing blueberries to preserve them
- giving blueberries to colleagues as a souvenir

J. Other

- experience becomes a memory representative of summer vacation
- meeting new people and deeping connection to them
- hunting wild blueberries
- sense that because going on the tour created the opportunity to visit other places as well, it was more reasonable than buying mail-order blueberries
- growing and picking own blueberries

Category A relates to people; Category B to transportation. Except for J, the other categories indicate spatial/temporal spread.

It took the analyst an hour to summarize the above categories. Thus, the analyst acquires this compilation of accounts of tourists' experiences in only 3 total hours from one year of blog entries using the proposed method (including 2 hours mentioned in Section 4).

5.2   Ideas for improvement

This analysis has been performed manually because careful consideration is important. Manual work ensures that the analyst can imagine many situations experienced by tourists. He can then come up with ideas for improvement.

For example, blueberry farms are located far from cities. Some people do not like such long bus rides. But if the tour service provides options for enjoyable transportation, those problems may turn into advantages. Additionally, it may be a good idea for a tour service to progressively target motorcyclist tour goers using special events like a biker meet-up or a motorcycle stamp tour that includes a blueberry farm as ways for them to enjoy getting to and from the farm. On the other hand, blog authors seem to like educational activities such as lectures about vegetables, growing blueberries, distinguishing different kinds of blueberries, and cooking/baking with blueberries. Therefore, focusing on intellectual activities can improve services at blueberry farms.

6   Discussion

An expert analyst performed the previous experiment. In this section, we evaluate whether a beginner analyst with a slightly refined method can perform as well as the expert.

6.1   Bottom-up and top-down approaches

This paper aims to help analysts to analyze experience accounts using the bottom-up approach. One of the goals of this approach is to find the categories mentioned in Section 5. At the next stage, once acquiring the categories, the top-down approach is available.

The top-down approach has two advantages. One is to reduce the cost of analysts. Another is to enable machine learning method to support analysts. After reading the categories, analysts may check whether or not each of experience account matches the categories. As the results, they will not miss the similar descriptions and will acquire accurate findings. At the same time, this work can be replaced with machine learning methods. The cooperative methods between the analyst and the machine learning tool will be required. The active learning method is one of them and can help to improve the coverage of the extraction [7].

On the other hand, the top-down approach may hide new viewpoints of analysts. Therefore, this section tries to improve the bottom-up approach method.

6.2   Another Experiment

During the process of clustering in Section 2, the vector was created with the bag-of-words model using verb, since we assumed that experience accounts would be mainly represented by verbs. The topic model LDA (latent dirichlet allocation) [8] has been discussed often recently. Thus, we also conducted another experiment that used LDA to create the vector. To do this, the topic model was gleaned from general blog entries (9,642,782 sentences, 100 topics), and then the vector of experience account text was created using topics from blueberry picking experience account texts, as estimated by the model [9]. In this case, the dimension of the vector was 100, for instance the topic number. It took 36 hours to glean the topic model and 4.5 hours to create the vector.

A beginner analyst who was an undergraduate student selected new clusters and summarized a new compilation of experience accounts. It took 4.75 hours to select the clusters and extract important texts, and then it took 2 hours to summarize the compilation by referring the first compilation mentioned in Section 5. Therefore, this experiment was a kind of combined bottom-up and top-down approach.

Table 3 shows the comparison between the two experiments. The method of verb-vector was explained in Section 2. The method of topic-vector was explained here. Despite being done by a beginner, the results of the analysis were acceptable. The effect of LDA was confirmed with the number of clusters discarded because of a lack of promising keywords, which totaled 3 clusters (255 sentences). This was more than in the experiment in Section 4. But this analyst read

Table 3   Comparison of remaining sentences between verb-vector method and topic-vector method.

#	Steps	Method	
		Verb-vector	*Topic-vector*
3	Filter by promising keywords	23 [clusters] 3,126[sentences] 79.6%	21[clusters] 2,949[sentences] 80.8%
4	Filter by extremely promising keywords	11 [clusters] 1,242 [sentences] 91.9%	9 [clusters] 1,992 [sentences] 87.0%

more texts than the first analyst in Section 4, because the selected clusters contained more sentences.

Since top-down analysis reduced the need for the analyst to consider category in the compilation, the analyst could take time to contemplate the experience account sentences. The beginner analyst wrote down more important sentences than the expert. Some were new, as follows:

Category A :
- friends from a club

Category C :
- taking photos

Category G :
- stopping at a lavender field
- stopping at a farmer's market
- collecting beetles
- going swimming at a pool
- seeing fireworks

Category H :
- visiting the farm as part of a bus tour

Category I :
- eating yogurt with blueberry jam
- selecting berries for eating or for making jam, depending on size and ripeness

These are 10 additional items. The number of the items shown in Section 5 was 51. Therefore, we estimate the coverage of the first experiment at 84% (= 51 / (51 + 10)). In order to improve this coverage, machine learning tool will be helpful as mentioned in Section 6.1.

We confirmed that top-down knowledge helps a beginner analyst come up with ideas, and LDA increases clustering accuracy. These results mean that our method (including the refined method) is feasible for supporting even beginner analysts.

7 Conclusions

This paper proposed a method to help analysts learn from blog entries about the experiences of tourists who go on specialized day tour (e.g., blueberry picking). The method comprises the following: extracting experience account texts, classifying them, inferring the emotion of the texts, and assigning keywords for each cluster. The analyst reads the cluster keywords and decides how carefully to read the cluster texts. In the experiment, an expert analyst came up with ideas and summarized a compilation of accounts of experiences from 15,328 blog entry sentences in 3 hours using this method. Here, the ratio of eliminated redundant sentences was 91.9%, and the reliability of keywords for each cluster was 88%. As a result, 10 categories were defined based on important perspective and used to order the compilation of blueberry picking experience account. The coverage of the experience accounts was estimated to be 84%. Further, we confirmed the feasibility of the categories and our method in helping a beginner analyst come up with ideas. Thus, this paper proposed a successful method to help analysts investigate tourist experiences.

References

[1] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. A. Reis, and J. Reynar: "Building a Sentiment Summarizer for Local Service Reviews", *Proc. of WWW Workshop on NLP Challenges in the Information Explosion Era* (NLPIX), 2008.

[2] Bayon - a simple and fast clustering tool - Google Project Hosting, http://code.google.com/p/bayon/

[3] K. Ikeda, K. Tanabe, and H. Okuda: "Using Experience Expressions to Extract Experience Information from Blogs", *Proc. of Data Engineering Workshop* (DEW2007), A8-1, 2007.

[4] M. Tokuhisa, T. Tanaka, S. Ikehara, and J. Murakami: "Emotion Reasoning Based on Valency Patterns - a prototype annotation of causal relationships", *Proc. of the 4th International Symposium on Human and Artificial Intelligence Systems* (HART2004), vol.1, pp.534-539, 2004.

[5] Y. Ohsawa, N. E. Benson, and M. Yachida: "KeyGraph: Automatic Indexing by Segmenting and Unifing Co-occurrence Graphs", *IEICE Trans.on Information and Systems*, vol.J82-D-1, no.2, pp.391-400, 1999.

[6] T. Kurashima, T. Tezuka, and K. Tanaka: "Mining and Visualizing Local Experiences from Blog Entries", *Database and Expert Systems Applications*, *Proc. of* DEXA 2006, LNCS 4080, pp.213-222, Springer, 2006.

[7] M. Tokuhisa, H. Shahana, M. Murata, J. Murakami: "An Active Learning based Support Tool for Extracting Hints of Tourism Development from Blog Articles", *Proc. of International Conference on Advanced Applied Informatics* (IIAI2012), pp.103-107, 2012.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan: "Latent Dirichlet Allocation", *Machine Learning Research*, vol.3, pp.993-1022, 2003.

[9] Yahoo_LDA, http://github.com/shravanmn/Yahoo_LDA

Program Completion as Constraint Satisfaction: Tight Logic Programs Revisited

Yun Bai and Yan Zhang

Artificial Intelligence Research Group

School of Computing, Engineering and Mathematics

University of Western Sydney

Locked Bag 1797, Penrith, NSW 2751, Australia

Email: ybai@scem.uws.edu.au, yan@scem.uws.edu.au

Abstract—**Research in logic programming shows an increasing interest in studying tight logic programs because as Fages proved, each stable model of a tight logic program is identical to a logic model of a corresponding propositional theory (called the Clark's completion of the program), and *vice versa*. Therefore, any algorithms for solving the satisfiability problem may be used to compute stable models of tight logic programs. Furthermore, it has been also observed that many important problems can be encoded into tight logic programs. However, it is still unclear whether we can give a better characterization on the tractability of tight logic programs although some obvious tractable subclass is easy to be recognized. In this paper, we investigate the computational complexity of propositional tight logic programs under stable model semantics. In particular, we provide explicit syntactic characterizations for various tractable subclasses of tight logic programs. Our approach is to transform the completions of tight logic programs to instances of the constraint satisfaction problem (CSP), and then apply Schaefer's Dichotomy Theorem to obtain sufficient tractability conditions for tight logic programs. Based on this approach, we further specify an interesting subclass of tight logic programs whose computational tractability may be decided through their decompositions.**

Key words: Knowledge representation, Automated problem solving, Reasoning strategies, Logic programming, Nonmonotonic logic [1]

I. Introduction

Research in logic programming shows an increasing interest in studying tight logic programs (e.g. [1], [2], [9], [15], [17]) because as Fages proved, each stable model of a tight logic program is identical to a logic model of a corresponding propositional theory (called the Clark's completion of the program), and *vice versa*. Therefore, any algorithms for solving the satisfiability problem may be used to compute stable models of tight logic programs [1], [6]. Furthermore, it has been also observed that many important problems can be encoded into tight logic programs.

However, the computational tractability property of tight logic programs has yet been thoroughly studied. Specifically, it is still not clear whether we can provide some syntactic but rather general characterizations for tractable subclasses of tight logic programs. In this paper, we investigate the computational

complexity of propositional tight logic programs under stable model semantics. In particular, we provide explicit syntactic characterizations for various tractable subclasses of tight logic programs. Our approach is to construct a polynomial transformation from the completions of tight logic programs to instances of the constraint satisfaction problem (CSP), and then apply Schaefer's Dichotomy Theorem to obtain sufficient tractability conditions for tight logic programs. Based on this approach, we further specify an interesting subclass of tight logic programs whose computational tractability may be decided through their decompositions. As will be shown in the paper, our tractability results are rather general and cover a wide range of tight logic programs, which may provide useful guidelines to implement efficient algorithms for computing tight logic programs.

The paper is organized as follows. Section 2 introduces some preliminary logic concepts and notations we need in this paper. Section 3 describes a polynomial transformation from the completion of a tight logic program to a Boolean CSP-instance. Section 4 gives the main tractability result of this paper and provides a formal proof. Section 5 specifies the class of decomposable tight logic programs and depicts how these programs' tractability may be recognized through their decompositions. Finally, section 6 discusses the related work and concludes the paper with some remarks.

II. Preliminaries

The relationship between Clark's completion semantics and stable model semantics has been intensively studied by researchers, e.g. [2], [9], [10], [12]. It has been showed that by compiling a logic program into a classical propositional theory - the completion of this program, each stable model of the program is a logical model of the theory, and under certain condition called *tightness*, each logical model of the propositional theory is also a stable model of this program. We are interested in the computational issue of this type of logic programs because it has been observed that many important problems can be encoded into tight logic programs.

We consider finite propositional normal logic programs in which each rule has the form:

$$a \leftarrow b_1, \cdots, b_m, not\ c_1, \cdots, not\ c_n \qquad (1)$$

where a is either a propositional atom or empty, $b_1, \cdots, b_m$, $c_1, \cdots, c_n$ are propositional atoms. When a is empty, rule (1)

is called a *constraint*. Given a rule r of the form (1), we denote $head(r) = a$, $pos(r) = \{b_1, \cdots, b_m\}$ and $neg(r) = \{c_1, \cdots, c_n\}$, and therefore, rule (1) may be represented as the form:

$$head(r) \leftarrow pos(r), not\ neg(r). \qquad (2)$$

We also use $Atom(\Pi)$ to denote the set of all propositional atoms occurring in program Π.

The stable model of a program Π is defined as follows. Firstly, we consider Π to be the program in which each rule does not contain negation as failure sign *not*. A finite set S of propositional atoms is called a *stable model* of Π if S is the smallest set such that for each rule $a \leftarrow b_1, \cdots, b_m$ from Π, if $b_1, \cdots, b_m \in S$, then $a \in S$. Now let Π be an arbitrary normal logic program. For any set S of propositional atoms, program Π^S is obtained from Π by deleting (1) each rule from Π that contains *not c* in the body and $c \in S$; and (2) all forms of *not c* in the bodies of the remaining rules[2]. Then S is a stable model of Π if and only if S is a stable model of Π^S. It is easy to see that a program may have one, more than one, or no stable models at all.

Definition 1: [3] Given a logic program Π consisting of rules of the form (1). Its *completion* $Comp(\Pi)$ is obtained in three steps:

- Step 1: Replace each rule of the form (1) with the formula:
  $b_1 \wedge \cdots \wedge b_m \wedge \neg c_1 \wedge \cdots \wedge \neg c_n \supset a$.
  We may use notion $body \supset a$ to represent this formula.
- Step 2: For each symbol a (a is not empty), let $Support(a)$ denote the set of formulas obtained from all rules with a in the head as showed in Step 1. Suppose $Support(a)$ is the set:
  $body_1 \supset a$,
  $\cdots$,
  $body_k \supset a$.
  Replace this set with a single formula
  $a \equiv body_1 \vee \cdots \vee body_k{}^3$.
  If $Support(a) = \emptyset$ then replace it by $\neg a$.
- Step 3: For each constraint in Π: $\leftarrow b_1, \cdots, b_m, not\ c_1, \cdots, not\ c_n$, replace it with a formula $Cons : \neg b_1 \vee \cdots \vee \neg b_m \vee c_1 \vee \cdots \vee c_n$.

Then $Comp(\Pi)$ is the set of all such $Support(a)$ and $Cons$.

Gelfond and Lifschitz showed that given a program Π, each stable model of Π is also a (minimal) model of $Comp(\Pi)$ [12]. However, the converse is usually not true. Consider program Π consisting of a single rule: $p \leftarrow p$. Clearly, $Comp(\Pi) = \{p \equiv p\}$, which has two models $\emptyset$ and $\{p\}$. But program Π only has one stable model $\emptyset$. Fages defined a syntactic condition on logic programs that ensures the equivalence between stable model semantics and completion semantics. This is so called *tightness* of a program.

---

[2] We call Π^S is the result of Gelfond-Lifschitz transformation on Π with S.

[3] If some $body_i \equiv \mathtt{T}$, then $Support(a) = a$.

Definition 2: Given a program Π. Π is *tight* if there exists a function λ from the set of all propositional atoms to ordinals such that for each rule (1) in Π,

$$\lambda(a) > \lambda(b_i) \text{ for all } i\ (1 \le i \le m).$$

Fages showed that for any tight logic program Π, a set of propositional atoms S is a stable model of Π if and only if S is a model of $Comp(\Pi)$ [10]. Immediately from this statement, we have the following result.

Proposition 1: A tight logic program Π has a stable model if and only if $Comp(\Pi)$ is satisfiable.

Let S be a set. The *size* of S, denoted as $|S|$, is defined to be the cardinality of S. Then since a program Π is a finite set of rules, the size of Π, i.e. $|\Pi|$, is the number of rules in Π. It is well known that for a given set S of propositional atoms and a logic program Π, deciding whether S is a stable model of Π can be solved in polynomial time in terms of the size of S and Π. But deciding whether a logic program has a stable model is NP-complete [18], which implies that in practice it is unlikely to implement a general algorithm to compute stable models of logic programs polynomially. However, we may identify tractable subclasses of tight logic programs by investigating the tractable computation of the models of their program completions as the following proposition states.

Proposition 2: Deciding whether a tight logic program Π has a stable model can be achieved in polynomial time if deciding the satisfiability of $Comp(\Pi)$ can be achieved in polynomial time.

III. PROGRAM COMPLETION AS CONSTRAINT SATISFACTION

Given a logic program, we can view the satisfiability of its completion as a *constraint satisfaction problem* (CSP). Formally, an *instance* of CSP (or say a CSP-instance) is a triple (V, D, C), consisting of a finite set V of variables, a finite domain D of values, and a set C of *constraints* (t, R) where t is a tuple $t = (x_1, \cdots, x_m)$ of variables for some m and R is a relation on D of arity m. Sometimes, we also call R a relation in the CSP-instance (V, D, C). A *solution* to a CSP-instance is a mapping $h : V \to D$ such that for every constraint $(t, R) \in C$, $h(t) = (h(x_1), \cdots, h(x_m)) \in R$.

The classical Boolean satisfiability problem SAT can be viewed as a CSP-instance. For example, given a CNF formula $\varphi = (p \vee q) \wedge (q \vee r \vee \neg s)$, we transform it to a CSP-instance (V, D, C), where $V = \{p, q, r, s\}$, $D = \{0, 1\}$ and C consists of two constraints $((p, q), \{0, 1\}^2 - \{(0, 0)\})$ and $((q, r, s), \{0, 1\}^3 - \{(0, 0, 1)\})$. In this CSP-instance, for each constraint $(t, R) \in C$, relation R is a subset of $\{0, 1\}^{|t|}$, and we call it *logical relation*[4]. A CSP-instance of this form is also called a *Boolean* CSP-instance. Given a logic program Π, we can always represent its completion $Comp(\Pi)$ as a CNF formula, and then transform this CNF formula to a Boolean CSP-instance, as shown by the following example.

Example 1: Consider program Π consisting of the following rules:

---

[4] $|t|$ denotes the cardinality of the set of variables occurring in tuple t.

$$p \leftarrow,$$
$$q \leftarrow,$$
$$r \leftarrow p, not\ s,$$
$$r \leftarrow q, not\ t.$$

It is easy to see that $Comp(\Pi) = \{p, q, \neg s, \neg t, r \equiv ((p \wedge \neg s) \vee (q \wedge \neg t))\}$, which is equivalent to the following set of clauses:

$$p,$$
$$q,$$
$$\neg s,$$
$$\neg t,$$
$$r \vee \neg p \vee s,$$
$$r \vee \neg q \vee t,$$
$$\neg r \vee p \vee q,$$
$$\neg r \vee p \vee \neg t,$$
$$\neg r \vee q \vee \neg s,$$
$$\neg r \vee \neg s \vee \neg t.$$

Then the corresponding CSP-instance is: (V, D, C), where $V = \{p, q, r, s, t\}$, $V = \{0, 1\}$, and C consists of the following constraints:

$$(\{p\}, \{(1)\}), \quad (\{r, q, t\}, \{0, 1\}^3 - \{(0, 1, 0)\}),$$
$$(\{q\}, \{(1)\}), \quad (\{r, p, q\}, \{0, 1\}^3 - \{(1, 0, 0)\}),$$
$$(\{s\}, \{(0)\}), \quad (\{r, p, t\}, \{0, 1\}^3 - \{(1, 0, 1)\}),$$
$$(\{t\}, \{(0)\}), \quad (\{r, q, s\}, \{0, 1\}^3 - \{(1, 0, 1)\}),$$
$$(\{r, p, s\}, \{0, 1\}^3 - \{(0, 1, 0)\}),$$
$$(\{r, s, t\}, \{0, 1\}^3 - \{(1, 1, 1)\}).$$

Given a logic program Π, we call the CSP-instance constructed as described in Example 1 the Π's *Boolean CSP-instance*, denoted as $CSP(\Pi)$. Then we have the following proposition.

Proposition 3: A tight logic program Π has a stable model if and only if Π's Boolean CSP-instance $CSP(\Pi)$ has a solution.

From previous discussion, we observe that for a given clause $l_1 \vee \cdots \vee l_k$, the logical relation in the corresponding constraint has a size of 2^k. Then it seems that encoding an arbitrary CNF formula into a CSP-instance may exponentially depend on the length of clauses in the CNF formula. We should note that this actually is not a barrier of using CSP techniques to study satisfiability problem because for each fixed k, transforming a CNF formula, in which the length of its each clause is bounded by k, into a Boolean CSP-instance takes polynomial time as illustrated in [7], [14].

Similarly, to transform $Comp(\Pi)$ into $CSP(\Pi)$ polynomially, we may fix the underlying logic program's rule length and support depth. Formally, consider a logic program Π consisting of rules of the form (1), we define the *rule length* of rule r to be the number of atoms occurring in r, i.e. $rl(r) = 1 + m + n$. For atom $a \in Atom(\Pi)$, if its support $Support(a)$ in Π is formula $a \equiv body_1 \vee \cdots \vee body_k$, then we define the *support depth* of a to be the number of $body$s occurring in this formula, i.e. $sd(a) = k$. Now we define the rule length and support depth of the program Π as follows:

$$rl(\Pi) = Max(rl(r) : r \in \Pi),$$
$$sd(\Pi) = Max(sd(a) : a \in Atom(\Pi)).$$

Theorem 1: Let Π be a logic program with fixed rule length l and support depth d. Π's Boolean CSP-instance

$CSP(\Pi)$ can be constructed in at most $\mathcal{O}((d2^l + l^d 2^{d+1})|\Pi|)$ steps.

Proof: For each formula in $Comp(\Pi)$, it has one of the three forms: $a, \neg a$ or $a \equiv body_1 \vee \cdots \vee body_k$, where $k \leq d$, and each $body$ is of the form $b_1 \wedge \cdots \wedge b_m \wedge \neg c_1 \wedge \neg c_n$, where $(1 + m + n) \leq l$. It is easy to see that $|Comp(\Pi)| \leq l|\Pi|$ and $Comp(\Pi)$ contains at most $|\Pi|$ formulas of the form $a \equiv body_1 \vee \cdots \vee body_k$. This formula is equivalent to the following two formulas:

$$a \vee (\neg body_1 \wedge \cdots \wedge \neg body_k),\ and$$
$$\neg a \vee body_1 \vee \cdots \vee body_k.$$

The first formula above can induce k clauses each with a length of at most l. Then each relation in $CSP(\Pi)$ constructed from these clauses needs at most 2^l steps. Therefore, at most constructing all relations in $CSP(\Pi)$ from clauses induced from $a \vee (\neg body_1 \wedge \cdots \wedge \neg body_k)$ needs at most $k2^l \leq d2^l$ steps.

On the other hand, it is observed that formula $\neg a \vee body_1 \vee \cdots \vee body_k$ can at most induce l^k clauses and each with a length of $k + 1$. So constructing relations of $CSP(\Pi)$ from these clauses needs at most $l^k 2^{k+1} \leq l^d 2^{d+1}$ steps. As $Comp(\Pi)$ contains at most $|\Pi|$ formulas of the form $a \equiv body_1 \vee \cdots \vee body_k$, altogether, constructing relations of $CSP(\Pi)$ from all clauses induced from $Comp(\Pi)$ needs at most $(d2^l + l^d 2^{d+1})|\Pi|$ steps. Also note that all other formulas in $Comp(\Pi)$ are of the form a or $\neg a$, and at most the number of these formulas in $Comp(\Pi)$ is $(l-1)|\Pi|$ which is less than $(l^d 2^{d+1})|\Pi|$ when $d \geq 1$. So given a logic program Π with the fixed rule length l and support depth d, we can construct $CSP(\Pi)$ in at most $\mathcal{O}((d2^l + l^d 2^{d+1})|\Pi|)$ steps. ∎

We should note that in practice, constructing the Boolean CSP-instance of a given logic program needs much less steps than what Theorem 1 states, because in a program, the rule length for most rules is usually quite small, and the support depth for most atoms occurring in the program is just 1! This has been observed from many classical problem domains (e.g. N-queen problem and Hamiltonian Circuit problem) and practical applications like robotic planning studied in logic programming [3]. In the rest of the paper, without explicit declaration, we will consider all logic programs with fixed rule length and support depth. The following propositions illustrate several interesting properties of $CSP(\Pi)$ constructed from a given Π.

Proposition 4: Let Π be a logic program with $rl(\Pi) = l$ and $sd(\Pi) = d$ where $l > d$. Then every relation in $CSP(\Pi)$ has at most arity l.

Proposition 5: Let Π be a logic program with $rl(\Pi) = l$ and $sd(\Pi) = d$ where $d > l$. Then every relation in $CSP(\Pi)$ has at most arity $(d + 1)$.

Proposition 6: Let Π be a logic program. If for each $a \in Atom(\Pi)$, there is a rule $r \in \Pi$ with $head(r) = a$, for any $r \in \Pi$ $rl(r) = l$, and $sd(\Pi) = 1$, then every relation in $CSP(\Pi)$ has exact arity l.

IV. TRACTABLE SUBCLASSES OF TIGHT LOGIC PROGRAMS

A. The Result

We first present the main characterization result for the computational tractability of tight logic programs.

Theorem 2: Let Π be an arbitrary tight logic program. Deciding whether Π has a stable model is solvable in polynomial time if one of the following five conditions is satisfied. In this case, we also call Π a *tractable tight logic program.*

1) For each rule $r \in \Pi$, $pos(r) \neq \emptyset$;
2) For each $a \in Atom(\Pi)$, there is a rule $r \in \Pi$ such that $head(r) = a$, and there exists at least one rule r' with $head(r') = a$ and $neg(r') = \emptyset$; for each rule $r \in \Pi$ with $head(r) = \emptyset$ (i.e. r is a constraint), $neg(r) \neq \emptyset$;
3) For each $a \in Atom(\Pi)$, there is a rule $r \in \Pi$ such that $head(r) = a$, $|pos(r) \cup neg(r)| = 1$ and $sd(a) = 1$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|pos(r) \cup neg(r)| = 2$;
4) For each rule $r \in \Pi$ with $head(r) \neq \emptyset$, $neg(r) = \emptyset$ and $|pos(r)| \leq 1$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|pos(r)| \leq 1$;
5) For each rule $r \in \Pi$ with $head(r) \neq \emptyset$, either $pos(r) \cup neg(r) = \emptyset$, or $sd(head(r)) = 1$ and $neg(r) = \emptyset$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|neg(r)| \leq 1$.

Example 2: Consider the following two tight logic programs:

Π_1: $\qquad\qquad\qquad$ Π_2:
$\quad \leftarrow a, not\ b, \qquad\qquad \leftarrow d, not\ a,$
$\quad b \leftarrow, \qquad\qquad\qquad\quad a \leftarrow,$
$\quad c \leftarrow, \qquad\qquad\qquad\quad b \leftarrow,$
$\quad d \leftarrow b, \qquad\qquad\qquad c \leftarrow a, b.$
$\quad e \leftarrow c.$

It is easy to check that Π_1 satisfies condition 4 and Π_2 satisfies condition 5 in Theorem 2 respectively. In section 5, we will illustrate other program examples that satisfy conditions 1, 2, and 3 in Theorem 2.

B. The Proof of Theorem 2

As we have showed in last section, given a logic program Π with fixed rule length l and support depth d, it can be transformed to a Boolean CSP-instance $CSP(\Pi)$ in polynomial time. Then if Π is a tight logic program, and deciding whether $CSP(\Pi)$ has a solution is solvable in polynomial time, it follows that deciding whether Π has a stable model is also solvable in polynomial time. Furthermore, each solution of $CSP(\Pi)$ (if there is one) is also a stable model of Π. The constraint satisfaction problem we considered for program completion is also called the *generalized satisfiability problem* for which Schaefer provided a complete characterization on its computational complexity [19]. Before we present Schaefer's Dichotomy Theorem, we first introduce following useful concepts.

Let R be a k-ary logical relation. R is *0-valid* if tuple $(0, \cdots, 0) \in R$. R is *1-valid* if tuple $(1, \cdots, 1) \in R$. R

is *bijunctive* if R is the set of truth assignments satisfying some 2CNF formula. R is *weakly positive* (or *weakly negative*, resp.) if R is the set of truth assignments satisfying some CNF formula having at most one negative (or positive, resp.) variable in each conjunct. R is *affine* if R is the set of solutions to a system of linear equations over the two-element field. That is, R is equivalent to a conjunction of formulas of the forms $x_1' \oplus \cdots \oplus x_m' = 0$ and $x_1'' \oplus \cdots \oplus x_n'' = 1$, where $\{x_1', \cdots, x_m', x_1'', \cdots, x_n''\} \subseteq \{x_1, \cdots, x_k\}$, and $\oplus$ denotes addition modulo 2.

Theorem 3: (Schaefer's Dichotomy Theorem [19]) Let $(V, \{0,1\}, C)$ be a Boolean CSP-instance. Deciding whether $(V, \{0,1\}, C)$ has a solution is solvable in polynomial time if one of the following six conditions is satisfied, otherwise it is NP-complete.

(a) Every logical relation R in $(V, \{0,1\}, C)$ is 0-valid;
(b) Every logical relation R in $(V, \{0,1\}, C)$ is 1-valid;
(c) Every logical relation R in $(V, \{0,1\}, C)$ is bijunctive;
(d) Every logical relation R in $(V, \{0,1\}, C)$ is weakly positive;
(e) Every logical relation R in $(V, \{0,1\}, C)$ is weakly negative;
(f) Every logical relation R in $(V, \{0,1\}, C)$ is affine.

Now we consider how this theorem is used to prove Theorem 2. Basically, we will show that a tight logic program Π satisfying one of conditions 1-5 in Theorem 2 will imply that for $CSP(\Pi)$, one of conditions (a)-(e) in Schaefer's Dichotomy Theorem (Theorem 3) also holds accordingly, where condition (f) in Theorem 3 will never hold for $CSP(\Pi)$.

Lemma 1: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0,1\}, C)$. Every logical relation R in $CSP(\Pi)$ is 0-valid if and only if for each rule $r \in \Pi$, $pos(r) \neq \emptyset$.

Proof: Suppose every relation R in $CSP(\Pi)$ is 0-valid. From the construction of $CSP(\Pi)$, we know that each k-ary relation R in $CSP(\Pi)$ is constructed from a clause of the form $l_1 \vee \cdots \vee l_k$, where l_i is either a variable in V or the negation of a variable in V. Note that R is formed as $\{0,1\}^k - \{X\}$, where $X = (\epsilon_1, \cdots, \epsilon_k)$, $\epsilon_i \in \{0,1\}$ and is an assignment satisfying $\overline{l_1} \wedge \cdots \wedge \overline{l_k}$. Since R is 0-valid, it means that $X \neq (0, \cdots, 0)$. Consequently, it implies that $l_1 \vee \cdots \vee l_k$ is *not* of the form $x_1 \vee \cdots \vee x_k$ where each $x_i \in V$. Without loss of generality, we assume that clause $l_1 \vee \cdots \vee l_k$ is induced from $Support(a)$ for some atom $a \in Atom(\Pi)$. Clearly, $Support(a)$ is of one of the three forms: a (i.e. a rule of the form $a \leftarrow$), $\neg a$ (i.e. no rule with a as the head), or $a \equiv body_1 \vee \cdots \vee body_h$. The first case is impossible since this implies that R is not 0-valid. The second case implies the result to be trivially true. Now consider $Support(a) = a \equiv body_1 \vee \cdots \vee body_h$, which is equivalent to the following two formulas:

$\neg a \vee body_1 \vee \cdots \vee body_h$, and
$a \vee (\neg body_1 \wedge \cdots \wedge \neg body_h)$,

where each $body$ is of the form $b_1 \wedge \cdots \wedge b_m \wedge \neg c_1 \wedge \cdots \wedge c_n$. It is easy to see that each clause induced from the first formula above will contain $\neg a$, and consequently, the relation constructed from this clause is 0-valid. Now consider clauses

induced from the second formula above. It is observed that each clause is of the form: $a \vee \neg b_1 \vee \cdots \vee \neg b_m \vee c_1 \vee \cdots \vee c_n$. As the relation in $CSP(\Pi)$ constructed from this clause contains tuple $(0, \cdots, 0)$, it implies at least one of b_i $(1 \leq i \leq m)$ should be in the clause $a \vee \neg b_1 \vee \cdots \vee \neg b_m \vee c_1 \vee \cdots \vee c_n$. This means $pos(r) \neq \emptyset$.

Now we assume that clause $l_1 \vee \cdots \vee l_k$ is induced from some constraint in Π:
$\leftarrow b_1, \cdots, b_n, not\ c_1, \cdots, not\ c_n$. Then the clause is of the form $\neg b_1 \vee \cdots \vee \neg b_m \vee c_1 \vee \cdots \vee c_n$. Clearly, this formula is not of the form $x_1 \vee \cdots \vee x_k$ where each $x_i \in V$. This implies $pos(r) \neq \emptyset$. Combining these two cases, it concludes that to ensure every relation R in $CSP(\Pi)$ is 0-valid, for each rule $r \in \Pi$, it must be the case $pos(r) \neq \emptyset$.

The other direction can be proved in a similar way. ∎

Lemma 2: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0, 1\}, C)$. Every logical relation R in $CSP(\Pi)$ is 1-valid if and only if for each $a \in Atom(\Pi)$ there is a rule $r \in \Pi$ such that $head(r) = a$, and there exists at least one rule $r' \in \Pi$ with $head(r') = a$ and $neg(r') = \emptyset$; for each rule $r \in \Pi$ with $head(r) = \emptyset$ (i.e. r is a constraint), $neg(r) \neq \emptyset$.

Lemma 3: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0, 1\}, C)$. Every logical relation R in $CSP(\Pi)$ is bijunctive if and only for each $a \in Atom(\Pi)$, there is a rule $r \in \Pi$ such that $head(r) = a$, $|pos(r) \cup neg(r)| = 1$ and $sd(a) = 1$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|pos(r) \cup neg(r)| = 2$.

Proof: We only prove one direction, while the other direction is proved in a similar way. Suppose each R in $CSP(\Pi)$ is bijunctive. That is, R is constructed from a clause $l_1 \vee l_2$ from either formula $Support(a)$ for some $a \in Atom(\Pi)$ or $Cons$ for some constraint in Π. We first consider the case of $Support(a)$. This implies that $Support(a)$ is not of the form $\neg a$ or a. So it concludes that for each $a \in Atom(\Pi)$, there is a rule $r : a \leftarrow pos(r), not\ neg(r)$, where $pos(r) \cup neg(r) \neq \emptyset$. Also, it must be the case $|pos(r) \cup neg(r)| = 1$. Now assume $sd(a) > 1$. For instance, suppose $sd(a) = 2$. Then we have $Support(a) = a \equiv l \vee l'$, where l and l' are atoms or negative atoms. Obviously, from $a \equiv l \vee l'$, clause $\neg a \vee l \vee l'$ is induced. This contradicts the fact that all relations in $CSP(\Pi)$ are bijunctive.

Now suppose that R is constructed from a constraint $r :\leftarrow pos(r), not\ neg(r)$. Obviously, the fact that R is bijunctive implies that $|pos(r) \cup neg(r)| = 2$ ∎

Lemma 4: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0, 1\}, C)$. Every logical relation R in $CSP(\Pi)$ is weakly positive if and only if for each rule $r \in \Pi$ with $head(r) \neq \emptyset$, $neg(r) = \emptyset$ and $|pos(r)| \leq 1$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|pos(r)| \leq 1$.

Lemma 5: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0, 1\}, C)$. Every logical relation R in $CSP(\Pi)$ is weakly negative if and only if for each rule $r \in \Pi$ with $head(r) \neq \emptyset$, either $pos(r) \cup neg(r) = \emptyset$, or $sl(head(r)) = 1$ and $neg(r) = \emptyset$; for each rule $r \in \Pi$ with $head(r) = \emptyset$, $|neg(r)| \leq 1$.

Lemma 6: Given a tight logic program Π and its corresponding $CSP(\Pi) = (V, \{0, 1\}, C)$. Then every logical relation R in $CSP(\Pi)$ is not affine.

Proof: From [19], it shows that the cardinality of an affine relation is always a power of 2. However, from the construction of $CSP(\Pi)$, we know that each relation R in $CSP(\Pi)$ is formed as $\{0, 1\}^{|t|} - \{(\epsilon_1, \cdots, \epsilon_{|t|})\}$ where $|t| \geq 1$ and $\epsilon_i \in \{0, 1\}$, which has a cardinality of an odd number. ∎

Proof of Theorem 2 We only need to show that conditions 1-5 in Theorem 2 implies conditions (a) - (e) in Schaefer's Dichotomy Theorem when the program is (polynomially) transformed to the corresponding Boolean CSP-instance. Lemmas 1- 5 have shown such equivalence between conditions (a), (b), (c) (d), (e) in Schaefer's Dichotomy Theorem and conditions 1, 2, 3, 4, and 5 in Theorem 2 respectively. Lemma 6, on the other hand, shows that for any program Π, every relation in $CSP(\Pi)$ is not affine, and hence condition (f) in Schaefer's Dichotomy Theorem is not applicable in our case. ∎

V. DECOMPOSABLE TIGHT LOGIC PROGRAMS

It is important to realize that the completeness feature of Schaefer's Dichotomy Theorem does not apply to Theorem 2. This is because our polynomial transformation from a tight logic program to its corresponding Boolean CSP-instance only ensures the program's tractability *if* solving the corresponding Boolean CSP-instance is tractable. Indeed, it is not difficult to find a subclass of tight logic programs which do not satisfy any of conditions 1-5 but are still tractable. Consider the following tight logic program:

Π:
$a \leftarrow,$
$b \leftarrow,$
$c \leftarrow a,$
$c \leftarrow b, d.$

Clearly, this program does not satisfy any of conditions 1-5 in Theorem 2, but since it is a Horn program, its model can be computed in polynomial time.

Nevertheless, under certain conditions, we still can recognize a tight logic program's tractability by using Theorem 2 even if this program does not satisfy any of conditions 1-5 in Theorem 2. To illustrate our idea, let us examine the above Horn program Π again. It is easy to observe that Π *almost* satisfies condition 5 in Theorem 2 except $sd(c) = 2 > 1$. However, we can actually *decompose* Π into two small programs:

Π_1: Π_2:
   $a \leftarrow,$ $a \leftarrow,$
   $b \leftarrow,$ $b \leftarrow,$
   $c \leftarrow a,$ $c \leftarrow b, d.$

Now note that both Π_1 and Π_2 satisfy condition 5 in Theorem 2. Furthermore, it is also observed that Π's model can be obtained from some standard set operation on Π_1's and Π_2's models, (i.e. $\{a, b, c\} = \{a, b, c\} \cup \{a, b\}$). It is easy to prove that a subclass of tight logic programs like this type are still tractable. This example motivates us to specify a more general

subclass of tight logic programs whose tractability may be *indirectly* decided by applying Theorem 2.

Definition 3: Let Π be an arbitrary tight logic program. Π is *decomposable* if Π can be polynomially decomposed to a sequence of tight logic programs $(\Pi_1, \cdots, \Pi_k)$ for some constant $k > 1$, such that $\Pi = \bigcup_{i=1}^{k} \Pi_i$ and Π has a stable model if and only if each Π_i $(1 \leq i \leq k)$ has a stable model. In this case, $(\Pi_1, \cdots, \Pi_k)$ is called a *decomposition* of Π.

Directly from Definition 3 and Theorem 2, we can prove the following result.

Theorem 4: Suppose that Π is a decomposable tight logic program and has a decomposition $(\Pi_1, \cdots, \Pi_k)$. Then Π is a tractable program if each Π_i $(1 \leq i \leq k)$ satisfies one of conditions 1-5 in Theorem 2.

Now an immediate question is: under what conditions, a tight logic program is decomposable and how to obtain a decomposition of this program? Intuitively, if a program is decomposable, it may have more than one decompositions. As a trivial example, we can see that both $(\{a \leftarrow, b \leftarrow\}, \{c \leftarrow\})$ and $(\{a \leftarrow\}, \{b \leftarrow\}, \{c \leftarrow\})$ are decompositions for program $\{a \leftarrow, b \leftarrow, c \leftarrow\}$. However, what our interest here is to study those decompositions that may contain useful information for deciding the tractability of the original program, e.g. as Theorem 4 indicates, whereas we fail to do so via Theorem 2 directly.

For this purpose, we first define the *dependency graph* for a program. Given a program Π, its *dependency graph*, denoted as $G(\Pi)$, is a directed graph $(Atom(\Pi), E)$, where $Atom(\Pi)$ is the set of vertices, and E is the set of edges. An edge $(a, b) \in E$ iff there is a rule $r \in \Pi$ such that $a \in pos(r) \cup neg(r)$ and $b = head(r)$. Edge (a, b) is labeled "positive" $(+)$ if $a \in pos(r)$ and "negative" $(-)$ if $a \in neg(r)$. We say that atom a is *positive* in $G(\Pi)$ if for any path starting from a, no negative edge is contained in the path.

We say that $G(\Pi)$ contains a *dependency subgraph* of Π, if there is a subgraph in $G(\Pi)$: (V_1, E_1) where $V_1 \subseteq Atom(\Pi)$ and $E_1 \subseteq E$, such that there does not exist an edge $(a, b) \in E$ with $a \in V_1$ and $b \in (Atom(\Pi) - V_1)$, or $a \in (Atom(\Pi) - V_1)$ and $b \in V_1$ (note that for each edge $(a, b) \in E_1$, $a, b \in V_1$). From this definition, we can see that Π's dependency subgraph actually provides a partition on Π where all rules related to the dependency subgraph are *self-contained* in the subgraph with no connection to other rules in Π. That is, for some $\Pi_1 \subseteq \Pi$, $V_1 = Atom(\Pi_1)$ and $Atom(\Pi_1) \cap Atom(\Pi - \Pi_1) = \emptyset$. Therefore, we can represent a dependency subgraph of Π as $G(\Pi_1) = (Atom(\Pi_1), E_1)$, where $\Pi_1 \subseteq \Pi$. Now we have the following general result.

Theorem 5: Let Π be a tight logic program and $G(\Pi)$ the dependency graph of Π. If $G(\Pi)$ contains dependency subgraphs $G(\Pi_1), \cdots, G(\Pi_k)$ where $\Pi = \bigcup_{i=1}^{k} \Pi_i$ $(k > 1)$, then $(\Pi_1, \cdots, \Pi_k)$ is a decomposition of Π.

Example 3: Consider the following tight logic program:

Π:
$\quad r_1 : a \leftarrow b, not\ c,$
$\quad r_2 : d \leftarrow not\ e,$
$\quad r_3 : d \leftarrow f$
$\quad r_4 : e \leftarrow,$
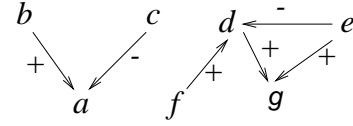
Fig. 1. Π's dependency graph $G(\Pi)$.

$\quad r_5 : f \leftarrow,$
$\quad r_6 : g \leftarrow d, e.$

It is easy to check that Π does not satisfy any of conditions 1-5 in Theorem 2. Π has the following dependency graph $G(\Pi)$.

Clearly $G(\Pi)$ contains two dependency subgraphs of Π: $G(\Pi_1)$ and $G(\Pi_2)$ where $\Pi_1 = \{r_1\}$ and $\Pi_2 = \{r_2,, r_3, r_4, r_5, r_6\}$. So according to Theorem 5, (Π_1, Π_2) is a decomposition of Π. Also note that Π_1 and Π_2 satisfy conditions 1 and 2 in Theorem 2 respectively.

The following theorem provides another way to decompose a program which is especially useful to decide the tractability of a program that has a support depth greater than 1.

Theorem 6: Let Π be a tight logic program and there is an atom $a \in Atom(\Pi)$ such that $sd(a) = d > 1$, that is Π contains rules $r_1, \cdots, r_d$ with $head(r_i) = a$ $(1 \leq i \leq d)$. If a is positive in Π's dependency graph $G(\Pi)$, then Π has a decomposition $(\Pi_1, \cdots, \Pi_d)$, where each Π_i $(1 \leq i \leq d)$ is formed as $\Pi_i = \Pi - \{r_j : a \leftarrow pos(r_j), not\ neg(r_j) \mid j \neq i\}$.

Example 4: Consider the following tight logic program:

Π:
$\quad r_1 : a \leftarrow b,$
$\quad r_2 : a \leftarrow c,$
$\quad r_3 : b \leftarrow not\ c,$
$\quad r_4 : c \leftarrow not\ b,$
$\quad r_5 : d \leftarrow a.$

Again, we observe that Π does not satisfy any of conditions 1-5 in Theorem 2. However, since $sd(a) = 2 > 1$, $head(r_1) = head(r_2) = a$, and atom a is positive in Π's dependency graph $G(\Pi)$, according to Theorem 6, (Π_1, Π_2) is a decomposition of Π, where $\Pi_1 = \{r_1, r_3, r_4, r_5\}$ and $\Pi_2 = \{r_2, r_3, r_4, r_5\}$. Then it is easy to see that both Π_1 and Π_2 satisfy condition 3 in Theorem 2.

Corollary 1: Every tight Horn logic program Π with $sd(\Pi) > 1$ is decomposable and has a decomposition where each program in the decomposition satisfies condition 5 in Theorem 2.

VI.   RELATED WORK AND CONCLUSION

Since Marek and Truszczyński proved that deciding whether a logic program has a stable model is NP-complete [18], the study on the computational issues of logic programs with stable model semantics has focused on two different methods: One is to investigate proper conditions under which the procedure of computing stable models may be optimized. For instance, various splitting techniques [16], [11]. The other approach is to identify tractable subclasses of logic programs so that efficient algorithms can be specifically developed for computing stable models of these tractable programs. In this paper, we followed the second approach to study the

tractability of tight logic programs by using CSP techniques. An interesting feature of our tractability characterizations is that they are purely syntax-based, and it only takes linear time to check whether a tigh logic program satisfies one of these syntactic forms (see Theorem 2).

The idea of using CSP techniques to study the tractability of logic programs was previously considered by Ben-Eliyahu and Dechter. In [4], Ben-Eliyahu and Dechter studied the tractability of head cycle-free extended disjunctive logic programs (HEDLPs) and defined two tractable subclasses of HEDLPs which correspond to two tractable subsets of CSP. Basically, Ben-Eliyahu and Dechter showed that the stable model (answer set) existence of an HEDLP can be decided in polynomial time with respect to the *clique width* and *cycle-cutset*'s cardinality of the HEDLP associated *interaction graph* respectively[5]. But these tractable subclasses are somehow not easy to be recognized because finding the clique width in a graph and finding the minimum size of cycle-cutset in a graph are NP-hard [7].

Ben-Eliyahu also proposed a tractable hierarchy for computing stable models [5]. In particular, Ben-Eliyahu presented a sequence of classes of programs $\Omega_1, \Omega_2, \cdots$ such that if a program Π belongs to class Ω_k, then Π at most has k stable models and each of them can be found in time polynomial in the size of Π and k. Further, Ben-Eliyahu proved that for an arbitrary program Π, we can find the minimum k where Π belongs to Ω_k in time polynomial in the size of Π. However, we observe that in general these results do not have much help for achieving the tractable computation of the stable models of logic programs because for a given class Ω_k, k may be of the exponential size of programs belonging to Ω_k, and computing stable models of programs in Ω_k may still remain intractable in terms of the size of these programs.

On the other hand, Gottlob, Scarcello and Sideri recently also studied the tractability of logic programs from a fixed parameter complexity perspective [13]. They showed that all related computational problems of logic programs (including the stable model existence problem) are fixed-parameter tractable with respect to the *feedback width* of the undirected dependency graph of the logic program. However, this approach also has some restrictions. For example, it was observed that positive programs with large feedback width are not recognized to be tractable although they may be clearly tractable, and it is not known yet whether computing feedback vertex sets of size k is fixed-parameter tractable for *directed graph* [13].

Our work presented in this paper can be extended in several directions. Firstly, we may extend our results to other logic programs. For instance, we may investigate whether our tractability results can be improved to cover *generalized tight logic programs* which were studied by researchers from various aspects [2], [9], [17]. Also, instead of considering the completion of a logic program, we may define an alternative equivalent propositional theory for a large class of logic programs (e.g. [4]), and then obtain similar syntactic tractability characterizations by studying the tractable subclasses of the satisfiability problem through CSP techniques. The other

direction is to identify more decomposable logic programs and extend this technique to first-order answer set programs [11], which will eventually simplify the computation of the underlying programs.

REFERENCES

[1] V. Asuncion, F. Lin, Y. Zhang and Y. Zhou, Ordered completion for first-order logic programs on finite structures. *Artificial Intelligence* **177-179** (2012) 1-24.

[2] Y. Babovich, E. Erdem and V. Lifschitz, Fages' theorem and answer set programming. In *Proceedings of the 8th International Workshop on Non-Monotonic Reasoning*, 2000.

[3] C. Baral, *Knowledge Representation, Reasoning and Declarative Problem Solving*, MIT Press, 2003.

[4] R. Ben-Eliyahu and R. Dechter, Propositional semantics for disjunctive logic programs. *Annals of Mathematics and Artificial Intelligence* **12** (1994) 53-87.

[5] R. Ben-Eliyahu, A hierarchy of tractable subsets for computing stable models. *Journal of Artificial Intelligence Research* **5** (1996) 27-52.

[6] Y. Chen, F. Lin, Y. Zhang, Y. Zhou, Loop-separable programs and their first-order definability. *Artificial Intelligence* **175** (2011) 890-913.

[7] R. Dechter, *Constraint Processing*. Morgan Kaufmann Publishers. 2003.

[8] R.G. Downey and M.R. Fellows, *Parameterized Complexity*. Springer 1999.

[9] E. Erdem and V. Lifschitz, Tight logic programs. *Theory and Practice of Logic Programming*, to appear.

[10] F. Fages, Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic and Computer Science* **1** (1994) 51-60.

[11] P. Ferraris, J. Lee, V. Lifschitz and R. Palla, Symmetric splitting in the general theory of stable models. In *Proceedings of IJCAI-2009*, pp 797-803, 2009.

[12] Gelfond, M. and Lifschitz, V., The stable model semantics for logic programming. In *Proceedings of the Fifth Joint International Conference and Symposium*, pp 1070-1080. MIT Press, 1988.

[13] G. Gottlob, F. Scarcello and M. Sideri, Fixed-parameter complexity in AI and nonmonotonic reasoning. *Artificial Intelligence* **138** (2002) 55-86.

[14] P. Kolaitis, Constraint satisfaction, database, and logic. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-2003)*, pp 1587-1595, 2003.

[15] V. Lifschitz, Foundation of logic programming. *Principles of Knowledge Representation*, G. Brewka ed., 69-128. 1996.

[16] V. Lifschitz and H. Turner, Splitting a logic program. In *Proceedings of the 11th International Conference on Logic Programming (ICLP-94)*, pp 23-37. MIT Press, 1994.

[17] F. Lin and J. Zhao, On tight logic programs and yet another translation from normal logic programs to propositional logic. In *Proceedings of IJCAI-2003*, pp 853-858. 2003.

[18] V.M. Marek and M. Truszczyński, Autoepistemic logic. *Journal of Association of Computing Machinery* **38**(3) (1991) 588-619.

[19] T.J. Schaefer, The complexity of satisfiability problems. In *Proceedings of the 10th ACM Symposium on Theory of Computing*, pp 216-226, 1978.

---

[5]Readers are referred to [4], [7] for the concepts of interaction graph, clique width, and cycle-cutset.

Hierarchical Classification Using FCA and the Cosine Similarity Function

Mauri Ferrandin[1]**, Júlio César Nievola**[1]**, Fabrício Enembreck**[1]**,**
Edson Emílio Scalabrin[1]**, Kelvin Vieira Kredens**[1] **and Bráulio Coelho Ávila**[1]
{maurif,nievola,fabricio,scalabrin,kelvin,avila}@ppgia.pucpr.br
[1]Graduate Program in Computer Science
Pontifícia Universidade Católica do Paraná
Curitiba, Paraná, Brazil - +55 41 3271 1669

Abstract—Classification is a common task in Machine Learning and Data Mining. Some classification problems need to take into account a hierarchical taxonomy establishing a order between involved classes and are called hierarchical classification problems. The protein function prediction can be treated as a classification problem. As protein's functions may be arranged in a hierarchy of classes, predicting the protein function is treated as a problem of hierarchical classification. This paper presents an algorithm for hierarchical classification using the global approach, called HMCS-FCA-SC for hierarchical multi-label classification. The proposed algorithm is a improved version of HMCS-FCA based on FCA techniques and it was evaluated in eight datasets based on EC and GPCR and compared with Clus-HMC. Preliminary results show that the HMCS-FCA-SC is one alternative that can be used for hierarchical classification.

Keywords: hierarchical classification, global classifier, formal concept analysis

I. INTRODUCTION

One of the most important problems in Machine Learning and Data Mining is the classification that consists of associate one or more classes from a set of predefined classes to an example from the database. The features of each example determines the classes it will be associated with. Some classification problems need to deal with hierarchical relations represented by a taxonomy established between the classes to be predicted, this kind of problems are named hierarchical classification.

A hierarchical taxonomy can be defined as an ordering of elements according to their function or importance and can be represented by a tree or a Directed Acyclic Graph (DAG). The predictive performance tends to decrease with increasing depth (specificity), whereas the amount of specific examples is smaller, making harder the training process and the prediction of the classes for a sample.

Three main features differentiate the hierarchical problems: (i) the kind of hierarchical taxonomy - a tree or a DAG; (ii) how deep the classification in the hierarchy is performed - restricted to the leaf nodes or in any level of the taxonomy;

and (iii) how the hierarchical structure is explored - with a local or global approach [1].

Based on this wide diversity of problems, specific algorithms for hierarchical classification are being developed. This paper presents an algorithm for hierarchical classification that uses techniques of Formal Conceptual Analysis (FCA) combined with the cosine similarity function to create a model that is used to hierarchical classification. Experiments with biological data sets were done and the obtained results were compared with the Clus-HMC [2] that represents the state of art in terms of hierarchical classification.

II. BASIC CONCEPTS

A. Formal Concept Analysis

The FCA deals with binary data tables describing relationship between objects and attributes, named formal contexts. According to [3], a formal context $K := (G, M, I)$ consists of two sets G and M and a relation I between G and M. The elements of G are the objects and the elements of M are the attributes of the context. To express that an object g is in relation I with an attribute m, we write gIm or $(g, m) \in I$ and read it as "the object g has the attribute m". Table I shows a formal context with $G = \{1, 2, 3, 4\}$, $M = \{a, b, c, d\}$ and $gIm = \{(1, b), (1, c), (2, a), (2, c), (2, d), (3, a), (3, b), (4, b), (4, c), (4, d)\}$.

Table I: Sample of a formal context.

.

	a	b	c	d
1	0	1	1	0
2	1	0	1	1
3	1	1	0	0
4	0	1	1	1

Given a formal context $K := (G, M, I)$, for a set $A \subseteq G$ of objects we define the set of attributes common to the objects in A named A' as shown in Equation 1.

Correspondingly, for a set B of attributes we define the set of objects common to the attributes in B named B' as shown in Equation 2. We also represent the composite operation $(A')'$ as A'' what means compute these two operations in sequence using as input to the second operation the results from the first one. The result of A'' represents the closure of A. The same can be done with the objects where B'' represents the closure of B.

$$A' := \{m \in M | gIm \; for \; all \; g \in A\}. \tag{1}$$

$$B' := \{g \in G | gIm \; for \; all \; m \in B\}. \tag{2}$$

Using the definitions (1) and (2) and given a formal context $K := (G, M, I)$ we can compute it's formal concepts. A formal concept of K is a pair (A, B) where:

$$A \subseteq G, \; B \subseteq M, \; A' = B \; and \; B' = A. \tag{3}$$

We name the set A as extent and B as intent of the formal concept (A, B). The set of all formal concepts of the formal context K is denoted by $C(G, M, I)$. Thus, a concept is identified by its extent (containing all objects that belongs to the concept) and its intent (containing all attributes shared by the objects of the context)[4]. For instance, from the Table I the concept $(14, bc)$ indicates that the objects $\{1, 4\}$ shares the attributes $\{b, c\}$, and to simplify the notation in samples we represented the sets in the formal concept $(\{1, 4\}, \{b, c\},)$ just as $(14, bc)$.

If (A_1, B_1) and (A_2, B_2) are concepts in $C(G, M, I)$, we say that (A_1, B_1) is a subconcept of a (A_2, B_2), or that (A_2, B_2) is a superconcept of (A_1, B_1) and we write $(A_1, B_1) \leqslant (A_2, B_2)$, if $A_1 \subseteq A_2$ (which is equivalent to $B_2 \subseteq B_1$). We can also say that (A_1, B_1) is smaller than (A_2, B_2), or that (A_2, B_2) is larger than (A_1, B_1). The relation $\leqslant$ is called hierarchical order (or simply order) of the concepts. The set of all concepts of a formal context $K := (G, M, I)$ ordered is denoted by $\mathfrak{B}(G, M, I)$ and is called concept lattice of the formal context (G, M, I) [4]. Figure 1 shows the complete lattice for the formal context in Table I in which every node represents a formal concept and the connections between nodes express the relations among them.

Regarding the algorithms for calculating the concepts of a formal context, there are currently a lot of proposals, like the most basic algorithms such as Naive and Intersections demonstrated in [4]. Also, the approaches proposed by [5] as the Next Closure until today perform well. With the growth of the databases size, newer and faster algorithms were proposed to deal with this scenario and also use parallel processing. [6],[7] and [8] presents recently approaches with better performance and parallel processing support in order to compute the formal concepts for big contexts. For more information about algorithms to compute formal concepts see [9] and [10].
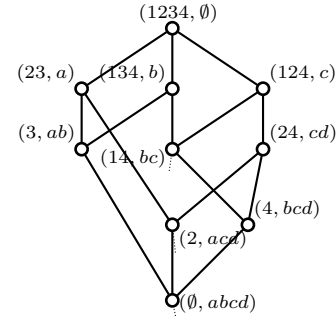
Figure 1: Concept lattice of the formal context in Table I.

B. Adding external knowledge to a context

In many applications to facilitate analysis and improve results in the data exploration by the use of lattices we can add external knowledge to them, such as information about a taxonomy. To better demonstrate how this knowledge addition can be done, let's look at the example proposed by [4]. Table II presents a context representing a list of documents $(d_1, d_2, ..., d_6)$, represented as objects of the context, and set of terms containing the subjects in the documents represented as attributes and composed by Artificial Intelligence (AI), Expert Systems (ES), Information Retrieval (IR), Cataloguing (Cat), Indexing (Ind), Information Sciences (IS), Information Retrieval Systems (IRS) and Knowledge Based Systems (KBS). In the context, e.g., the document d_1 is related to AI, ES and IR. Figure II shows the lattice for the Table II.

Table II: Context representing documents and the terms.

	AI	ES	IR	Cat	Ind	IS	IRS	KBS
d_1	1	1	1	0	0	0	0	0
d_2	1	1	0	1	0	0	0	0
d_3	1	1	0	0	1	0	0	0
d_4	1	1	0	0	0	1	0	0
d_5	1	1	0	1	0	0	1	0
d_6	1	0	1	0	0	0	0	1
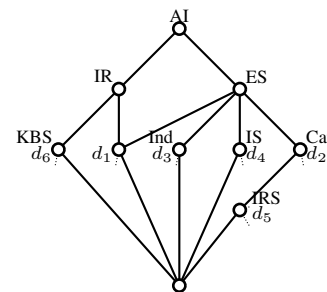
Figure 2: Concept lattice of the context in the Table II

Suppose now that there is a predefined organization between the terms we associate with documents, i.e., a

taxonomy that represents a hierarchical organization of terms associated with objects in Table II. Figure 3 shows an example of taxonomy for the terms associated with the documents in Table II. According to this taxonomy, a document having the term IR, also possesses the terms IS and CA. Its also important to note that some terms used in the initial context are not present in the taxonomy, e.g., the term AI that will be kept unchanged in the augmented context.
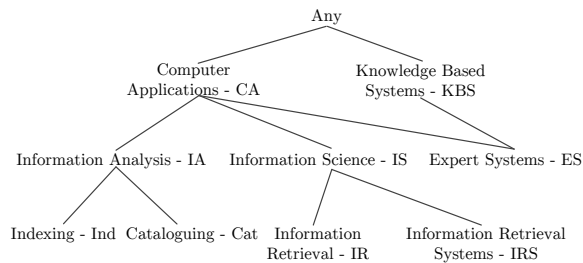

Figure 3: Graph representing the documents taxonomy

Taking into account the taxonomy shown in Figure 3, the information represented in the hierarchical taxonomy is mapped to the original context of Table II resulting in a new context that represents the correlations between documents and terms respecting the hierarchical taxonomy of terms as shown in Table III.

Table III: Formal context from Table II augmented.

	AI	ES	IR	Cat	Ind	IS	IRS	KBS	CA	IA
d_1	1	1	1	0	0	1	0	1	1	0
d_2	1	1	0	1	0	0	0	1	1	1
d_3	1	1	0	0	1	0	0	1	1	1
d_4	1	1	0	0	0	1	0	1	1	0
d_5	1	1	0	1	0	1	1	1	1	1
d_6	1	0	1	0	0	1	0	1	1	0

After constructing the augmented context we compute the lattice with the information to obtain a representation of correlations between documents and terms, obeying a preset taxonomy that represents the hierarchy between terms. The lattice is shown in Figure 4, where is possible to see that the document d_1, firstly in Figure 2 associated with the terms IR, ES and AI, now after the imposition the hierarchy is also linked to terms IS, CA and KBS.

There are a lot of works related to the use of addition of background knowledge to a context for data representation and exploration, among then we highlight [11], [12] and [13].

In this section we addressed the basic notions of the formal concept analysis. More detailed content about the foundations of FCA can be found in [3] and [4].
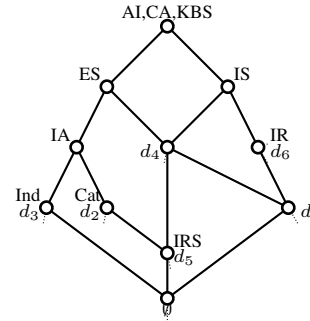
Figure 4: Concept lattice of context in the Table III.

III. RELATED WORKS

Hierarchical classification is an approach that has been widely used in text mining since the 90s. Among the works in this context can be cited [14], [15], [16], [17]. The field of bioinformatics presents several problems to be solved by hierarchical classification, but it is, unfortunately, still little explored. Some works have been published using this approach, specifically in the protein function prediction, but restricted to a tree structured taxonomy [18], [19], [20], [21], [22], [23].

In the work developed by [23], [24] and [25] the authors dismissed the class hierarchy, i.e., the concept of ancestor and descendant was not applied. Thus, one can say that the problem of hierarchical classification was transformed into flat classification problem, where the conventional classification algorithms are used.

Local binary classifiers for each node were created by [26] and [27]. Although the set of binary classifiers represent the hierarchy of classes, there is no guarantee that they produce consistent results. Moreover, if an error occurs in a class at a certain level, it can be propagated to the classes of the levels below. Thus, one can say that the results presented by the authors are questionable because the approach has these drawbacks.

A hierarchical classification model named Clus for the DAG structure using the global or big-bang approach was developed in [2]. In this work the authors discuss three kinds of classification: single-label classification (SC), hierarchical single-label classification (HSC) and multi-label hierarchical classification (HMC). For the development of these classifiers the authors used the induction of decision trees and showed how this model can be modified for use in hierarchical DAG structures.

There are a lot of proposes using different techniques to hierarchical classification, like [28]: a model named Multi-Label Hierarchical Classification with an Artificial Immune System (MHC-AIS) that generates rules IF-THEN and two main versions were presented exploring both local and global approaches; [29]: a method that is an extension of the flat

classification algorithm naive Bayes adapted to hierarchical classification and [30]: optimization based in an ant colony to predict classes in problems of hierarchical classification.

Recently, [31]: search through greedy algorithms to find the best sub graph of the tree or DAG to classify examples; [32]: a method based on the k-Nearest Neighbours algorithm; [33]: an algorithm based on the global approach using a competitive artificial neural network for the protein function prediction datasets; and [34]: an adapted Learning Classifier Systems (LCS) in order to predict protein functions described in the GO format, called HLCS (Hierarchical Learning Classifier System) builds a global classifier to predict all classes in the application domain and its is expressed as a set of IF-THEN classification rules.

Regarding the use of FCA to data mining tasks, there are lot of proposed works, although not for hierarchical classification problems, we can highlight for future research the algorithms Grand [35], Rulearner [36], Legal [37], [38], Galois [39], [40], Similares1 e Similares2 [41], [42]. An algorithm for extracting association rules in the presence of a taxonomy using FCA was proposed by[43]. In that work rules are found through a filter mechanism for determining whether a sample is positive or negative. The survey by [44] presents a study involving 702 publications which used FCA in KDD in between the years 2003 and 2009.

IV. PROPOSED ALGORITHM

The central idea of the proposed algorithm named Hierarchical Multi-label Classifier System - Formal Concept Analysis with Similarity Cosine (HMCS-FCA-SC) is to use the FCA techniques to create a model for hierarchical classification, having as base the information present in the formal concepts obtained and the cosine similarity function. The last was used to choose the formal concepts that will determine the predicted classes to a new instance not submitted to the model during the training phase. The main steps are shown in Algorithm 1.

The first step consists of to take a training partition composed by a set of attributes and transform it in a formal context (line 1). To achieve this goal, the attributes of the instance are mapped to attributes of the formal context. If the dataset contains continuous attributes a discretization algorithm is applied to convert them into a set of nominal ones. The instances are mapped as objects of the context.

The next step is to add the information about all classes considering the taxonomy through the process of addition of background knowledge to a formal context (line 2). This process is done as represented in the Section II-B and must keep the associations between the the instances represented and its original classes.

Following, the formal formal concepts of the augmented context are computed using any traditional algorithm for this purpose (line 3). After this step, every formal concept found will contain in it's intent a set of attributes and classes, e.g.,

---

Algorithm 1 HMCS-FCA-SC

Require: The sets of instances for: training TR, testing TE; The taxonomy of classes H; a parameter k;
1: Create a formal context CT with TR;
2: Add information from H to CT;
3: Compute the set FC of formal concepts from CT;
4: **foreach** ($t_i \in TE$) **do**
5:     Find a subset x_i from FC with k formal concepts more similar to the instance t_i;
6:     Sort x_i by similarity with t_i in a decreasing order;
7: **end for**
8: **for** $z = 1$ to k **do**
9:     Predict for every test instance t_i the classes present in the z first formal concepts of it's respective set x_i;
10:     Compute hP and hR ;
11: **end for**

$\{a_1, a_2, a_3, r, c_1, c_2\}$ where a_n are attributes, r is the root class of the hierarchy and c_n are the classes. Also, every formal concept will contain in it's extent the set of objects (that represents the instances from the training partition) which contains the attributes and classes in the intent, e.g., $\{o_1, o_2, o_3, o_4\}$ in the extent, where o_n is an object identifier, indicates that the sample intent represented earlier is present in those objects.

The next step is to find for every instance in the testing partition a set with k formal concepts that will be used to classify the instance, this selection is done through a cosine similarity function (lines 4 and 5). In this step the attributes of the instance being tested (t_i) and the attributes present in the formal concept (cf_j) are mapped to a vectorial space and normalized. Then using the cosine similarity function represented in the Equation 4 a similarity between every instance being tested and every formal concept is computed. For every instance (t_i) being tested a set (x_i) containing k selected concepts are stored and sorted by decreasing order of similarity.

$$cos(t_i, fc_j) = \frac{t_i \cdot fc_j}{||t_i|| \, ||fc_j||} \qquad (4)$$

The last step consists in evaluate the classifier hits and misses using the evaluation metric (for loop in the line 8). For evaluate the results of the proposed algorithm we used a similar approach to the used by Clus-HMC [2] that uses as base two hierarchical measures named hierarchical precision and recall, and a threshold chosen by the user ranging from 0 to 1 that produces different values for the precision and recall, allowing the user to select by changing the threshold results with best values for precision or recall. To obtain a unique measure considering the results of all operation points of the classifier, the precision-recall curves are drawn with the precision and recall points and the area

under precision-recall curve is computed.

In the proposed algorithm we used the interval from 1 to k represented as z as threshold for produce results with more precision or recall. To achieve this we take the instances of the test partition with its ordered set of selected concepts and assign for every instance the classes found in the first z formal concepts of the ordered set (line 9). Using this approach we obtain (normally) more precision when the value of z is near to 1 and more recall when it's value is increased. For each value of z (line 10), the final values for precision and for recall are obtained by Equation 5, according to the proposal of [45]. In Equation 5, according to the author we assume that one instance i belongs to a set of classes C and will have a set of predicted classes denoted by C'. The extended sets $\hat{C}$ and $\hat{C}'$ represent respectively the classes in the sets C and C' with the addition of all ancestors classes of each set considering the taxonomy.

$$hP = \frac{\sum_i |\hat{C}_i \cap \hat{C}'_i|}{\sum_i |\hat{C}'_i|}, \quad hR = \frac{\sum_i |\hat{C}_i \cap \hat{C}'_i|}{\sum_i |\hat{C}_i|} \quad (5)$$

After obtaining all precision-recall pairs for different thresholds we can represent them by a precision-recall curve and calculate the area under the curve to obtain a single measure of performance of the classifier considering all operation points. The points at the extremes were added by interpolation process to obtain a complete curve and the method used is the same proposed by [46] equivalent to the method used in [2]. The Area Under the Precision-Recall Curve (AUPRC) obtained is used as a unique measure of the performance of the classifier taking into account that 1 is the result of a classifier with 100 % accuracy.

V. EXPERIMENTS

A. Datasets

The two databases used in this article involve the families of G-Protein Coupled Receptor (GPCR) and Enzymes. The protein functional classes are given unique hierarchical indexes by [47] in the case of GPCRs and by Enzyme Commission Codes [48] in the case of enzymes. These databases were used in the work of [29] and are available at https://sites.google.com/site/carlossillajr/resources. Enzymes are catalysts that accelerate chemical reactions while GPCRs are proteins involved in signaling and are particularly important in medical applications as it is believed that from 40% to 50% of current medical drugs target GPCR activity [49] Each dataset has four different versions based on different kinds of predictor attributes, and in each dataset the classes to be predicted are hierarchical protein functions. Each type of binary predictor attribute indicates whether or not a "protein signature" (or motif) occurs in a protein [29]. The motifs used in this work were: Interpro Entries, FingerPrints from the Prints database, Prosite Patterns and Pfam. Apart from the presence/absence of several motifs according to the

signature method, each protein has two additional attributes: the molecular weight and the sequence length.

Before performing the experiments, the following preprocessing steps were applied to the datasets: (i) Every class with fewer than 10 examples was recursively merged with its parent classes until to have at least 10 examples or to be labeled to the Root class. (ii) All examples whose most specific class was the root class were removed. (iii) A class blind discretization algorithm based on equal-frequency binning (using 20 bins) was applied to the molecular weight and sequence length attributes, which were the only two continuous attributes in each dataset. Table IV shows the main characteristics of datasets after the preprocessing steps which are detailed in [29]. In all datasets, each protein (example) is assigned at least to one class at each level of the hierarchy.

Table IV: Enzime and GPCR dataset main characteristics.

Protein Type	Signature	Attributes	Examples	Classes/Level
Enzyme	Interpro	1,216	14,027	6/41/96/187
	Pfam	708	13,987	6/41/96/190
	Prints	382	14,025	6/45/92/208
	Prosite	585	14,041	6/42/89/187
GPCR	Interpro	450	7,444	12/54/82/50
	Pfam	75	7,053	12/52/79/49
	Prints	283	5,404	8/46/76/49
	Prosite	129	6,246	9/50/79/49

B. Results

The experiments were done using a simple dataset split considering 2/3 of the exemples for training and 1/3 for testing. The k parameter was set to 100. Figure 5 and Figure 6 show the obtained results for GPCR and EC datasets respectively comparing them with the Clus-HMC [2] results.

Statistical tests based on Wilcoxon Rank-Sum comparing the results extracted from the precision-recall curve of each classifier with $\alpha = 0.05$ demonstrated that there is no significant statistical differences between the results of proposed method named HMCS-FCA-SC and Clus-HMC.

VI. CONCLUSION

This paper has presented a new algorithm named HMCS-FCA-SC for the hierarchical classification problem of predicting protein functions supporting taxonomies organized as a tree or DAG. The algorithm proposed is based on FCA techniques combined with the use of the cosine similarity function.

This classification approach has the advantage of considering all hierarchy in the prediction process (global approach), works with both types of taxonomies: tree or DAG, and can make multi-label predictions, even though the last feature
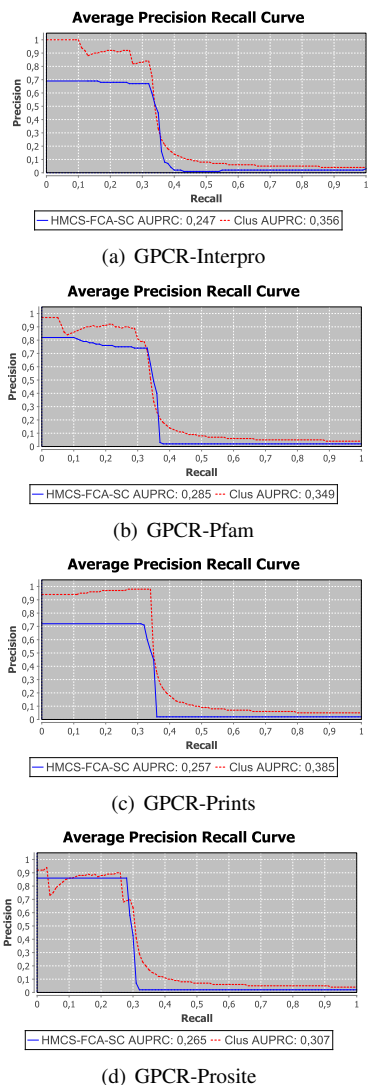
(a) GPCR-Interpro

(b) GPCR-Pfam

(c) GPCR-Prints

(d) GPCR-Prosite

Figure 5: HMCS-FCA-SC x Clus-HMC - GPCR Datasets.

(a) EC-Interpro

(b) EC-Pfam

(c) EC-Prints

(d) EC-Prosite

Figure 6: HMCS-FCA-SC x Clus-HMC - EC Datasets.

was not explored in this work. The main disadvantage is the processing time due to the exponential behaviour - in the worst case - of the FCA techniques.

The HMCS-FCA-SC was applied to 8 datasets from GPCR and Enzimes. The results of predictions were measured using the hierarchical precision and recall, and the precision-recall curves to evaluate the classifier along its different operation points.

The use of some not commonly used techniques for classification in the development of the proposed algorithm was quite challenging. The results of performed experiments are promising and some improvements will be done in future works.

REFERENCES

[1] C. N. Silla, Jr. and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Min. Knowl. Discov.*, vol. 22, pp. 31–72, January 2011.

[2] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Mach. Learn.*, vol. 73, pp. 185–214, November 2008.

[3] B. Ganter and R. Wille, *Formal concept analysis - mathematical foundations*. Springer, 1999.

[4] C. Carpineto and G. Romano, *Formal concept analysis - mathematical foundationsConcept Data Analysis: Theory and Applications*. Wiley, 2004.

[5] B. Ganter, "Two basic algorithms in concept analysis," TH Darmstadt, FB4–Preprint 831, 1984.

[6] P. Krajca and V. Vychodil, "Distributed algorithm for computing formal concepts using map-reduce framework," in *Proceedings of the 8th International Symposium on Intelligent Data Analysis: Advances in Intelligent Data Analysis VIII*, ser. IDA '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 333–344.

[7] P. Krajca, J. Outrata, and V. Vychodil, "Advances in algorithms based on cbo," in *CLA*, 2010, pp. 325–337.

[8] S. Andrews, "In-close, a fast algorithm for computing formal concepts," in *International Conference on Conceptual Structures (ICCS)*, January 2009. [Online]. Available: http://shura.shu.ac.uk/38/

[9] S. O. Kuznetsov and S. Obiedkov, "Comparing performance of algorithms for generating concept lattices," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 14, pp. 189–216, 2002.

[10] F. Strok and A. Neznanov, "Comparing and analyzing the computational complexity of FCA algorithms," in *Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists*, ser. SAICSIT '10. New York, NY, USA: ACM, 2010, pp. 417–420.

[11] B. Ganter, "Attribute exploration with background knowledge," *Theor. Comput. Sci.*, vol. 217, no. 2, pp. 215–233, 1999.

[12] R. Belohlavek and V. Vychodil, "Formal concept analysis with background knowledge: attribute priorities," *Trans. Sys. Man Cyber Part C*, vol. 39, pp. 399–409, July 2009.

[13] R. Belohlávek and V. Vychodil, "Background knowledge in formal concept analysis: constraints via closure operators," in *SAC*, 2010, pp. 1113–1114.

[14] D. Koller and M. Sahami, "Hierarchically classifying documents using very few words," in *Proceedings of the Fourteenth International Conference on Machine Learning*, ser. ICML '97. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, pp. 170–178.

[15] A. Sun and E.-P. Lim, "Hierarchical text classification and evaluation," in *Proceedings of the 2001 IEEE International Conference on Data Mining*, ser. ICDM '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 521–528.

[16] A. Sun, E. peng Lim, and W. keong Ng, "Performance measurement framework for hierarchical text classification," *Journal of the American Society for Information Science and Technology*, vol. 54, pp. 1014–1028, 2003.

[17] S. Kiritchenko, S. Matwin, R. Nock, and A. F. Famili, "Learning and evaluation in the presence of class hierarchies: Application to text categorization," in *Canadian Conference on AI*, 2006, pp. 395–406.

[18] H. Blockeel, M. Bruynooghe, S. Dzeroski, J. Ramon, and J. Struyf, "Hierarchical multi-classification," in *MRDM02*, S. Džeroski, L. D. Raedt, and S. Wrobel, Eds. University of Alberta, Edmonton, Canada, July 2002, pp. 21–35.

[19] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *In: Lecture Notes in Computer Science*. Springer, 2001, pp. 42–53.

[20] A. Clare and R. D. King, "Predicting gene function in saccharomyces cerevisiae," in *ECCB*, 2003, pp. 42–49.

[21] N. Holden, "A hybrid particle swarm/ant colony algorithm for the classification of hierarchical biological data," in *In Proceedings of the IEEE Swarm Intelligence Symposium (SIS-2005)*. IEEE Press, 2005, pp. 100–107.

[22] N. Holden and A. A. Freitas, "Hierarchical classification of g-protein-coupled receptors with a pso/aco algorithm," in *in Proceedings of the IEEE Swarm Intelligence Symposium (SIS ?06*, 2006, pp. 77–84.

[23] L. J. Jensen, R. Gupta, N. Blom, D. Devos, J. Tamames, C. Kesmir, H. Nielsen, H. H. Stærfeldt, K. Rapacki, C. Workman, C. A. F. Andersen, S. Knudsen, A. Krogh, A. Valencia, S. Brunak, and B. Cnbcsic, "Prediction of human protein function from post-translational modifications and localization features," *J Mol Biol*, vol. 319, pp. 1257–1265, 2002.

[24] A. Lagreid, T. R. Hvidsten, H. Midelfart, J. Komorowski, and A. K. Sandvik, "Predicting gene ontology biological process from temporal gene expression patterns." *Genome Res*, vol. 13, no. 5, pp. 965–79, 2003.

[25] K. Tu, H. Yu, Z. Guo, and X. Li, "Learnability-based further prediction of gene functions in Gene Ontology," *Genomics*, vol. 84, pp. 922–928, 2004.

[26] Z. Barutçuoglu, R. E. Schapire, and O. G. Troyanskaya, "Hierarchical multi-label prediction of gene function," *Bioinformatics*, vol. 22, no. 7, pp. 830–836, 2006.

[27] Y. Guan, C. Myers, D. Hess, Z. Barutcuoglu, A. Caudy, and O. Troyanskaya, "Predicting gene function in a hierarchical context with an ensemble of classifiers," *Genome Biology*, vol. 9, no. Suppl 1, p. S3, 2008.

[28] R. T. Alves, M. R. Delgado, and A. A. Freitas, "Multi-label hierarchical classification of protein functions with artificial immune systems," in *Proceedings of the 3rd Brazilian symposium on Bioinformatics: Advances in Bioinformatics and Computational Biology*, ser. BSB '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 1–12.

[29] C. N. Silla. and A. A. Freitas, "A global-model naive bayes approach to the hierarchical prediction of protein functions," in *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, ser. ICDM '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 992–997.

[30] F. E. B. Otero, A. A. Freitas, and C. G. Johnson, "A hierarchical multi-label classification ant colony algorithm for protein function prediction." *Memetic Computing*, pp. 165–181, 2010.

[31] W. Bi and J. Kwok, "Multi-label classification on tree- and dag-structured hierarchies," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ser. ICML '11. New York, NY, USA: ACM, June 2011, pp. 17–24.

[32] M. Pugelj and S. Džeroski, "Predicting structured outputs k-nearest neighbours method," in *Proceedings of the 14th international conference on Discovery science*, ser. DS'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 262–276.

[33] H. Borges and J. Nievola, "Hierarchical classification using a competitive neural network for protein function prediction," in *Proceedings of the 2012 Internacional Conference on Artificial Inteligence (ICAI'12)*, vol. 2, july 2012, pp. 237–243.

[34] L. M. Romão and J. C. Nievola, "Hierarchical classification of gene ontology with learning classifier systems," in *Advances in Artificial Intelligence - IBERAMIA 2012*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, vol. 7637, pp. 120–129.

[35] G. D. Oosthuizen, "The use of a lattice in knowledge processing," Ph.D. dissertation, Glasgow, Scotland, UK, UK, 1988, uMI Order No. GAXD-96805.

[36] M. Sahami, "Learning classification rules using lattices (extended abstract)," in *ECML*, 1995, pp. 343–346.

[37] M. Liquière and E. M. Nguifo, "Legal: un système d'apprentissage de concepts à partir d'exemples." in *Journées Françaises sur l'Apprentissage*, 1990.

[38] E. M. Nguifo, "Galois lattice: A framework for concept learning-design, evaluation and refinement," in *ICTAI*, 1994, pp. 461–467.

[39] C. Carpineto and G. Romano, "Galois: An order-theoretic approach to conceptual clustering," in *ICML*, 1993, pp. 33–40.

[40] C. Carpineto and G. Romano, "A lattice conceptual clustering system and its application to browsing retrieval," *Machine Learning*, vol. 24, no. 2, pp. 95–122, 1996.

[41] J. Silva, L. Zarate, and N. Vieira, "Formal concept analysis for data mining: Theoretical and practical approaches," *International Conference on Engineering of Intelligent Systems*, 2006.

[42] J. Silva, "Algoritmos de classificação baseados em análise formal de conceitos," Master's thesis, Universidade Federal de Minas Gerais, 2007.

[43] P. Cellier, S. Ferré, O. Ridoux, and M. Ducassé, "A parameterized algorithm for exploring concept lattices," in *Proceedings of the 5th international conference on Formal concept analysis*, ser. ICFCA'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 114–129.

[44] J. Poelmans, P. Elzinga, S. Viaene, and G. Dedene, "Formal concept analysis in knowledge discovery: a survey," in *Proceedings of the 18th international conference on Conceptual structures: from information to intelligence*, ser. ICCS'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 139–153.

[45] S. Kiritchenko, S. Matwin, and A. F. Famili, "Functional annotation of genes using hierarchical text categorization," in *in Proc. of the BioLINK SIG: Linking Literature, Information and Knowledge for Biology (held at ISMB-05*, 2005.

[46] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 233–240.

[47] F. Horn, E. Bettler, L. Oliveira, F. Campagne, F. E. Cohen, and G. Vriend, "Gpcrdb information system for g protein-coupled receptors," *Nucleic Acids Research*, vol. 31, no. 1, pp. 294–297, 2003.

[48] K. F. Tipton and S. Boyce, "History of the enzyme nomenclature system," *Bioinformatics*, vol. 16, no. 1, pp. 34–40, 2000.

[49] D. Filmore, "It's a GPCR world." *Modern Drug Discovery*, vol. 7, pp. 24–28, 2004.

MedTrad⁺: An Expert System for Traditional Medicine

Konan M. BROU[1], Ibrahim LOKPO[1] and Bi Tra GOORE[1]

[1]Department of Mathematics and Informatics, INP-HB, P. O. Box 1093 Yamoussoukro, Côte d'Ivoire

Abstract - *In this paper, we propose an expert system generator which is based on the use of icons rather than textual content to represent facts. It has been designed, implemented and used to create an expert system in the African traditional medicine where actors are often illiterate. Thanks to its iconic interface which is suitable for traditional medicine practice knowledge, this system will enable users to be totally independent, they don't necessarily need intermediaries in the management and usage process of their own knowledge base.*

Keywords: Knowledge base, Knowledge-Based system, traditional medicine, medicinal plants, iconic interface.

1   Introduction

Traditional medicine holds a prominent position in developing countries' health policy, and mainly those countries in Africa. According to WHO's 2007 report [1], up 80% of the African populations use traditional medicine to satisfy their health needs. However, this ancestral knowledge that its practitioners jealously transmit orally to other generations is passing away with the disappearance of these practitioners. Knowledge-Based Systems (KBS) have then been conceived in order to preserve them. [8] [7] [4]. The cognisor is the exclusive manipulator of the system in these KBS. The KBS interface is badly hooked to traditional medicine, whose users are not computer science specialists. In fact, most traditional medicine practitioners are reluctant to use the informatics tool at best because they are illiterate and worse because they refuse to share their knowledge and know-how with other persons. To remedy this problem and make the practitioners trust the informatics tool, we have proposed an expert system that we have called MedTrad⁺. The new system is based on the use of icons which are more suitable to the traditional medicine practitioner's culture. The system will allow practitioners to get rid of intermediaries, in order not only to feed their own KBS, but to update it, and submit it requests.

We will present existing works on it in chapter2, and the description of MedTrad⁺ in chapter3. Chapter4 displays the usage procedure of MedTrad⁺, and the conclusion makes the point of the work which has been conducted.

2   Material and methods

2.1   Existing works

The traditional medicine issue has been tackled in the process of conducting the following works:

The SEIGOGA expert system [8] allows the identification of plants in an iconic way and the tracking of their curative virtues. Another system allows the capture of a patient's symptoms interactively, and the arrival to diagnosis, followed by some recommendations in terms of curing with plants. Knowledge on diseases is represented with rules. This system is based on a geographic exploitation, which requires some competence from the user (i.e. dividing the screen into several zones and choosing functions and comparison operators...).

MEDTRA system [7] is a traditional medicine Knowledge Base (KB) creation tool, which will move towards a multi-expert system that deals with plants. Although it generally concerns African natural medication substances it is particular to Cameroonian ones. This system is not meant for herbalists mainly, but rather for experts who are involved in its conception (i.e. computer scientists, botanists, biochemists, chemists, pharmacologists, and agronomists).

The MedTrad [2] [3] system is based on the exploitation of a concept base on medicinal plants. Its objectives are on the one hand to capitalize the African ancestral knowledge on physiotherapy which present rational aspects and convincing results as felt by numerous African researchers [1], [10] and [12]. On the other hand it brings support to traditional healers in their diagnosis process and their attempt to propose rigorous therapy. This system is a Web application which includes a database but does not allow any inferences.

2.2   GExpert⁺ architecture

GExpert is an expert system generator ranks 0+ that the IUT of Bayonne (France) developed then the students of INP-HB of Yamoussoukro (Côte d'Ivoire) improved [9] by adding to it an iconic graphic interface (GExpert+). The KB contains all the information which is specific to the field in terms of facts and rules.

MI is a program which knows how to use the KB to solve a problem. It models the expert's reasoning within the system. It uses two modes of reasoning: front chaining and back chaining. The iconic interfaces facilitate communication between the system and the user or the expert.
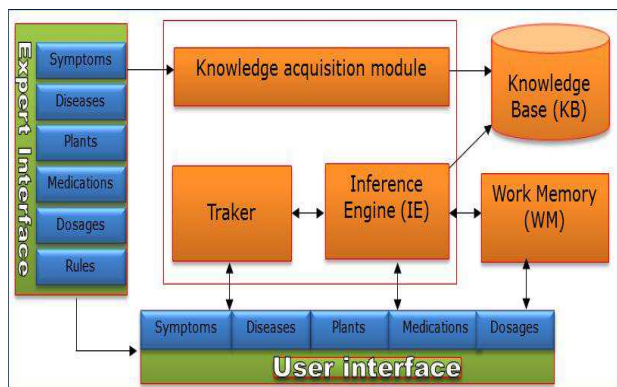
Table 1: GExpert$^+$ architecture.

Knowledge acquisition module is a tool that helps the expert to structure the domain knowledge, to identify and formalize the domain concepts. Work Memory or Fact Base contains the proper facts which relate to a problem that requires a solution. The Tracker enables the explanation of the WM reasoning system by providing a hint which is made of applied rules in order to check a given hypothesis (Table1).

2.2.1 Icons in GExpert$^+$

The particularity of GExpert$^+$ is its use of icons to represent facts instead of their textual content. This benefits the creation of expert systems which are accessible to illiterate informatics users. Their use permits to set a universal communication system between system users, depending neither on the language that we speak (French, English, etc.) nor on our origins (ethnic language), nor on our tradition [5] [6]. They can be created in conjunction with other icons as displayed in table2.
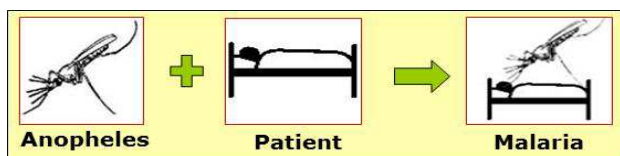
Table 2: Disease icons.

This malaria representation displays an image of insect over a patient. This means that the insect dominates the patient or it causes the disease.

2.2.2 Rules representation

KB includes all the information which relate to traditional medicine. It involves four kinds of rule:

Diagnosis rules: Their hypothesis contain the symptoms and their conclusion the diseases. Example: IF fever AND headaches AND bile vomits THEN malaria.

Table 3: Malaria diagnosis.

Therapeutic rules: Their hypotheses contain the diseases and their conclusion the plants. Example:

IF malaria THEN Azadirachta indica or IF malaria THEN papaya root.

Azadirachta indica is the scientific name of neem

Table 4: Healing prescription.

Medication preparation rules: The rules show how medication recipes are prepared. Medication1: Grind a male papaya root and mix it with lukewarm water.

Table 5: Preparation of medication.

Taking rules of medication1: They show how to take the medication and the way to take it. Example: Medication1 is taken twice a day in compliance with a purgative method
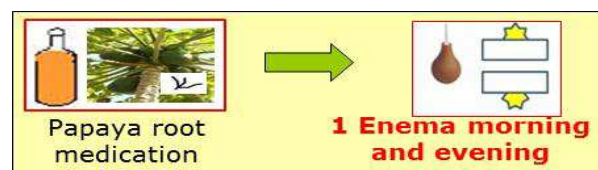
Table 6: Taking procedure of medication1.

2.2.3 Iconic language grammar

A formal grammar has been designed with regard to herbalists' language in GExpert+. This ensures that adding a rule to KB remains a coherent process; this also means that the added rule is consonant with the syntactic rule (diagnosis, therapeutic rules, etc...). For instance, the conclusion to a diagnosis rule must simply relate to a disease. Here are below

the main rules of this grammar. Their edition has been based upon this grammar.

```
<diagnosis> →    if <symptom> then <disease>
<symptom>→    {<sign>}
<sign> →    fever |general pain |... | headaches
<disease> →    fever |malaria |...| coughs
<curing> → <medication> <disease> |if <disease> then
<medication>
<medication> →    <dose> <taking procedure>
<dose> →    <part of plant>< preparation>
<taking procedure> → two purgatives a day|...| three
glasses a day
<part of plant> →    <stem>|flower>|  <root>|<bark>
|<leave>
<preparation> → grind |boil|
<stem> → Acacia stem|...|neem stem
<flower> → Acacia flower|...| neem flower
<root> → Acacia root|...|neem root
<bark> →Acacia bark|...|neem bark
<leave> → Acacia leave|....|neem leave
```

---

3   Results

3.1   MedTrad+ architecture

A web interface helps the traditional medicine practitioners to manage and use his KBS (Table 8). The inference engine has been developed as a CGI (Common Gateway Interface) program in third part architecture as shown in table 7.
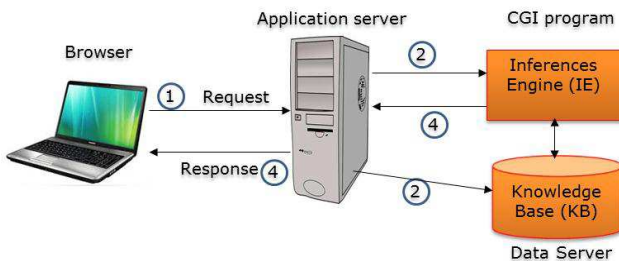
Table 7: MedTrad+ Architecture

1. The client submits a query request containing the data of the problem and the hypothesis to be tested in the Application server.
2. The Application server sends the facts and the rules to the KB or sends the data of the problem and the hypothesis to be tested to the inference engine (CGI program) that runs it and returns the response.
3. The CGI program returns the result of the execution of the request to the Web server.
4. The Web server returns the response to the client.

3.2   Use of MedTrad+

The first step consists in creating a KB. Table 9 shows the fact base and Table 10 display the creation of a relating rule by catching the bases upon which they rest, with their subsequent conclusion.
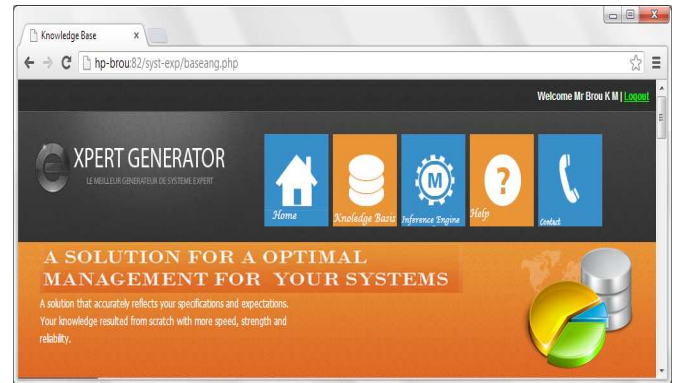
Table 8: MedTrad+ Web interface.

Table 9: MedTrad+.fact base
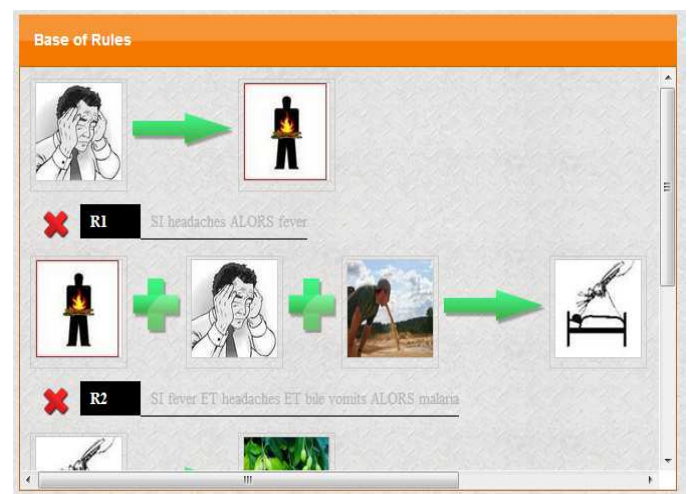
Table 10: Devising rules with MedTrad+.

The second step of the process is concerned with the effective use of the system. Table12 displays the deduction of Malaria together with the plant that cures it after grasping its symptoms which are included the rule below (table 11).

Table 11: Facts in working memory.
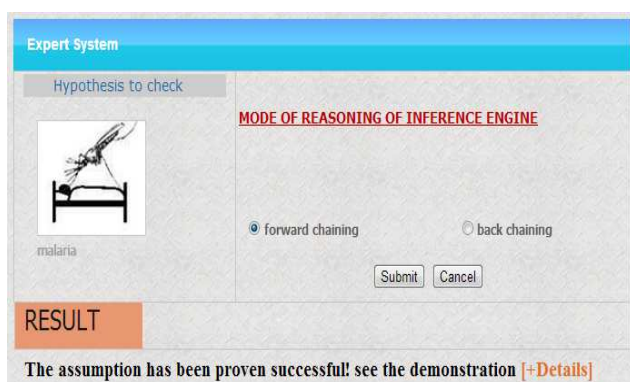
Table 12: Detecting disease with MedTrad.

4 Discussion and conclusion

4.1 Discussion

The analysis of the results shows that all traditional medicine practitioners can use the system MedTrad + to build its knowledge base with very little help. This assistance is limited to scan the images and icons to install on his machine. traditional medicine practitioners may grant rights to query other users. MedTrad+ certainly can not replace a traditional medicine practitioners, but it can be used to: provide medical assistance to patients, learning herbalism, learning traditional medicine.

Traditional medicine practitioners that wins with this system? MedTrad+ can allow it to sustain its knowledge, bring awareness for this on the web, increase productivity, increase its customers easier and cooperation with other healers.

4.2 Conclusion

Our goal was to conceive an expert system for traditional medicine. First of all, the work required that a GExpert+ be conceived. This conceived tool is a generator of an expert system, the main purpose of which is to use icons to represent facts. GExpert+ has then allowed the realization

of the MedTrad$^+$ expert system that traditional medicine practitioners are able to use more easily. Traditional healers from Yamoussoukro region are working toward its validation, and it includes 134 facts and 66 rules. This is also followed by an attempt to conceive a MedTrad$^+$ iconic language interpreter, which will permit a non-recurring KB to remain in a coherent state of use.

Referencies

[1] L. A. Assy : "Plantes utilisées dans la médecine traditionnelle en Afrique de l'Ouest", 1996, Édition Roche Basse, Suisse.

[2] K. M. BROU : "Base de concepts : Contribution à la représentation et à l'exploitation hypertextuelle de concepts-Le système CoDb-Web", Thèse de doctorat de l'Université de Pau et des Pays de l'Adour, novembre 1997 ;

[3] K. M. BROU : "CoDB-Web : UN système de représentation et de gestion hypertexte de concepts", Actes du 4eme Colloque Africain sur la Recherche en Informatique - CARI'98 Presses Universitaires de Dakar, UCAD, OCTOBRE 1998 pp 613 à 624.

[4] K. M. BROU : "Le modèle de représentation et de gestion hypertexte des concepts d'un domaine dans le système CoDB-Web", JUIN 1999, RINT n° 19, Revue semestrielle coéditée par l'Agence de la francophonie et la Communauté française de Belgique PAGE 89 à 100.

[5] M. CARTIER : "La grammaire iconique", http://www.michelcartier.com/

[6] P. E. DOBE : "Conception et réalisation d'un langage iconique pour la médecine traditionnelle", Mémoire de DEA, Université d'Abobo-Adjamé, 2009.

[7] L. P. FOTSO : "MEDITRA ou Système auteur de Création d'une base de Connaissances en plantes médicinales et médecine traditionnelle", 1996, CARI'96 : actes du 3ème colloque africain sur la recherche en informatique, pp 755 à 764.

[8] C. FRASSON, F. HOUTSA, P. OBENSON : "Interface visuelle pour l'aide au diagnostic médical en médecine traditionnelle", ICO, Vol. 4, n° 1,2, 1992, pp17-25.

[9] N. L. KOFFI, E. B. ASSOUMOU : "Conception et réalisation d'une interface iconique pour Medtrad", Rapport de projet 2e année ingénieur INP-HB, 2009.

[10] G. MAYNART, M. LO, D. FORTIN, "Plantes médicinales du sahel", Série Etudes et Recherches, 1990.

[11] Rapport   OMS   2006-2007,
http://www.who.int/medicines/publications/BiennialReport0
607FR.pdf

[12] A. SIJELMASSI : "Les plantes médicinales du Maroc",
Edition Le Fennec, 3e édition 1993.

[13] Nathalie Cindy Kuicheu, Laure Pauline Fotso, François
Siewe, "Iconic Communication System by XML Language
(SCILX)", 2007

Improve Path Planning Performance by Monitoring Human Decisions

CheeChian Cheng, Henry Hexmoor
Computer Science Department, Southern Illinois University Carbondale, Carbondale IL 62901, USA

Abstract - *Behavioral based control is usually based on a set of previously codified rules to which the system must adhere. Adaptive behavioral modeling involves learning new domain knowledge and applying the newly learned knowledge to the situation while still complying with the rules in the process of performing the task. This paper proposes a solution to improve path planning by observing human decisions in path planning and chunking pieces that are superior to stored paths.*

Keywords: Artificial intelligence, machine learning, path planning, chunking, adaptive behavioral modeling

1   Introduction

Traditionally, machine learning was viewed as a branch of artificial intelligence, but starting in 2000, many researchers began to treat machine learning as a separate field by itself. According to Patrick Langley, this is a great loss to both fields [15]. Chunking is a psychological memory mechanism to perform memory tasks by reproducing smaller pieces of information [19]. Miller argued that most people can remember about seven (7) chunks in their short term memory (STM). By using chunking, an agent will be able to learn and apply his past experience (memory based learning) to solve a given problem.

In the gaming industry, most games, especially Role Playing Games (RPG), require a player to perform a series of steps to complete the level. The player (or agent) will need to explore the game and learn from exploration before exploiting the game. Hence, there are expert players who can play the game well based on his/her knowledge (or past experience) in the game while novice players spend most of their time exploring the game, learning and acquiring knowledge for most of the play time. One of the goals of modeling agent's decision is to create an intelligent agent (i.e., computer player) that plays like a human – expert player.

2   Related Work

2.1   Chunking

Chunking often refers to a process of forming a group of information that will act like a long-term memory to the agent [10].   The agent will perform memory tasks by recalling chunks of information and reconstructing them into vital information that helps them to perform various tasks. Chunking is often considered as a mature technology and has been widely used in various areas (especially in machine learning) including phrase (i.e. part of speech) recognition, general problem solving approaches as used in SOAR, chunking with support vector machine and other areas. Grover and Tobin have proposed a rule-based chunking approach to implement chunking and at the same time, provided the property of reusability [10]. They argued that by using XML as the basis for chunking rules, it would provide some degree of flexibility with respect to reusability that made it very appealing in machine learning.

2.2   Cognitive Maps

Cognitive Map is credited to Edward Tolman when he started his research on rats' behavior in exploring mazes [35]. Tolman argued that the rats must have known something about the maze's topography and the rats must have formed some kind of cognitive map during the training. Later, Tolman et al. argued that rats often take the shortest path to the goal box whenever possible and coined this behavior as a type of spatial insight [36]. The rats were able to construct and accumulate spatial knowledge about the maze, which allowed them to visualize the maze during the testing condition and figure out the right path to reach the goal box. By converting each junction of these paths into a state (of state diagram) and applying proper reinforcement (reward or punishment), a route can be derived by applying various algorithms, including reinforcement learning and Markov Decision Process, to simulate the rats' behavior.

2.3   Modeling of Ecology Movement

Researchers in ecology proposed a framework for fitting multiple random walks to animal movement paths by modeling the animals' movement [21]. By using the animal's step lengths and turning angles as features, the proposed framework will be able to classify the behavioral states of that animal by any machine learning approach, e.g. Support Machine Vector, K-Nearest Neighbor, Artificial Neural Network, etc.

2.4   Belief, Desire, and Intention (BDI) Model

Belief, Desire (Goal) and Intention model the behavior of a rational software agent by imbuing human practical

reasoning [8]. Belief (of an agent) refers to the state of the environment or the facts which may include forward inference rules that will lead to new beliefs (which can be implemented by any rule-based engine like Jess, CLIPS and Drools). Instead of using the term knowledge like most of the expert systems do, the agenda or output of the rule-based engine (the newly inferred belief) might not be true, in contrast to expert systems where the agenda of the rule-based engine is knowledge. Desire represents the goal which the agent wishes to accomplish. An agent can have a set of desires based on its current beliefs. Intention represents the action to engage the desire, in other words, what the agent has decided to do by choosing from the desire sets. In this framework, one of the limitations is the ability to learn from the agent's past decisions and adapt to a new environment.

2.5   Subsumption Architecture

Subsumption is an approach to prioritize behaviors into a hierarchy and output the behavioral action based on the agent's input sensors [1]. Each layer in the hierarchy describes a behavior or an action that the agent needs to take. These behaviors constantly compete among themselves based on its current inputs and its goals (intention), thus producing an output as the behavioral-based action. The main disadvantage of using this architecture is that the behavior-based action in each layer of the hierarchy tends to conflict with each other when there are too many layers. Another more democratic approach of behavioral competition is to use the voting system where each behavior is given a voting count based on its degree of relevancy towards its current inputs and goal [31].

2.6   Artificial Neural Network

Artificial Neural Network (ANN) consists of neurons, synaptic weight among neurons, and threshold/activation function; it is mimicking the neurophysiologic brain model with the ability to learn and work with inaccurate information as input and subsequently producing a set of outputs [27]. There are various types of ANNs; they are single perceptron, feed forward back propagation, Hopfield network, Bidirectional Associative Memory (BAM), Hebbian learning, Self-Organizing Map, hybrids of fuzzy and ANN, evolutionary ANN (Genetic Algorithm + ANN), and etc.

In single perceptron and feed forward back propagation ANN, ANN learns by repeatedly adjusting its synaptic weight between neurons until it reaches satisfactory results [38]. This process is called training in the field of ANN.

2.7   Genetic Algorithm

Genetic Algorithm (GA) is a stochastic search algorithm based on Neo-Darwinian natural evolution theory [23]. The key concepts in GA are reproduction, mutation, competition, and selection [20]. In Neo-Darwinian natural selection theory, the fitter organism will have a higher chance of being selected

for reproduction. This, in turn, will produce a better next generation (fitter next generation) and if done repeatedly for a number of generations, it should reach maxima in term of fitness. However, this might be a local maxima, thus mutation is introduced so that the search can reach global maxima. Learning in GA can be perceived as producing offspring that has better fitness over generations.

2.8   Reinforcement Learning

Reinforcement Learning is an approach of machine learning that lies in between supervised and unsupervised learning. It was pioneered by Richard S. Sutton and Andrew G. Barto. This approach will observe two results from every lesson: win or loss; a "reward" will be awarded to the winning result and "reinforcement" to the losing result. This approach is sometimes named as "trial-and-error learning" besides being coined as "Reinforcement learning" [26]. In short, reinforcement learning is about how to traverse through various states and finally get rewarded, which is identical to, for example, a pet (animal) is given rewards (treat/food) when it had successfully performed a series of actions.

One of the learning approaches for reinforcement learning is Q-learning [39]. Q-learning learns the policy by assigning random numbers to every possible action in each state and readjusting these numbers based on its training or experience. This process continues until each action in the state points towards the goal state [26].

2.9   Problem Space

Problem space is the fundamental organizational unit of all human goal-oriented symbolic activity [24]. As defined by Newell, a problem space consists of symbolic structures and operators, where symbolic structures may represent the states of the space while operators take a state as input and produce a new state as output. A problem in problem spaces consists of a set of initial states, a set of goal states, and a set of path constraints. Problem space is mental constructs and may lead to external actions.

2.10   Production System

A production system refers to a system where it produces products based on a given set of input (facts) as shown in Figure 1. Facts are stored in working memory (WM) and serve as the input to the production system and the products of the production system are the output.
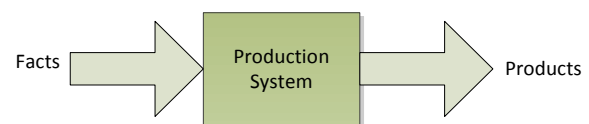

Figure 1. A Production System

A typical rule-based production system is made up of various rules, usually drawn out by domain expert, which will translate known facts to useful outputs. A rule is made up with antecedent (condition) and consequent (action or output) [23] and sometimes, the antecedent part of the rule is referred as Left-Hand-Side (LHS) while the consequent is referred as Right Hand Side (RHS) of the rules. These notations are interchangeable but LHS and RHS are commonly used throughout the A.I. community. One important difference between production system and computer program is production system matches the rules in parallel and continuous processing (analogy to a circuit board logic gates) while computer program does it sequentially.

2.11 RETE Algorithm

Charles Forgy proposed a fast algorithm for the many pattern/many object pattern match problem by building a network of nodes that processes the delta-change of working memory and produces conflict set [7]. Working memory is the content of data operated on by the productions, which is the content of variable for the conditional test or Left Hand Side (LHS). Conflict set is the output of the matching process of the productions. The conflict set will then be past to conflict resolution to select one production (rule) with a satisfied test condition. RETE algorithm will translate the production rules into a network (tree like) with nodes that hold memory [7].

2.12 Shortest Path Algorithm

In 1956, a Dutch computer scientist came up with an algorithm to find shortest path in a graph (Dijkstra's algorithm) and later in 1968, Hart, Nilsson, and Raphael came up with A-star (A*), a variant of Dijkstra's algorithm, which is more efficient in finding minimum-cost paths in a graph. Unlike Dijkstra's algorithm, A* uses heuristic and best-first search method to find the minimum-cost path by skipping most of the evaluation process in other nodes that will not produce a minimum-cost sub-graph [11]. A* is widely used in game industry due to its simplicity in implementation, low time and space complexity in finding shortest path from one point to another. However, A* will not work with negative cost edge and requires a heuristic function to work accurately; otherwise, it will not produce the shortest path (if heuristic function overestimated the distant to the destination).

3 Proposed Approach

The basic idea is to use a rule-based engine, preferably RETE based engine to produce output based on current states of the agent, current states of the environment and the chunking between two points. Each chunk is given a score based on a scoring function. A typical scoring function might be using the distance and elapse time between two points.

The system will learn from a human agent by comparing the chunk's score in its memory (rules). If human decision has a better score than its corresponding rules (based on current states of the agent and the environment), a new rule will be constructed and the old rule in the system will be replaced as shown in Figure 2.
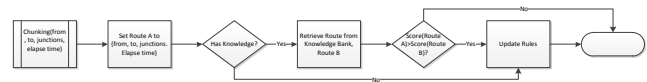
Figure 2. System Flow Chart

4 Implementation

A testbed was built for agent to run errand on every round of the game. An agent will be given k errands. The agent is free to choose any location to start with and the agent has to return to his/her base upon finishing the errand. Performance is measured by the distant travelled and time taken to complete the errand. This is identical to modeling human player or bot (agent) playing video or computer game and solving the riddle based on the human player's past experience. In each round of the game, the time and distance travelled will be recorded. Upon starting of the new round/game, the system will suggest to the player the route he/she should take based on his/her past experience. Of course, the player has the choice whether to follow the proposed route or exploring new route. After each exploration, a new chunk might be created if it has a better score (time and distance) and will be stored as a new knowledge. Average speed between two points is used as the scoring function in the implementation.

The testbed consists of Manhattan grid with several points of interest that the agent must visit. In each tour, the agent will travel through all the points of interest based on the agent's current state (urgency and service level) and the environment factors such as weather and time. During implementation, a RETE Rule-Based Engine was used for pattern matching. The agent's service level, urgency, weather and the time of the day were used as states and served as part of the chunking. However, in all machine learning approaches, they are constrained by boundaries of the modeled environment (in this case, they are service level, urgency, weather and the time of the day). Often, humans learn parameters of the environment that would not perceive pertinent prior to that point. No known machine learning technique is capable of ascertaining new environmental parameters.
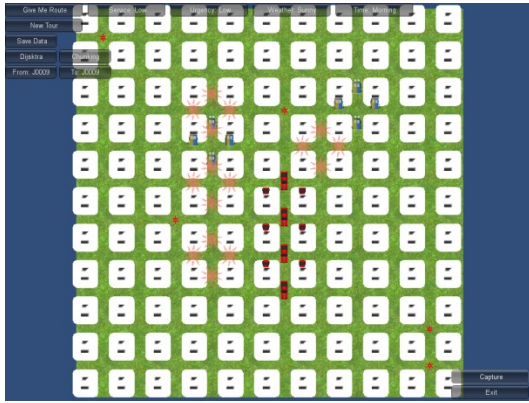
Figure 3. Implementation

At any given time, an agent will be given a series of errand, which forms a tour. In this case, an agent refers to the human. The agent always begins its tour at a point and ends the tour at the same point. Agent has the flexibility to choose which point to run his/her errand first. Upon choosing the destination point, the agent will recall from its long term memory for the route it has taken before under the same conditions (states), i.e., service, urgency, weather and time. If the agent has never run that route before, Dijkstra's algorithm will be used for finding the shortest path to the destination. The route will serve as a memory chunk and will be stored into the memory by adding a production rule into the agent's knowledge based system. A chunk can be defined as follows.

$$Chunk = \{(from, to, service, urgency, weather, time), junction\ names, elapse\ time\}$$

In this case, the production rule will serve as a long term memory for the agent's knowledge. Rete engine is used for production by feeding in the Left Hand Side (LHS), i.e., from, to, service, urgency, weather and time states into the production system before producing the route as an output.

5   Conclusions

In conclusion, by improving agent's path planning with the approach mentioned in above, we have developed an approach to enhance agent route planning. Chunks of knowledge that represent the agent's past experience and knowledge are being reused for successive route planning. A suitable scoring function (distance between two points was used in the implementation) was used for determining whether a new chunk of information should supersede the previous chunk of knowledge. Our approach has considered nuances in routes that are meaningful for human route planners, and this has added a degree of cognitive reasoning and realism to route planning as well.

More effects of various categories of nuances on route planning should be considered, for example danger, outlook/exploration, pit and refuel stops, marker dropping and removal, and repair. The model will react and behave better with human-like cognitive reasoning when more categories of

nuances are identified. However, the proposed approach does not analyze the cognitive reasoning of why the human agent makes that decision; it simply duplicates the decision made by human to mimic and imbue the cognitive reasoning. This is not suitable for systems such as diagnosis expert system in the medical field.

6   References

[1]   Arkin, R. C. "Behavior-Based Robotics". The MIT Press, 1998.

[2]   Bellman, R. E. "A Markov decision process"; Journal of Mathematical, 679-684, 1957.

[3]   Braitenberg, V. "Vehicles: Experiments in Synthetic Psychology". The MIT Press, 1986.

[4]   Derbinsky, N., Laird, J. E. "Extending Soar with Dissociated Symbolic Memories"; Proceedings of the Symposium on Human Memory for Artificial Agents 36th, AISB, 2010.

[5]   Douglass, S. A., Ball, J., Rodgers, S. M. "Large declarative memories in ACT-R"; 9th International Conference of Cognitive Modeling, , Manchester, United Kingdom, 234, 2009.

[6]   Finnsson, H., Bjornsson, Y. "Learning Simulation Control in General Game-Playing Agents"; National Conference on Artificial Intelligence, Atlanta, Georgia, 2010.

[7]   Forgy, C. L. "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem"; Artificial Intelligence, 19(1), 17-37, 1982.

[8]   Georgeff, M., Pell, B., Pollack, M., Tambe, M., Wooldridge, M. "The Belief-Desire-Intention Model of Agency". Springer, 1999.

[9]   Gross, J. L., Yellen, J. "Graph Theory and Its Application". Chapman & Hall/CRC, 2006.

[10] Grover, C., Tobin, R. "Rule-Based chunking and reusability"; Proceedings of LREC 2006, 873-878, 2006.

[11] Hart, P. E., Nilsson, N. J., Raphael, B. "A Formal Basis for the Heuristic Determination of Minimum Cost Paths"; Artificial Intelligence Group of the Applied Physics Laboratory, 4(2), 100-107, 1968.

[12] Hu, Q., Yue, W. "Markov Decision Processes With Their Applications". Springer, 2008.

[13] Kleinberg, J., Tardos, E. "Algorithm Design". Pearson Education Inc, 2005.

[14] Koning, K. D., Bredeweg, B., Breuker, J., Wielinga, B. "Model-based reasoning about learner behavior"; Artificial Intelligence, 117, 173–229, 2000.

[15] Langley, P. "The changing science of machine learning"; Machine Learning, 82(3), 275-279, 2011.

[16] Larid, J. E., Rosenbloom, P. S., Newell, A. "Chunking in Soar: The anatomy of a general learning mechanism"; Machine Learning, 1, 11-46, 1986.

[17] Loh, C. "Researching and Developing Serious Games as Interactive Learning Instructions"; International Journal of Gaming and Computer Mediated Simulations, 1(4), 1-19, 2009.

[18] McDermott, J., Newell, A., Moore, J. "The efficiency of certain production system implementations"; SIGART Bull, 63, 1977.

[19] Miller, G. A. "The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information"; The Psychological Review, 63, 81-97, 1956.

[20] Mitchell, M. "An Introduction to Genetic Algorithms". The MIT Press, 1998.

[21] Morales, J. M., Haydon, D. T., Frai, J. "Extracting more out of relocation data: Building movement models as mixtures of random walks"; Ecology, 85(9), 2436-2445, 2004.

[22] Nayak, P., Gupta, A., Rosenbloom, P. "Comparison of the RETE and TREAT production matchers for Soar (a summary)"; The Soar papers (The MIT Press), 1993.

[23] Negnevitsky, M. "Artificial Intelligence: A Guide to Intelligent Systems". Addison-Wesley, 2005.

[24] Newell, A. "Reasoning, problem solving and decision processes: The problem space as a fundamental category"; Attention and performance VIII, 1979.

[25] Nilsson, N. J. "Aritificial Intelligent: A New Synthesis". Morgan Kaufmann Publishers Inc., 1998.

[26] Nilsson, N. J. "The Quest For Artificial Intelligence: A history of ideas and achievements". Cambridge University Press, 2010.

[27] Rabuñal, J. R., Dorado, J. "Artificial Neural Networks In Real-Life Applications". Idea Group Inc., 2006.

[28] Reid, A., Staddon, J. "A Dynamic Route Finder for the Cognitive Map"; Psychological Review, 105(3), 585-601, 1998.

[29] Resnick, M. "Turtles, Termites, and Traffic Jams: Explorations in massively parallel microworlds". The MIT Press, 1997.

[30] Rosenblatt, J. K. "DAMN: A Distributed Architecture for"; Experimental & Theoretical Artificial Intelligence 9(2-3), 339-360, 1997.

[31] Rosenblatt, J. K., Payton, D. W. "A Fine-Grained Alternative to the Subsumption"; IEEE/INNS International Joint 2, 317-324, 1989.

[32] Roth, B. H. "An architecture for adaptive intelligent systems"; Artificial Intelligence, 72(1-2), 329-365, 1995.

[33] Scholkpof, B., Mallot, H. A. "View-Based Cognitive Mapping and Path Planning"; Adaptive Behavior, 3, 331-348, 1995.

[34] Sutton, R. S., Barto, G. A. "Reinforcement Learning: An Introduction". The MIT Press, 1998.

[35] Tolman, E. C., Honzik, C. H. ""Insight" in rats"; University of California Publications in Psychology, 4, 215-232, 1930.

[36] Tolman, E. C., Ritchie, B. F., Kalish, D. "Studies in spatial learning: I. Orientation and the short-cut"; Journal of Experimental Psychology, 36, 13-24, 1946.

[37] Turing, A. M. "Computing Machinery and Intelligence"; Mind, 59(236), 433-460, 1950.

[38] Veelenturf, L. "Analysis and Application of Artificial Neural Networks". Prentice Hall International (UK) Limited, 1995.

[39] Watkins, C. J. "Learning from Delayed Rewards, Ph.D. thesis". Cambridge, England: Cambridge University, 1989

SESSION

PREDICTION AND FORECASTING METHODS + HUMAN ACTIVITY DISCOVERY

Chair(s)

TBA

Can We Predict Speciation and Species Extinction Using an Individual-Based Ecosystem Simulation?

M. Mashayekhi[1], M. Hosseini Sedehi[2], and R. Gras[3]
[1,2,3] School of Computer Science, University of Windsor, Windsor, ON, Canada

Abstract - *Speciation and extinction are two essential phenomena in evolutionary biology. Many factors are involved in the emergence and disappearance of species. Due to the complexity of the interactions between various factors and the long time required for the observation, studying these phenomena is difficult in nature. Individual-based modeling is one solution to simulate such phenomena for investigating the important factors involved in their emergence in a reasonably short time. In this paper, using EcoSim, an individual-based ecosystem simulation, we attempt to predict speciation and extinction based on numerous factors, applying several machine learning techniques. Experimental results showed that factors, such as demographics, genetics, and environment are important in the occurrence of these two events in EcoSim as it has been observed in nature. The discovered rules showed good accordance with biological evidence. Finally, we argue that study of species behavior through EcoSim can give new and realistic insight into these events.*

Keywords: Individual-based system, Speciation, Extinction, Classification, Rule extraction

1   Introduction

One application of individual-based systems is to simulate ecological systems. For instance, studying species as one of the important building block of the ecosystems and one of the important concepts in ecology, using individual-based models, is interesting for ecologist where the complexity of natural ecosystems is a strong bottleneck.

A species is a group of individuals that are capable to exchange genes within themselves, but are reproductively isolated from other such groups. Consequently, there is no gene flow between two species [1]. Speciation is the division of one single species into two or more genetically distinct ones. It extends through time and leads to a hierarchal tree of historical relationship between species. It consists of two steps [2]. First, a new population should be established. This new population can exist in the same habitat or can be completely separated from the main population, depending on the type of speciation mechanism. For example, in sympatric speciation, a new population emerges from a single local population while in allopatric speciation a physical barrier separates a sub-population from the initial population. Second, a reproductive isolation should occur, based on different factors like genetic divergence, different habitats, and physical barrier, that reduce or prevent gene flow between organisms of different species. Additionally, a species is vulnerable to extreme situations such as environmental variations, which increase the possibility of extinction. One species cannot survive when its individuals are, for instance, not able to reproduce or cannot tolerate environmental conditions.

These two phenomena, speciation and extinction, can be affected by several factors that need more investigation. Based on Darwin theory, natural selection is the main reason for speciation and emerging genetics studies strengthened this theory by explaining variation in a population via genetic operations [3]. Pre- and post-zygotic barriers, which lead to reproductive isolation, are also very important in speciation. Geographically isolated populations tend to form new species as well [4, 5]. Moreover, sexual selection plays an important role in speciation [6]. Likewise, there are many factors involved in extinction that can be classified into the three main areas of demographics, genetics and environmental factors [7, 8]. Demographic factors can affect the birth rate and the death rate of the population. If the death rate is greater than the birth ratio, extinction can occur before the population can revive to a safe state. Additionally, the effect of demographic stochasticity is greater in smaller than in larger populations [9]. There is also possibility of genes lost when huge reduction occurs in a population and the gene frequencies may be changed due to drift or inbreeding [10]. Diminishing genetic variation may increase extinction risk by limiting the adaptation ability to stressful environments. Lastly, environmental factors such as natural catastrophes, availability of food, competitors, predators, and diseases influence the population by changing the demographic parameters and increase the likelihood of the extinction.

Predicting these two phenomena and discovering important factors involved, would bring new insights in evolutionary and conservation biology. However, observing and studying species in nature to extract species information is a difficult and time consuming process. In addition, speciation and extinction processes need a long time to appear and most of the time is not possible to observe them in nature. Individual-based modeling is a possible theoretical approach to overcome these limitations. The interest of this approach is that it allows complex interactions between multiple agents to shape the whole behavior of the system making it a powerful

tool to study how individuals' actions have consequences on a global ecosystem.

Several ecosystem simulations exist such as Avida [11], PolyWorld [12], and Echo [13]. However, they are not suitable for this type of study because of the lack of complex interactions between individuals, or the shortage of large scale system support in terms of number of individuals and spatial dimension.

In this paper, using EcoSim [14], a virtual ecosystem, we attempt to predict speciation and extinction in near future. For this purpose, we calculate several demographical, genetic, environmental, and spatial features, which are likely effective on these two events. Afterward, based on prediction results, we investigate the important factors (called features from now on) involved in these events. Moreover, we extract some rules based on these features for both speciation and extinction that can show accordance of obtained results with biological evidence. Being able to show that the emergence of species and their extinction in EcoSim is similar to what happen in nature would allow ecologists to propose more specific studies that can be performed using EcoSim. Moreover, this work shows the interest of the application of artificial intelligence concepts such as multi-agent systems and machine learning techniques in ecology. These kinds of studies can consolidate the interdisciplinary research between computer science and ecology.

Subsequent to this introduction, we explain EcoSim in the second and the data set preparation phase in the third section. In section four, the classification algorithm and evaluation metrics are described. Section five discusses experimental results of the speciation and extinction prediction. Finally, the last section includes conclusion and future works.

2   EcoSim

Ecosim is an individual-based evolving predator-prey ecosystem simulation to simulate complex ecosystems consisting of huge number of intelligent agents (also called individuals) interacting and evolving in a dynamic environment [14]. This virtual ecosystem benefits from diverse biological concepts such as speciation, extinction, food chain, reproduction, predation, and survival, which can be used to study various ecological theories.

In this simulation, two organism types, prey and predator, are evolved in an interactive habitat. This habitat is a torus world consisting of a matrix of 1000×1000 cells. Every cell contains an amount of grass, which functions as food for prey, as well as some meat, which remains from dead prey. Predator can hunt prey or eat available meat in the cell. Each individual is able to perform some actions based on whether it is prey or predator. For example, prey can move in the world, eat grass, escape from predator, mate with other prey (if they are genetically similar enough) and produce an offspring. All individuals act based on their behavioral model implemented by a fuzzy cognitive map (FCM) [15]. A FCM is a weighted graph in which each node represents one sensitive, internal, or motor concept (Fig. 1). Edges of a FCM show inhibitory or

excitatory effects between two concepts. Each concept has an activation level, which depends on the current perception of the individual about its vicinity and its past internal states. Sensitive concepts such as distance to food or distance to a sexual partner imply the individual's perception of its environment. The internal concepts such as fear, hunger, or satisfaction represent the interior sensation of an individual and finally the motor concepts like evasion, eat, or reproduce express the desire of an individual to perform each action. The perceived environmental information is used to compute the activation level of the sensitive concepts using fuzzification, which in turn affect the activation level of the internal concepts.
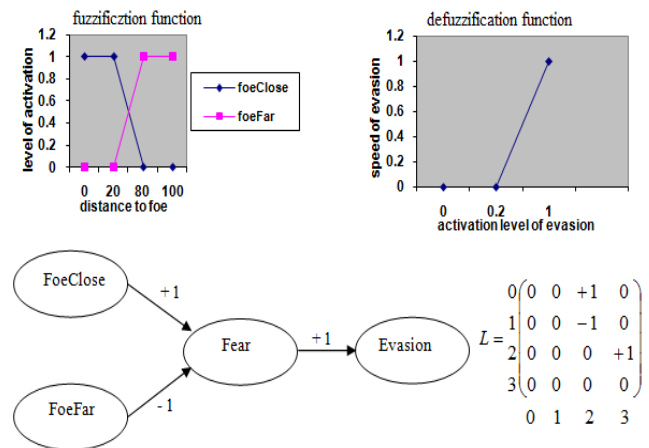
Fig. 1 [14] A simple FCM to perceive predator distance (using fuzzification) and make decision (using defuzzification) to evade along with its corresponding matrix, L (0 for 'Foe close', 1 for 'Foe far', 2 for 'Fear' and 3 for 'Evasion')

Then, the action of one agent is selected based on the activation levels computed by defuzzification of the motor concepts, which are affected by the sensitive and internal concepts (Fig. 1). The action with higher activation level is selected for the agent to perform. The FCM is coded into the genome of each individual and can evolve through reproduction. When a new offspring is created, its genome is formed by combining the parents' genomes and applying crossover and mutations operations. A time step of the simulation includes the computation of the behavioral model; the application of the action of all individuals; and, afterward, the updating of the world state, which consists of the emergence and the extinction of species and the growth and diffusion of grass. In this simulation, species concept is represented as a set of individuals having a level of genetic similarity, which can be computed by measuring the genetic distance (Euclidian distance between genomes) between individuals of one species. At the beginning of the simulation, the system starts with one species of prey and one species of predator. A speciation occurs after several time steps if two individuals in a species have a mutual genetic distance greater than a predefined threshold. Two new sister species are created by splitting the previous species using a 2-mean clustering algorithm presented in [16]. The two new sister

species are initially quite similar, allowing for some likely hybridization to occur. However, they quickly diverge genetically and become two completely independent species. Several studies have been done using Ecosim demonstrating its suitability to study fundamental concepts in biology. For example, Devaurs and Gras [17] have shown that the behavior of EcoSim is realistic by comparing the species abundance pattern in the simulation environment with real communities of species. Furthermore, the chaotic behavior of the system with multi-fractal properties has been proven by Golestani and Gras [18] as it also has been observed for real ecosystems.

3 Data Set Preparation

We used the result of 10000 time steps of three different runs of Ecosim. In each time step, there was a variable number of species with their corresponding features. It should be mentioned that each run is different from the others in terms of demography, environmental and genetic features due to stochastic processes in the model. The features were used for this study are as follows.

The spatial diversity measures the distribution of individuals in one species based on the physical locations of its individuals and is computed in two steps. First, the spatial centre of the species is located. Because EcoSim has a torus world, which means that the opposite borders of the grid are adjacent, circular statistics [19] was applied to compute the centre of the species' spatial distribution. Second, the total distance of all individuals to the center was calculated along with the spatial standard deviation. SpatialDiversityRatio is the ratio between spatial diversity to the number of individuals in the species.

Several defined features [20, 21] were also calculated to characterize the complex spatial dynamics of the world. Most of these metrics describe the composition and configuration of patches i.e. contiguous cells containing the same category value, in the landscape. In EcoSim, the individuals of each species are located in several cells in the world and we considered the adjacent cells containing individuals of the same species as one patch. Therefore, each species may inhabit in several patches called 'patch type' like shaded patch type in Fig. 2.

The features are number of patches (PatchNum), average patch size (PatchSizeAvg), patch area ratio (the ratio of the number of cells in one patch type to the area of the world), patch circumference (the number of cells in a patch with faces not shared by adjacent cells of the same patch type). For calculating adjacency, we used Moore neighborhood, which comprises the eight cells surrounding the central cell like in Fig. 2. Fractal dimension, computed using the box counting method [22], was used to quantitatively describe how one species occupies the world. Contagion was calculated to measure dispersion of a patch type. It is computed as the probability of finding a cell of patch type i next to a cell of patch type j. Lower value of contagion shows many small patches and higher value indicates few large ones. We modified the STC measure (SpatioTemporal complexity) to

be applicable for the two-dimension space and called it SC or Spatial Complexity. It describes how one patch type occupies the space and is calculated by counting number of cells occupied by patch type i in a window of dimension $n \times n$, where n is much smaller than the world size (n=5 in our case). SC value is lower for uniform or ordered patch shapes and is higher for complex shapes. Both contagion and SC are between 0 and 1.
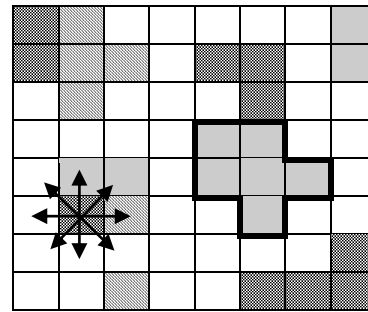
Fig. 2 A sample part of the world. There are three patch types, which show three species. The cells, surrounded by thick solid lines, shape one patch with surface area=6 and circumference=12. Neighbor cells of one cell are also showed by the arrows.

The genetic diversity of a species measures how much diversity exists in the gene pool of the individuals of a species and corresponds to the entropy of the set of genomes. The entropy measure is commonly used as an index of diversity in ecology and increasingly used in genetics [23].

We calculated some demographic features like number of species, ratio of individuals in one species to the whole population (popRatio), ratio of new born individuals to the whole population (birthRatio), average population per cell (popDensity), interbreedingRatio, which is the ratio of new born individuals with parents from two different species to the whole number of new born individuals, the ratio of prey killed by the predators to the total number of individuals (killedRatio), and the ratio of dead prey because of old age and low energy to the total number of individuals (deadAgeRatio and deadEnergyRatio respectively).

We also used several features related to individuals' actions, which show the percentage of the individuals in a species choosing one action such as reproduction (reproductionAction), search for food (SearchFoodAction), and eat (EatAction). In addition, some features related to perception were chosen depicting the individual perception of its environment such as distance to predator (distancePred), distance to food resource (distanceFood), distance to other preys (distancePrey), distance to partner (distancePartner), distance to predator (distancePred), etc. These features give some insight about the local environment properties of the individuals. Moreover, some features related to the age and energy of individuals were calculated. For instance, the average energy and the average age of the dead individuals (deathEnergy and deathAge respectively), and the average age (age) and the average energy (energy) of individuals can be mentioned. In addition, we used some features related to

mating like average reproduction age and energy of parents at the breeding time (parentReproduceAge and parentReproduce Energy). MatingDistance is the genetic distance between two parents' genome. Finally, stateofBirth was calculated, which indicates the amount of energy that prey invests in the breeding process.

We created a dataset using 49 features, each sample of the dataset shows the information about one species at a given time step. Species were classified based on a label feature that specifies if one species will split in the next 100 time steps. Similarly, there is a label feature to determine if one species will go extinct in the next 100 time steps. If the species split or become extinct within this time, the label is positive, otherwise it is negative. Therefore, we have similar features for classification, but the target is different for speciation and extinction's datasets.

Massive raw data, related to an average of 20 (5), 27 (9), and 32 (7) species and 126000 (32000), 179000 (27000), and 197000 (26000) individuals respectively, at any given time step, for the three runs of EcoSim (the values in parenthesis are standard deviation), were generated and analysied, representing 2200 hours of computation.

4   Classification using Machine-Learning Techniques

To build the train set, we mixed the dataset of two runs and after randomization selected 70000 records as the train set and the rest (about 480000 instances) for the test set. We mixed two runs hoping to obtain more generalized rules for classification. For better evaluation of the generalization ability of the classifier, we used the third run as a validation set.

For the speciation dataset, about 75 percent of the samples belonged to the negative class and the rest were in the positive class. It means that only 25 percent of species split in the next 100 time steps leading to a slightly imbalanced dataset. There are two main approaches to address the imbalanced data set problem [24]. One of them is to assign distinct costs to misclassified samples and try to minimize the overall cost on the training set. The second approach is re-sampling, either by under-sampling the major class or over-sampling the minor class. We used the over-sampling techniques, which consist in adding more samples of the minority class, because we had enough positive class samples when we merged two different runs to build the training set. Hence, we avoided the overfitting problem of oversampling technique because, instead of appending replicated data to the training set, we added some new positive instances. After examining different balances of the two classes, the dataset consisting of 50% positive and 50% negative class instances was selected because it gave the best results for the validation set. We did not change the initial class distribution in the test and validation sets to guaranty that the classifier had the capacity to classify the initial distribution of both classes' instances accurately. We did not have such a problem with the extinction dataset due to lifespan of species. Therefore,

we left the initial class distribution in the training set for the extinction prediction.

We applied the C4.5 [25] algorithm to build a decision tree based on features mentioned in section three for the training set. The benefit of using such an approach is that the obtained tree can be used for prediction, but also for extracting rules that can effectively determine the most important features in speciation and extinction.

To obtain the most important factors having influence on speciation and extinction, we used different feature selection algorithms like Greedy Stepwise, Ranker (using gain ratio evaluator), Genetic Search, and Best First implemented in Weka [26] along with 10-fold cross validation. Each algorithm provides a subset of the features and we picked the common features using a voting mechanism. We had three aims when applying the feature selection algorithms. The first was to capture important and effective features involved in speciation and extinction. The second was creating a model with a few numbers of parameters that was less prone to the overfitting problem. Finally, by using this method, along with pruning, we were able to reduce the size of the tree generated by C4.5 algorithm, which facilitated the interpretation of the rules predicting these two events. For all the experiments, we specified the pruning rate by setting the number of objects per leaf equal to 100.

The performance of a machine learning algorithm is typically evaluated by overall accuracy. However, it is not applicable for an imbalance dataset. For example, when there are 95% negative and 5% positive samples in a given dataset, the accuracy of one classifier that detects 100% of negative class and 0% of positive class will be 95%. In this case, the training algorithm mostly learns the major class (Negative class) while the minor class is highly important because in our case it shows the correct prediction of samples with a speciation or extinction. Consequently, simple overall accuracy is not a good measure to evaluate our classifiers performance. For evaluating the performance of these kind of classifiers, we used: true positive rate (TP Rate), false positive rate (FP Rate), area under ROC curve (AUC), and F-measure [27] in addition to the overall accuracy.

5   Experimental Results

In this section, we discuss the results of our experiments and also investigate the effect of the different features we used for speciation and extinction prediction.

5.1   Speciation

For the speciation prediction problem, first we used the original training set without changing the class distribution to build the classifier. The result of the classifier for the train, test, and validation sets is shown in Table 1.

TP Rate for minor class was low because the classifier tends to learn the samples from the majority class and almost ignore the ones from the minority class. Because the train and test sets were built from the mixture of result of two runs, their TP

rate were close, but for the validation set, it was about 12 percent less than for the train and the test sets. It shows that the generalization ability of the classifier is not good enough to classify species for other runs. The F-Measure value was also low for the validation set.

Table 1 Result of Speciation Prediction using Imbalanced Train Set

Data Set	TP Rate	FP Rate	F-Measure	AUC	Accuracy
Train	0.749	0.082	0.756	0.934	87.41%
Test	0.743	0.076	0.758	0.935	87.66%
Validation	0.625	0.082	0.679	0.917	83.64%

Applying the oversampling technique, explained in the section four, on the dataset, led to a train set with even class distribution. However, the class distribution of the test and validation sets remained unchanged. Afterward, we built the classifier and the result in Table 2 indicates that TP Rate improved by 21.2%, 21.2% and 30.7% for train, test, and validation set respectively. Using this method, we observed only 10% decrease in TN Rate, but on the other hand we improved the TP Rate about 25% on average. F-Measure improvement for validation set was about 7%.

Table 2 Result of Speciation Prediction using Balanced Train Set

Data Set	TP Rate	FP Rate	F-Measure	AUC	Accuracy
Train	0.908	0.168	0.875	0.932	87.00%
Test	0.901	0.154	0.736	0.933	85.83%
Validation	0.817	0.177	0.729	0.904	83.21%

Therefore, the classifier was able to predict the positive class, especially for validation set, with higher accuracy. In this case, the classifier was more generalized being able to classify species in a completely different run with a good accuracy. The last experiment for speciation was to use the most common features (13 out of 49 features in Table 5) chosen by different feature selection algorithms and removing the rest in the train, test and validation sets (Table 3). As it shows, we obtained an improvement in the TP Rate and F-measure, especially for validation set.

Table 3 Result of Speciation Prediction using Selected Features

Data Set	TP Rate	FP Rate	F-Measure	AUC	Accuracy
Train	0.911	0.184	0.870	0.924	86.36%
Test	0.916	0.191	0.705	0.925	83.20%
Validation	0.899	0.206	0.738	0.923	82.30%

5.2 Extinction

We made two experiments to study the impact of features on the prediction of species' extinction. Similarly to speciation prediction, first we used all the features to evaluate their potentiality for the prediction of extinction (Table 4). The overall accuracy for the train set was about 94% which demonstrates the strong capacity of prediction of extinction in the next 100 time steps.

Moreover, other evaluation criteria like TP-Rate, ROC, and F-

Measure acknowledged the quality of this prediction. This fact was also true for the test and the validation sets showing that a combination of demographic, environmental and genetic factors can predict the extinction of species with high accuracy even in different environmental conditions.

Table 4 Result of Extinction Prediction

Data Set	TP Rate	FP Rate	F-Measure	AUC	Accuracy
Train	0.942	0.075	0.941	0.971	94.16%
Test	0.942	0.073	0.942	0.971	94.20%
Validation	0.935	0.089	0.930	0.965	93.35%

Table 5 Selected Features for Speciation and Extinction

Selected Features	Category	Spec	Extinc
popRatio	Demography	✓	✓
interbreedingRatio	Demography		✓
killedRatio	Demography	✓	✓
deadAgeRatio	Demography	✓	
deathAge	Demography	✓	
birthRatio	Demography		
popDensity	Demography		✓
distancetoPrey	Environmental	✓	✓
searchFoodAction	Environmental	✓	✓
deadEnergyRatio	Environmental	✓	
distancetoPred	Environmental		
parent2_reproduceAge	Reproduction		✓
parent2_reproduceEnergy	Reproduction		✓
matingDistance	Reproduction	✓	✓
reproductionAction	Reproduction		✓
parent1_reproduceAge	Reproduction		✓
parent1_reproduceEnergy	Reproduction		✓
stateofBirth	Reproduction		✓
patchAreaRatio	Spatial	✓	✓
diversitySpatialRatio	Spatial	✓	
SC	Spatial	✓	
fractal dimension	Spatial	✓	
patch Circumference	Spatial	✓	✓

Afterward, based on our feature selection technique, the effect of the selected features on the accuracy of the prediction was studied. Using 15 selected features including demographics, genetics, and environmental features (Table 5), we obtained almost the same accuracy as when we applied all the features. In addition, the complexity of the classifier was reduced so that it also decreased the risk of overfitting and made the model easier to interpret.

5.3 Biological Evidence

Seven features were common in both experiments and others were specific for each experiment. Removing some features did not mean they were not effective on speciation or extinction; instead it can mean that they were covered by the selected features. To investigate this coverage, we extracted the dependencies between several features in both problems by applying Bays Net classifier. For example, in both speciation and extinction predictions, genetic diversity, which seems to be an important feature, can be replaced by

population ratio. This makes sense because increasing these two features makes a larger gene pool, which increases the speciation probability. On the other hand, decreasing these features leads to less variation in the gene pool and an excessive decline in it, causes extinction. Another example is in the spatial information category where patch area ratio can cover some features like SC and patch circumference.

To investigate the validity of obtained results based on real nature, we extracted several rules for both events prediction (Table 6; t values are thresholds for each feature. Hit ratio is percentage of samples that match to one rule and Accuracy is the accuracy of the rule on the matched samples for validation set). When patch area ratio is greater than a threshold, it shows the individuals of the species are more dispersed, which increases the possibility of speciation as discussed in [4].

Table 6 Several Samples of the Extracted Rules

Condition	Result	Hit Ratio	Accuracy
patchAreaRatio<= t_a	no speciation	45%	89%
patchAreaRatio> t_a	speciation	54%	82%
indvNoRatio<= t_i	no speciation	44%	90%
indvNoRatio> t_i	speciation	56%	81%
matingDistance > t_m	no extinction	31%	96%
matingDistance ≤ t_m	extinction	69%	92%
indvNoRatio > t_{i1}	no extinction	52%	98%
indvNoRatio ≤ t_{i2}	extinction	32%	94%

In both predictions, population ratio had a critical role. Having more individuals means a gene pool with higher variation. Therefore, one species with more individuals are more likely close to speciation. On the other hand, small populations are more prone to extinction because of demographic stochasticity as it has been shown in [28] or individuals may suffer reduced fitness from insufficient cooperative interactions with conspecifics or individuals may have difficulty encountering potential partners. These effects can cause negative growth rates of populations and lead to an unstable equilibrium at small population size below which the population more likely become extinct [29].

For the extinction rules based on population ratio, we found two thresholds (t_{i1}, t_{i2} where t_{i1}> t_{i2}), which indicate that between these two thresholds, population may go extinct or it may survive based on different factors. The mating distance indicates how genetically similar the two parents are. If the average mating distance of the population is less than a threshold, the species will go extinct. This feature can be seen as inbreeding factor. The lower value for this feature shows that mating has occurred between two individuals with very similar genetic material leading to inbreeding depression that decreases individual fitness and population growth rates as it has been observed in nature [30]. Moreover, Frankham [31] showed that inbreeding decreases the effective population size and can lead to extinction. This result is very important because the inbreeding effect is a fully emergent phenomenon of our system. It proves that inbreeding is an essential feature

of an ecosytem which is needed to maintain a high level of genetic diversity in a population.

In addition to these rules, Table 5 shows that spatial features were more effective on speciation while reproduction category mostly influenced extinction. In both speciation and extinction, demographic and environmental features were important.

6   Conclusion

In this study, we used an individual-based model, EcoSim, to investigate whether speciation and extinction are two predictable events. If that is true, what are the significant and effective factors in these two important phenomena? We computed 49 demographics, genetics, environment and spatial distribution features for the species observed in EcoSim and investigated how these features affect speciation and extinction. After adjusting the class distribution, using oversampling technique for the speciation problem but keeping the original class distribution for the extinction problem, we obtained very promising results. This means that the calculated features are effective in prediction of these two important phenomena and can help to understand them better. Moreover, using feature selection strategies, we were able to reduce the number of features to capture more precise information involved in speciation and extinction. Finally, these techniques helped to reduce the size of the tree generated by C4.5 algorithm, which facilitates the extraction of hypothesis for these two events for future work.

Afterward, we extracted several simple rules from the constructed decision tree. These rules are semantically clear and sound reasonable based on biological evidence. This is an important result as the proposed approach has proven to have the capability of generating realistic rules when compared with real biological data. Moreover, we have shown that complex important phenomena such as inbreeding can emerge from our system reinforcing its biological significance.

The next step will be the evaluation of some specific and finer grained hypothesis made by the biologists about species with the possibility to take into account many different situations and many different parameters.

Acknowledgments

References

[1] B. D. Mishler and M. J. Donoghue. "Species concepts: a case for pluralism ," *Systematic Zoology*, vol. 31, pp.491-503, 1982.

[2] E. Mayr. "Ecological Factors in Speciation," *Evolution*, vol. 1, pp.263-288, 1947.

[3] R. J. Safran and P. Nosil, "Speciation: The Origin of New Species," *Nature Education Knowledge,* vol. 3, 2012.

[4] E. Mayr. *Animal Species and Evolution*. Harvard, MA: Harvard University Press, 1963.

[5] M. Mashayekhi and R. Gras. "Investigating the Effect of Spatial Distribution and Spatiotemporal Information on Speciation using Individual-Based Ecosystem Simulation ," *Journal of Computing*, vol. 2, pp.56-62, 2011.

[6] M. G. Ritchie, "Sexual selection and speciation ," Annual Review of Ecology, *Evolution, and Systematics,* vol. 38, 2007.

[7] B. Griffen and J. Drake, "A review of extinction in experimental populations , " *The Journal of Animal Ecology*, vol. 77, pp. 1274-87, 2008.

[8] M.H. Sedehi, R. Gras, and Md. I.. "Prediction of Imminent Species' Extinction in EcoSim , " *In proc Agents and Artificial Intelligence*, 2012, p. 318.

[9] U. K. Rai, "Minimum Sizes for Viable Population and Conservation Biology ," *Our Nature* 1, 3-9, 2003.

[10] D. H. Reed, E. H. Lowe, D. A. Briscoe, and R. Frankham, "Inbreeding and extinction: effects of rate of inbreeding ," Conservation Genetics, vol. 4, pp.405-410, 2003.

[11] R. E. Lenski, C. Ofria, , T. C., Collier, and C. Adami, Genome complexity, robustness, and genetic interactions in digital organisms. Nature 400, pp. 661–664, 1999.

[12] L. Yaeger, 1992 Computational genetics, physiology, metabolism, neural systems, learning, vision, and behavior or PolyWorld: life in a new context. In Proc. Artificial Life III, vol. 17, pp. 263–298, 1992.

[13] D. L. DeAngelis, and W. M. Mooij, Individual-based modeling of ecological and evolutionary processes, Annu.Rev. Ecol. Evol. Syst. vol. 36, pp. 147–168, 2005.

[14] R. Gras, D. Devaurs, A. Wozniak, and A. Aspinall, "An individual-based evolving predator-prey ecosystem simulation using a fuzzy cognitive map as the behavior model , " *Artificial Life* vol. 15, pp. 423-63, 2009.

[15] B. Kosko. "Fuzzy cognitive maps , " *International Journal of Man-Machine Studies*, vol. 24, pp. 65-75, 1986.

[16] A. Aspinall and R. Gras, "K-means clustering as a speciation mechanism within an individual-based evolving predator-prey ecosystem simulation, " *In proc. Active Media Technology*, 2010, p. 318.

[17] D. Devaurs and R. Gras, "Species abundance patterns in an ecosystem simulation studied through Fisher's logseries , " *Simulation Modelling Practice and Theory*, vol. 18, pp. 100-123, 2010.

[18] A. G. Golestani and R. Gras. "Regularity analysis of an individual-based ecosystem simulation , " *Chaos* vol. 20, 043120, 2010.

[19] S. R. Jammalamadaka and A. SenGupta. *Topics in circular statistics*. World Scientific Pub Co Inc, 2001.

[20] H. L. and J. F. Reynolds. "A new contagion index to quantify spatial patterns of landscapes ," Landscape Ecology, vol. 8, pp.155-162, 1993.

[21] L. Parrott, R. Proulx, and X. T.Plante. "Three-dimensional metrics for the analysis of spatiotemporal data in ecology," *Ecological Informatics*, vol. 3, pp. 343-353, 2008.

[22] H.O. Peitgen, H. Jürgens, and D. Saupe. Chaos and Fractals. Springler-Verlag, New York, 1992.

[23] W. B. Sherwin, "Entropy and Information Approaches to Genetic Diversity and its Expression: Genomic Geography ," *Entropy*, vol. 12, pp. 1765-1798, 2011.

[24] H. He and E. A. Garcia. "Learning from Imbalanced Data. Knowledge Creation Diffusion Utilization, " vol. 21, pp.1263-1284, 2009.

[25] R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993.

[26] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P.eutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update , " *SIGKDD Explorations,* vol. 11, pp. 10-18, 2009.

[27] N. V. Chawla. Data Mining For Imbalanced Datasets: An Overview, Data Mining and Knowledge Discovery Handbook, Springer, Heidelberg, 2010.

[28] R. Lande, S. Engen, and B. E. Saether. *Stochastic Population Dynamics in Ecology and Conservation*. Oxford University Press, 2003.

[29] R. Lande, "Anthropogenic, ecological and genetic factors in extinction and conservation," *Researches on Population Ecolog*y vol. 40, pp.259-26, 1998.

[30] B. W. Brook, D.W. Tonkyn, J. J. O'Grady, and R. Frankham, "Contribution of inbreeding to extinction risk in threatened species , " *Conservation Ecology, vol.* 6,pp. 1-11, 2002.

[31] R. Frankham. "Genetics and extinction, " *Biological Conservation* vol. 126, pp.131-140, 2005.

Using Data Mining Techniques to Predict Product Quality from Physicochemical Data

A. Nachev[1], M. Hogan[1]
[1]Business Information Systems, Cairnes Business School, NUI, Galway, Ireland

Abstract - *Product quality certification is sometimes expensive and time consuming, particularly if it requires assessment made by human experts. This study explores experimentally how data mining techniques can facilitate that process. We use a dataset of physicochemical characteristics of red and white wine samples, available from laboratory tests, in order to build models that predict wine quality. Four data mining techniques are used: multilayer perceptrons, cascade-correlation neural networks, general regression neural networks, and support vector machines. We study how hyper-parameters of the models influence their predictive abilities and how reduction of dimensionality affects their performance. We also compare the models by the metrics prediction accuracy, mean absolute deviation, and area over the regression error characteristics curve.*

Keywords: data mining, neural networks, cascade-correlation neural networks, general regression neural networks, support vector machines.

1   Introduction

Today, with improvement of technologies, industries become more efficient and the production processes become quicker. In many cases, however, the human expertise is still essential for the product quality assurance process. With increase of demand for goods, quality certification becomes an expensive step in the production process.

The aim of this study is to explore the potential of four predictive techniques: neural networks (NN) a.k.a. multilayer perceptrons (MLP), cascade-correlation neural networks (CCNN), general regression neural network (GRNN), and support vector machines (SVM), to facilitate the quality certification of a product, based on available product characteristics. This would allow automating the process and minimizing usage of human expertise.

Here we focus on the wine quality prediction using data from both physicochemical laboratory tests and sensory tests. Wine is usually characterised by density, alcohol or various acids, which can be obtained by lab tests, while sensory tests are done by human experts. Wine classification is not an easy task as the relationships between physicochemical analysis and sensory tests analysis are complex and not well understood [12].

Predicting wine quality by data mining techniques is still in an early stage, but there are some promising results in the domain. Sun et al. [16] used NNs fed with 15 input variables used to predict six geographic wine origins. The data included 170 samples. Vlassides et al. [19] used NNs to classify three sensory attributes (e.g. sweetness) of Californian wine, based on grape maturity levels and chemical analysis. Moreno et al. [13] used probabilistic neural networks (PNN) to discriminate 54 wine samples into two red wine classes. Yu et al. [20] used spectral measurements from 147 bottles of rice wine to predict 3 categories of wine. Fei et al., [8] utilized least squares support vector machines on physicochemical data of red wine samples. These chemometrics were obtained through the use of visible and near infrared (Vis/NIR) transmittance spectroscopy. Beltran et al. [5] utilize SVM in addition to, and in comparison with, radial basis function neural networks (RBFNN) and linear discriminant analysis (LDA), in the classification of Chilean wine. The analyses are carried out on data derived from wine aroma chromatograms of three different Chilean wine varieties. Bapna and Gangopadhyay [4] and Cortez et al. [6] compared several data mining techniques for classification of wine.

In this paper, we estimate performance of NN, CCNN, GRNN, and SVM in predicting red and wine quality based on 11 physicochemical characteristics and explore how model hyper-parameters influence their ability to discriminate between quality classes.

The paper is organized as follows: Section 2 provides an overview of the multilayer perceptrons, cascade-correlation neural networks, general regression neural network, and support vector machines used build a predictive models; Section 3 discusses the dataset used in the study, its features, preprocessing steps, and feature selection; Section 4 presents and discuses the experimental results; and Section 5 gives the conclusions.

2   Data Mining Models

We adopt four predictive techniques: the most common NN type - MLP, cascade-correlation neural networks, general regression neural networks, and support vector machines. This section outlines briefly each of those.

2.1   Multilayer Perceptrons

An MLP is a feedforward NN model that maps sets of input data onto a set of appropriate output, either values or class labels. It uses three layers of neurons, called nodes (see Figure 1), with nonlinear activation functions that can

distinguish non-linearly separable data, or separable by a hyperplane. Nodes of two adjacent layers are fully connected by weighted links represented by matrices *IW*, *LW*, and bias vectors *b*. The two activation functions are both sigmoids:

$$f_H(x) = \frac{e^{2x}-1}{e^{2x}+1} \quad f_L(x) = \frac{1}{1+e^{-\beta x}} \ , \qquad (1)$$

where f_H is a hyperbolic tangent which ranges from -1 to 1; f_L is a log-sigmoid function, equivalent in shape, but ranges from 0 to 1. Here x is the weighted sum of the inputs. Finding an optimal size of the hidden layer is a general problem with all MLP. We used the heuristic:

$$n_h = \frac{n_s}{\alpha(n_i + n_o)} \qquad (2)$$

where n_h is the size of the hidden layer; n_s is the number of training samples; n_i and n_o are the size of the input and output layers respectively; $\alpha \in [5, 10]$ is a scaling factor, smaller of noisy data and larger for relatively less noisy data. The network was trained by Levenberg-Marquardt (LM) backpropagation (BP) algorithm [9]. LM is a second-order nonlinear optimization technique that uses an approximation to the Hessian matrix. It was chosen from the various BP training algorithms as it trains a moderate size NN 10 to 100 times faster than the usual gradient descent backpropagation method and produces better results.
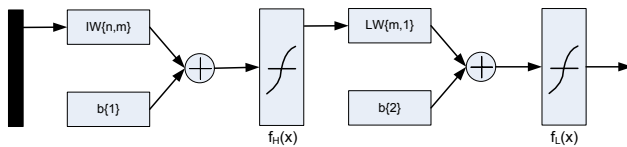
Figure 1. Architecture of an MLP neural network with one hidden layer.

2.2 Cascade-Correlation Neural Networks

CCNN are self-growing neural networks similar to MLP, but they don't have fixed size or topology [2]. The CCNN have three layers: input, hidden, and output, similarly to MLP. The output layer consists of a single node if the network is used for regression problems, or contains several nodes for classification problems, one per class label. In contrast to MPL, the CCNN start training without hidden layer - input nodes are fully connected to the output nodes with adjustable weights. During the training, the network adds new hidden nodes. It creates a multi-layer structure called a 'cascade', because the output from all input and hidden nodes existing already in the network, feed new nodes. In the beginning,every input is connected to every output neuron by a connection with an adjustable weight. The network adds new hidden nodes one by one (Figure 2) until the residual error gets acceptably small or the user interrupts this process.
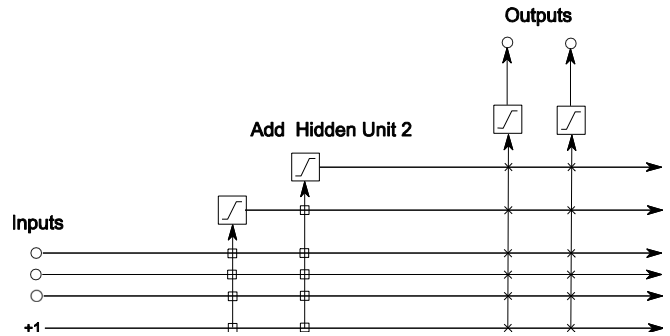
Figure 2. Cascade architecture after adding two hidden nodes (adapted from [2]). The vertical lines sum all incoming activation. Boxed connections are frozen, 'x' connections are trained repeatedly.

To install a new hidden neuron, instead of a single candidate, the system uses a pool of trainable candidate nodes (usually four to eight), each with a different set of randomly selected initial weights. All candidates receive the same input signals and see the same residual error for each training pattern, but because they are not installed yet and do not interact with one another or affect the active neural network during training, all of these candidate units are trained in parallel; when no further progress is being made in training, the network installs the candidate whose score is the best (minimises the residual error). The use of this pool of candidates is beneficial in two ways: it greatly reduces the chance that a useless unit will be permanently installed, and it speeds up the training because many parts of weight-space can be explored simultaneously.

While the candidate weights are being trained, none of the weights in the active network are changed. Once a new hidden node has been added to the network, its input-side weights (boxed connections in Figure 2) are frozen; the output-side connections ('x' connections) continue to be adjustable. The learning algorithm modifies the weights attempting to minimize the residual error of the network. Each new neuron becomes a permanent feature-detector in the network, available for producing outputs or for creating other, more complex feature detectors.

Among advantages of CCNN can be mentioned self-organizing architecture, quick learning, applicable to large datasets, obtaining good results with little or no adjustment parameters and less chance to get trapped in local minima, compared to the MLP. They have, however, a significant potential for overfitting the training data, which results in a very good accuracy on the training dataset but not always good accuracy on new, unseen during the training data.

2.3 General Regression Neural Networks

The GRNN are a kind of radial basis function (RBF) NN proposed by Specht [15]. They are a powerful regression tool, which features simple structure and implementation and fast training. A GRNN consists of four layers: input, hidden, summation, and output (Figure 3). The function of the input layer is to pass the input values x_i to the hidden layer.
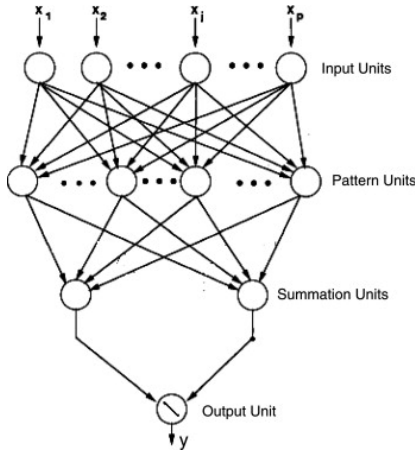
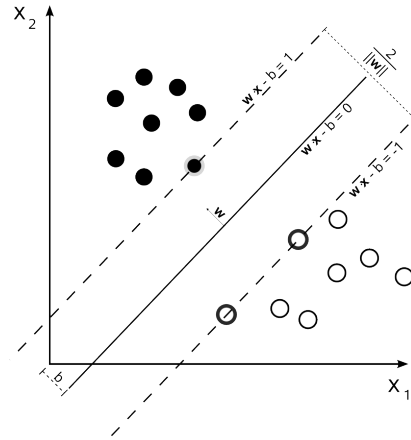*Figure 3. GRNN architecture: feed-forward NN with input, hidden, summation, and output layers.*

Figure 4. Maximum–margin hyperplane for a SVM trained with samples from two classes. Samples on the margin are support vectors.

The hidden layer consists of all training patterns X_i. When an unknown pattern X is presented to the network, the squared distance $D_i^2 = (X - X_i)^T(X - X_i)$ between the X and each X_i is calculated and passed to the kernel function. The summation layer has two nodes (units), A and B, where A computes the summation function, which is numerator of (3), and B computes the denominator.

$$Y(X) = \frac{\sum_{i=1}^{n} Y_i \exp(\frac{-D_i^2}{2\sigma^2})}{\sum_{i=1}^{n} \exp(\frac{-D_i^2}{2\sigma^2})} \; , \qquad (3)$$

where σ is the width of the kernel. The output node computes A/B, which is Y.

2.4 Support Vector Machines

SVM, originally introduced by Vapnik in 1990s [17], provide a new approach to the problem of pattern recognition with clear connections to the underlying statistical learning theory. They differ radically from comparable approaches such as NN because SVM training always finds a global minimum in contrast to NN [18]. SVMs are supervised learning methods used for classification and regression. Training data is a set of points of the form

$$D = \{(x_i, c_i) \mid x_i \in \Re^p, \; c_i \in \{-1, 1\}\}_{i=1}^{n} \qquad (4)$$

where the c_i is either 1 or −1, indicating the class to which the point x_i belongs. Each data point x_i is a p-dimensional real vector. During training a linear SVM constructs a $p-1$-dimensional hyperplane that separates the points into two classes (Figure 4). Any hyperplane can be represented by $w \cdot x - b = 0$ where w is a normal vector and $\cdot$ denotes dot product. Among all possible hyperplanes that might classify the data, SVM selects one with maximal distance (margin) to the nearest data points (support vectors).

When the classes are not linearly separable (there is no hyperplane that can split the two classes), a variant of SVM, called soft-margin SVM, chooses a hyperplane that splits the points as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. The method introduces slack variables, ξ_i, which measure the degree of misclassification of the datum x_i. Soft-margin SVM penalizes misclassification errors and employs a parameter (the soft-margin constant C) to control the cost of misclassification. Training a linear SVM classier solves the constrained optimization problem (5).

$$\min_{w, b, \xi_k} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \qquad (5)$$

$$s.t. \quad w \cdot x_i + b \geq 1 - \xi_i$$

In dual form the optimization problem can be represented by:

$$\min_{\alpha_i} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i \cdot x_j - \sum_{i=1}^{n} \alpha_i \qquad (6)$$

$$s.t. \quad 0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{n} \alpha_i c_i = 0$$

The resulting decision function $f(x) = w \cdot x + b$ has weight vector $w = \sum_{k=1}^{n} \alpha_k y_k x_k$. Data points x_i for which $\alpha_i > 0$ are called support vectors, since they uniquely define the maximum margin hyperplane. Maximizing the margin allows one to minimize bounds on generalization error.

If every dot product is replaced by a non-linear kernel function, it transforms the feature space into higher-dimensional, thus though the classifier is a hyperplane in the high-dimensional feature space it may be non-linear in the original input space. The resulting classifier fits the

maximum-margin hyperplane in the transformed feature space. Some common kernels include:

- Polynomial kernel $K(\vec{x}_i, \vec{x}_j) = (\gamma \vec{x}_i^T \vec{x}_j + r)^d$

- RBF kernel $K(\vec{x}_i, \vec{x}_j) = \exp(\gamma \|\vec{x}_i - \vec{x}_j\|^2)$

- Sigmoid kernel $K(\vec{x}_i, \vec{x}_j) = \tanh(\gamma \vec{x}_i^T \vec{x}_j + r)$

A non-linear SVM is largely characterized by the choice of its kernel, and SVMs thus link the problems they are designed for with a large body of existing work on kernel based methods. Once the kernel is fixed, SVM classifiers have few user-chosen parameters. The best choice of kernel for a given problem is still a research issue. Because the size of the margin does not depend on the data dimension, SVM are robust with respect to data with high input dimension. However, SVM are sensitive to the presence of outliers, due to the regularization term for penalizing misclassification (which depends on the choice of C). The SVM algorithm requires $O(n^2)$ storage and $O(n^3)$ to learn.

3   Dataset and Preprocessing

The data used in this study represent wine sample collection of Vinho Verde wines, white and red (CVRVV, 2008), which consists of two distinct sets made up of 4898 white and 1599 red samples. Each instance consists of 12 physiochemical variables: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, and a quality rating. The quality rating is based on a sensory taste test carried out by at least three sommeliers and scaled in 11 quality classes from 0 - very bad to 10 - very excellent. A summary the datasets is presented in Table 1.

Table 1: The physicochemical data statistics per wine type

Attributes	Red Wine			White wine		
	Min	Max	Mean	Min	Max	Mean
Fixed acidity	4.6	15.9	8.3	3.8	14.2	6.9
Volatile acidity	0.1	1.6	0.5	0.1	1.1	0.3
Citric acid	0.0	1.0	0.3	0.0	1.7	0.3
Residual sugar	0.9	15.5	2.5	0.6	65.8	6.4
Chlorides	0.01	0.61	0.08	0.01	0.35	0.05
Free sulpher dioxide	1	72	14	2	289	35
Total sulfur dioxide	6	289	46	9	440	138
Density	0.990	1.004	0.996	0.987	1.039	0.994
pH	2.7	4.0	3.3	2.7	3.8	3.1
Sulphates	0.3	2.0	0.7	0.2	1.1	0.5
Alcohol	8.4	14.9	10.4	8.0	14.2	10.4

Using the data in their original format for building models is inappropriate due to some deficiencies. A specific problem is the large amplitude of the variable values due to the different nature and different units of measurements of those values, e.g. sulfur dioxide (1 – 72) vs. sulfates (0.3 – 2). Such an inconsistency could affect the predictive abilities of the models by making some variables more 'influential'

than others. Moreover, some models require inputs within the unit hypercube, i.e. between 0 and 1. A natural approach of meeting that requirement could be a linear transformation that divides all input values by the dataset maximum, however mostly of the input values will fall very close to zero, and the model would perform poorly. A better approach is to process each data variable (data column) separately. We did so by using the transformation

$$x_i^{new} = \frac{x_i^{old} - \min_i}{\max_i - \min_i}, \qquad (7)$$

which scales down the variables within the unit hypercube.

Another problem with utilizing the original data without preprocessing is that using all features of a dataset does not always lead to best or even satisfactory results. This is due to the fact that too much information used for both tratning and testing can lead to overfitting or overtraining. We explored how presence or absence of variables presented to the model for training and testing affects the performance.

Variable selection, or reduction of dimensionality, is a technique commonly used in machine learning for building robust learning models. Removing most irrelevant and redundant features from the data usually helps to alleviate the effect of the curse of dimensionality and to enhance the generalization capability of the model, yet to speed up the learning process and to improve the model interpretability. The variable selection also helps to acquire better understanding about data and how they are related with each other. Dimensionality reduction is considered as an application-specific problem, which is not backed by a universal theory. The exhaustive search approach that considers all possible subsets is the best strategy applicable for datasets with small cardinality, but impractical for large number of features, as our case is.

There are two distinct groupings of variable selection algorithms, specifically wrapper methods and filter methods. The wrapper methods employ the feature subset selection algorithm in unison with an induction algorithm. The selection algorithm proceeds to unearth a favorable subset of data whilst using this induction algorithm to evaluate proposed subsets. The filter methods use a preprocessing step and autonomously select variables independent of the induction algorithm. There are a number of algorithms that fall under the umbrella of the filter approach, such as the relief algorithm, which assigns a weighting of relevance to each feature, that is, the relevance of the selected variable to the target output; and the decision tree algorithm, which is used to select feature subsets for the nearest neighbor algorithm [11].

Rueda et al. [14] highlight a particular strength possessed by wrapper algorithms. The authors state that if variables are highly correlated with the response, the filter algorithm would typically include them, even if they diminished the overall algorithm performance. While in the wrapper approach, the induction algorithm may discover these diminishing effects, and exclude them [3].

4 Empirical Results

Using the data described above, we built and tested a number of predictive models, based on the four techniques - SVM, CCNN, GRNN, and NN.

In order to estimate the models performance we used the following metrics:

- *Prediction accuracy ACC_t* at certain error tolerance values t = 0.25, 0.5, 1, and 2.
- *Mean Absolute Deviation* (MAD), which is a robust performance measure of the model variability [1]

$$MAD = \frac{1}{N} \sum_{i=1}^{N} |y_i - \widehat{y}_i|, \qquad (8)$$

where y_i and $\hat{y}_i$ are the class label and predicted value, respectively.

- *Area over the regression error characteristic curve*. The regression error characteristic (REC) curve plots the error tolerances along the horizontal axis versus the prediction accuracy on the vertical. The area over the REC curve (AOC) is a scalar value that estimates the overall model performance regardless of the error tolerance values applied to each model instance. The lower the AOC is, the better it performs.

We tested the models with a number of hyper-parameters in order to find their optimal values and ensure maximal performance. Avoiding bias in training and testing, we applied 5-fold cross-validation. The dataset was divided on five subsets, each of which is 20%. The overall performance estimation metrics were calculated using each of those 20% for testing after training the model on the remaining 80% of data.

In order to estimate the influence of reduction of data dimensionality in the model performance, we applied wrapper and filter attribute evaluator methods outlined above. These methods combined differing search techniques, which resulted in combinations of proposed variable subsets. Models were tested with different combinations of variables and the results were compared by the aforementioned metrics. The results obtained were different for the red wine and the white wine datasets.

In the red wine case, the best results were obtained by the chi-squared attribute evaluation technique [10], which calculates chi-squared worth of each attribute with respect to the class. Results are summarized in Table 2 The experiments showed that chi-squared worth cut-off point between 169.86 and 145.40 performs best, which resulted in four red wine attributes used in training and testing the models, namely alcohol, volatile acidity, sulphates, and citric acid.

We found that the optimal SVM parameters used to produce a minimal mean squared error (MSE) of the model are: c=1.398; ε =0.746; kernel=polynomial; d=1; γ =0.572; and r=0.530.

Results obtained from the error tolerance study of the models are compared by REC curves in Figure 5. It is the model with the least area over the curve (AOC) that is most accurate, with the point closest to the 100% accurate and zero threshold intersection, indicating the best threshold level of the model. Figure 5 is quantitatively summarised in the Table 3.

Table 2 Chi-squared attribute evaluation for red wine.

Attribute	Chi-squared worth	Percentage importance
Alcohol	497.7464	29.61
Volatile acidity	354.4793	21.09
Sulphates	252.0535	15.00
Citric acid	169.8607	10.11
Total sulphur dioxide	145.3958	8.65
Density	130.73	7.78
Chlorides	82.6207	4.92
Fixed acidity	48.0288	2.86
pH	0	0.00
Residual sugar	0	0.00
Free sulphur dioxide	0	0.00
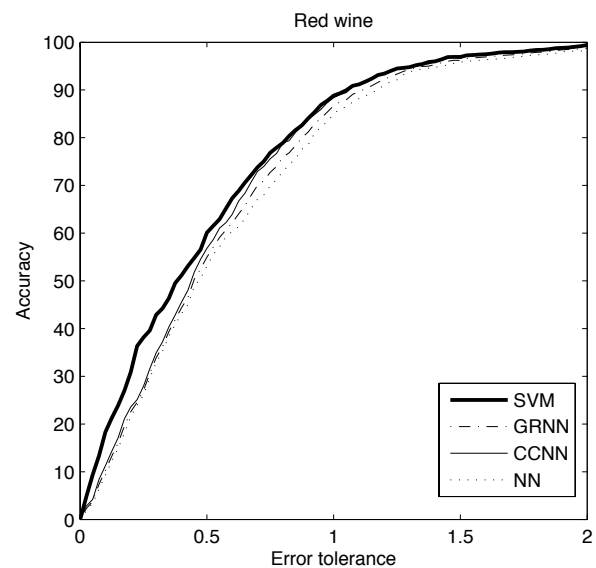
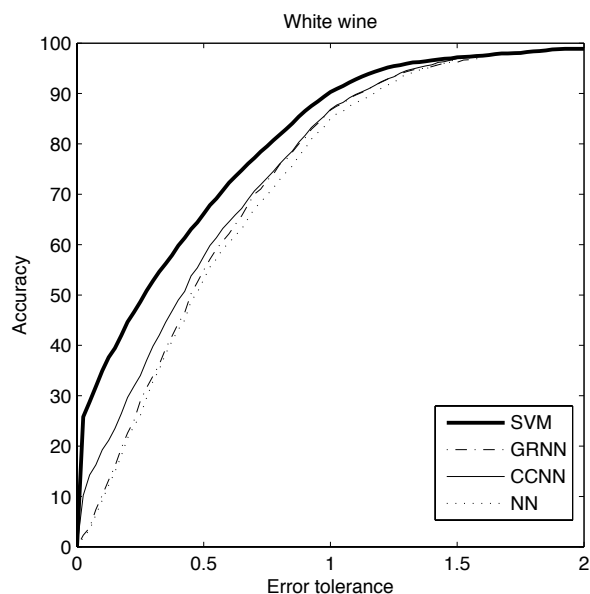*Figure 5. REC curves of red wine test set. SVM - tick solid line; CCNN - thin solid line; GRNN - dash-dot line; NN - dot line.*

Table 3 Performance of red wine quality prediction models. Estimation metrics include: accuracy at certain error tolerances (ACCt), mean absolute deviation (MAD), and area above the REC curve (AOC).

	$ACC_{0.25}$	$ACC_{0.5}$	ACC_1	$ACC_{1.5}$	ACC_2	MAD	AOC
ANN	0.261	0.531	0.850	0.958	0.981	0.592	0.662
CCNN	0.279	0.568	0.884	0.966	0.983	0.548	0.630
GRNN	0.27	0.549	0.867	0.962	0.979	0.577	0.644
SVM	0.381	0.601	0.888	0.969	0.994	0.496	0.506

It should be noted that according to the metrics MAD and AOC, SVM outperform all other models. They show clear advantage in the low error tolerance ranges where direct hits in predictions is important, or one-away hits, where error tolerance less than 0.5 is acceptable. When error tolerance increases and requirement for correct classifications relaxes, CCNN networks become equally good to SVM. Last in performance is the classic feed-forward NN and the second last is GRNN, which is between NN and CCNN.

Similarly, we explored dimensionality reduction in the white wine case. Results showed that best technique for ranking attributes is symmetrical uncertainty ranking, which is one of the most effective entropy-based feature selection approaches. Experimentally we found that alcohol content in white wine bears most importance (26.47%); density ranks second in importance (19.19%); with chlorides following next (14.35%). Total sulfur dioxide, citric acid, free sulfur dioxide and volatile acidity complete the model, all registering close importance percentage between 9.9% and 10.4%. Results are summarized in Table 4.

Table 4 Symmetrical uncertainty attribute evaluation for white wine.

Attribute	Symmetrical Uncertainty	Percentage importance
alcohol	0.08998	26.46
density	0.06524	19.18
chlorides	0.04878	14.34
total sulphur dioxide	0.03513	10.33
citric acid	0.03468	10.20
free sulphur dioxide	0.03376	9.92
volatile acidity	0.03241	9.53

Figure 6. REC curves of red wine test set. SVM - tick solid line; CCNN - thin solid line; GRNN - dash-dot line; NN - dot line.

Findings also show that with white wine, SVM performs best with c=2.438; ε =0.684; kernel=polynomial; d=1; γ =1.266; and r=1.522.

Figure 6 graphically compares the models performance in the terms of REC and Table 5 summarizes the estimation metrics. Similarly to the red wine case, the white wine results show that SVM outperforms the three neural networks, with even higher accuracy in the low error tolerance values, but it also outperforms the other models in higher error tolerance values (between 0.5 and 1.5).

The three neural network models show similar performance with little advantage of CCNN over GRNN and the classic NN. In a relatively large error tolerance (above 1), CCNN and GRNN perform similarly and slightly better than NN.

Table 5 Performance of white wine quality prediction models. Estimation metrics include: accuracy at certain error tolerances (ACCt), mean absolute deviation (MAD), and area above the REC curve (AOC).

	$ACC_{0.25}$	$ACC_{0.5}$	ACC_1	$ACC_{1.5}$	ACC_2	MAD	AOC
ANN	0.261	0.531	0.850	0.968	0.988	0.594	0.658
CCNN	0.339	0.576	0.868	9.969	0.988	0.514	0.581
GRNN	0.290	0.549	0.867	0.962	0.988	0.589	0.630
SVM	0.486	0.661	0.902	0.971	0.988	0.477	0.566

Finally, it can be summarized that SVM could be a better alternative of prediction models based on neural networks for application areas, like the one explored here. At the same time, certain neural network types, such as CCNN and GRNN can be considered as good candidates for predicting models, both outperforming the classic neural network.

5   Conclusions

Recently, wine industry expands its marketplace, which encourages adoption of advanced technologies in the production process. The quality certification is an important step in that. Traditionally, it is based on sensory tests carried out by human experts. This, however, is not as efficient as needed, because the procedure is time consuming and expensive. Data mining may help in the quality certification by processing physicochemical laboratory test data and building models that predict product quality classes. Various modeling techniques can be applied to solve the task and each of them shows specific performance characteristics.

The goal of this study is to explore how model hyper-parameters of the classic backpropagation neural network, cascade-correlation neural network, general regression neural network, and support vector machine, affect their predictive abilities in solving that task. We used an existing data set of 1599 red wine samples, and 4898 white wine samples, each of which consisting of 11 physicochemical characteristics. In order to quantify the model performance, we used metrics, such as prediction accuracy, mean absolute deviation, and area over the regression error characteristics curve. Our findings show that support vector machine with polynomial kernel outperforms the three neural network

models in all the metrics. The SVM advantage can clearly be seen with small values of error tolerance, that is where predicted quality is required to be very close to the real one. From another hand, the CCNN and GRNN show similar performance with little advantage of the CCNN over GRNN. Last in ranking is the classic NN, which despite its popularity as classification and regression tool, is not the best choice in this application domain. We also tested how various techniques for reduction of dimensionality influence the models performance. Empirically we found that best variable set selection techniques are chi-squared attribute evaluation and symmetrical uncertainty ranking for the red and white wine, respectively.

6   References

[1]   J. Bi and K. P. Bennett. Regression error characteristic curves. In Proceedings of the 20th International Conference on Machine Learning, 2003.

[2]   Fahlman, S. and Lebiere C. "The Cascade-Correlation Learning Architecture" in D. S. Touretzky (ed.), Advances in Neural Information Processing Systems 2, Morgan Kaufmann, 1990.

[3]   Guetlein, M., Frank, E., Hall, M., Karwath, A. "Large Scale Attribute Selection Using Wrappers"; In Proc. IEEE Symposium on CIDM, pp.332-339, 2009.

[4]   Bapna, S. and Gangopadhyay, A. A Wavelet-Based Approach to Preserve Privacy for Classification Mining. Decision Sciences, 37, 623-642, 2006.

[5]   Beltran, N. H., Duarte-Mermoud, M. A., Soto Vicencio, V. A., Salah, S. A. & Bustos, M. A. Chilean Wine Classification Using Volatile Organic Compounds Data Obtained With a Fast GC Analyzer. IEEE Transactions on Instrumentation and Measurement, 57, 2421-2436, 2008.

[6]   Cortez, P., Cerdeira, A., Almeida, F., Matos, T. & Reis, J. Modeling wine preferences by data mining from physicochemical properties. Decision Support Systems, 47, 547-553, 2009.

[7]   CVRVV. Portuguese Wine - Vinho Verde. Comissão de Viticultura da Região dos Vinhos Verdes (CVRVV), http://www.vinhoverde.pt, July 2008.

[8]   Fei, L., Li, W. & Yong, H. Application of least squares support vector machines for discrimination of red wine using visible and near infrared spectroscopy. Intelligent System and Knowledge Engineering, ISKE' 08, 2008.

[9]   Hagan, M. T., and Menhaj, M. B. Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks, 5, 989-993, 1994.

[10] Hall, M., Frank, E., Holmes, G., Fahringer, B., Reuteman, P. & Witten, I. H. The WEKA Data Mining Software: An Update. SIGKDD *Explorations,* 11, 2009.

[11] Kohavi, R. & John, G. H. 1997. Wrappers for feature subset selection. Artificial Intelligence, 97, 273-324, 1997.

[12] Legin, A., Rudnitskaya, A., Luvova, L., Vlasov, Y.,Natale, C., and D'Amico, A. Evaluation of Italian wine by the electronic tongue: recognition, quantitative analysis, and correlation with human sensory perception. Analytica Chimica Acta, 33-34, 2003.

[13] Moreno, I., D. Gonzalez-Weller, V. Gutierrez, M. Marino, A. Camean, A. Gonzalez, and A. Hardisson. Differentiation of two Canary DO red wines according to their metal content from inductively coupled plasma optical emission spectrometry and graphite furnace atomic absorption spectrometry by using Probabilistic Neural Networks. Talanta, 72:263–268, 2007.

[14] Rueda, I. E. A., Arciniegas, F. A. & Embrechts, M. J. SVM sensitivity analysis: an application to currency crises aftermaths. IEEE Transactions on Systems, Man and Cybernetics, 34, 387-398, 2004.

[15] Specht, D. Enhancement to probabilistic neural networks. In: Proceedings of the International Joint Conference on Neural Networks, vol.1, pp. 761-768, 1991.

[16] Sun, L., K. Danzer, and G. Thiel. Classification of wine samples by means of artificial neural networks and discrimination analytical methods. Fresenius' Journal of Analytical Chemistry, 359:143–149, 1997.

[17] Vapnik, V., The Nature of Statistical Learning Theory. Springer, New York, 1995.

[18] Vapnik, V., Kotz, S., Estimation of Dependences Based on Empirical Data, Springer, New York, 2006.

[19] Vlassides, S., J. Ferrier, and D. Block. Using Historical Data for Bioprocess Optimization: Modeling Wine Characteristics Using Artificial Neural Networks and Archived Process Information. Biotechnology and Bioengineering, 73(1), 2001.

[20] Yu, H. Lin, H. Xu, Y. Ying, B. Li, and X. Pan. Prediction of Enological Parameters and Discrimination of Rice Wine Age Using Least-Squares Support Vector Machines and Near Infrared Spectroscopy. Agricultural and Food Chemistry, 56:307–313, 2008.

Investigation of Cluster Validity Indices for Unsupervised Human Activity Discovery

Wee-Hong Ong[1], **Takafumi Koseki**[1] and **Leon Palafox**[2]

[1]Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo, Japan
[2]Department of Radiology, University of California, Los Angeles, Los Angeles, USA

Abstract - *An approach for unsupervised human activity discovery has been proposed in this paper. The approach automatically discover unknown activities from unlabeled data and has the ability to reject random activities. This ability will enable intelligent systems to discover and learn new activities autonomously. K-means is used to cluster a pool of unlabeled activity observations into groups of different activities. The system requires no prior knowledge of how many activities to be discovered. It uses cluster validity indices to automatically estimate the required number of clusters and further evaluate cluster homogeneity to accept clusters with homogenous activity and reject clusters with random activities. Experimental results showed the potential of the approach and identified suitable validity indices to achieve unsupervised human activity discovery.*

Keywords: human activity detection; human activity discovery; unsupervised learning; clustering; RGBD sensor

1   Introduction

From the detailed survey on human activity analysis by Aggarwal and Ryoo [4], we can see that there have been significant efforts to accurately capture human motion, i.e., solving computer vision problem and recognize the activity from the motion, i.e., solving modeling and learning problems. Systems with high accuracy in capturing human motion are expensive and require infrastructure setup. Many such systems require markers attached to the subjects. Further, those systems use supervised learning algorithms to learn models of activities. In these systems, pre-labeled examples are provided to the learning algorithm. However, to enable intelligent systems to autonomously learn new activities, they will be required to deal with unlabeled data. For this reason, there have been increasing interest to investigate human activity discovery using unsupervised learning. In the knowledge of the authors, most of the works relating unsupervised learning with activity recognition have been either focusing on solving computer vision problems or they require alternative form of pre-labeled data from wearable sensors or other sources. For examples, Song et al. [17] developed an EM-like algorithm on decomposable triangulated graphs to extract human as

foreground from background clutter. Huynh et al. [12] used clustering to generate a vocabulary of labels from sensor data, which are then used for pattern extraction using topic models to recognize daily routines. They used data from custom made wearable sensors. Stikic et al. [9] applied two weakly supervised methods to discover activities from two published datasets obtained from wearable sensors. Wyatt et al. [2] described their techniques for mining object models from the web and use the information to recognize activities based on the interaction of user with objects. They attached RFID tags to the objects. We observed that current state-of-art of human activity recognition technologies is not cost effective and not suitable in our natural living environment. They either require expensive setup, wearable sensors or pre-labeled data.

Our work is motivated by the intend to create a fully autonomous personal intelligent agent, for example a personal robot, that is capable of understanding what its user, or owner, is doing and consequently provide appropriate support. Therefore, human activity recognition lies at the core of such system. The system should be able to be deployed in the natural human living environment with minimal changes, at low-cost and work with unlabeled data. Further, the system should not require users to wear markers or sensors. These requirements call for the use of low-cost components, lightweight algorithms, marker-less vision and/or audio sensors, and unsupervised learning. Recently, the availability of low-cost RGBD (RGB-Depth) sensors has enabled accurate capture of human poses. In our earlier work [15], [16], we demonstrated the ability of K-means to distinguish different activities using just the skeleton data obtained from low-cost RGBD sensor, Microsoft Kinect. The problems with the use of K-means are the need to specify the number of clusters, *k* value, apriori and such system do not have the ability to reject random activities. Random activities will be assigned to a cluster anyway. In this paper, we propose an approach to address these two problems. We propose an approach to perform unsupervised human activity discovery. The approach uses cluster validity indices to evaluate the quality of clustering outcome and automatically determines suitable number of clusters and evaluates each cluster to identify highly homogeneous clusters as new activities.

2   Proposed approach

Assuming an intelligent system or agent has been observing daily activities of a subject. An example situation is a personal robot accompanying its owner. The system has collected a large pool of unlabeled activity observations (data). To be low cost with minimal requirements, we allow the system to be a best-effort one. The approach resembles how children learn activities around them. Children have the ability to distinguish different activities and ask adults to label the activities, i.e., they learn from unlabeled data (observations) and post-label the model they have formed. Children can learn from one subject, e.g., a parent, and adapt the model to different subjects. While there are many activities going on, children don't learn all of them at once. Finally, there are times that children are confused with similar activities.
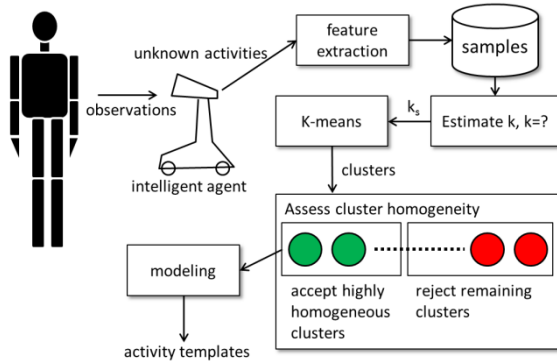
Figure 1. Proposed approach for unsupervised activity discovery in human activity recognition.

We propose the following steps to autonomously discover new activities, as illustrated in Fig. 1. The different cluster validity indices are described in Section 3.

1. Collect observations of unknown activities.
2. Use global internal cluster validity index to estimate the number of clusters, k.
3. Do clustering using the suggested number of clusters, k_s, from above.
4. Assess homogeneity of each of the k_s clusters using local internal cluster validity measure.
5. Accept high ranking clusters and reject clusters ranked low by the validity measure.

3   Unsupervised learning

3.1   K-means clustering

K-means [5] clustering is one of the simplest unsupervised learning algorithms. It looks for similarity among the examples in the dataset by using simple distance measurement. Given the required number of clusters, k, K-means group the points (examples) in the dataset by minimizing the distance from each data point to a cluster center (centroid). We have used K-means clustering that minimizes the regular cost function given in Eq. (1).

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| x_i^{(j)} - c_j \right\|^2 \tag{1}$$

where k is the number of clusters, $x_i^{(j)}$ is ith data point in Cluster j and c_j is the centroid of Cluster j, i.e., C_j, n_j is the number of data points in C_j. Note that each data point is a row vector in all equations presented in this paper.

3.2   Cluster validation

We expect the intelligent system to autonomously discover new activities and the number of activities to be found is unknown to the system. Various cluster validity indices [13] have been proposed to assess the quality of clustering outcome and determine the appropriate k. Since we will deal with unlabeled data, we consider only internal cluster validity indices that do not require labeled data. In this paper, we tested five indices as given below.

1. Silhouette (Sil) [10]. For each data point, $a(x_i)$ is the average distance from the point to other points in its cluster, and $b(x_i)$ is the average distance from it to all points in nearest cluster. The objective is to maximize *Sil(k)*.

$$per\ point\ Sil(x_i) = \frac{b(x_i) - a(x_i)}{\max\{b(x_i), a(x_i)\}} \tag{2}$$

$$per\ cluster\ Sil(C_j) = \frac{1}{|C_j|} \sum_{x_i \in C_j} Sil\left(x_i^{(j)}\right) \tag{3}$$

$$overall\ Sil(k) = \frac{1}{k} \sum_{r=1}^{k} Sil(C_j) \tag{4}$$

2. Davies-Bouldin (DB) index [1]. The objective is to minimize the *DB(k)* index.

$$DB(k) = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left\{ \frac{s_i + s_j}{d_{ij}} \right\} \tag{5}$$

$$s_j = \frac{1}{n_j} \sum_{x_i \in C_j} \left\| x_i^{(j)} - c_j \right\|, \quad d_{ij} = \|c_i - c_j\| \tag{6}$$

3. Calinski-Harabasz (CH) index [11]. The objective is to maximize *CH(k)*.

$$CH(k) = \frac{trace(SSW(k))/_{k-1}}{trace(SSB(k))/_{n-k}} \tag{7}$$

where $SSW(k)$ is the (sum-of-square) within-cluster scatter matrix and, $SSB(k)$ is the (sum-of-square) between-cluster scatter matrix as given below:

$$SSW(k) = \sum_{j=1}^{k} \sum_{x_i \in C_j} \left(x_i^{(j)} - c_j\right)^T \left(x_i^{(j)} - c_j\right) \tag{8}$$

$$SSB(k) = \sum_{j=1}^{k} n_j(c_j - \mu)^T(c_j - \mu) \qquad (9)$$

where μ is the mean of the whole dataset.

4. Krzanowski-Lai (KL) index [14]. The objective is to maximize *KL(k)*.

$$KL(k) = \left| \frac{DIFF(k)}{DIFF(k+1)} \right| \qquad (10)$$

$$DIFF(k) = (k-1)^{2/p} trace(SSW(k-1)) \\ - k^{2/p} trace(SSW(k)) \qquad (11)$$

where p is the dimension (number of variables/features) of the data point; $p = 630$ in our case, see Section 4.

5. Hartigan (Ha) index [3]. The objective is to add cluster until Ha is below a threshold. *Ha≤10* is typically used and we have used this value in our work reported in this paper.

$$Ha(k) = \left(\frac{trace(SSW(k))}{trace(SSW(k+1))} - 1 \right)(n - k - 1) \qquad (12)$$

where *n* is the number of data points in whole data set.

The above indices are global as they consider all clusters in the validation. Even with the value of *k* given, there is no guarantee that K-means as well as any other clustering algorithm will group all observations of the same activity into same cluster. To assess the homogeneity, i.e., cohesion and compactness of individual cluster, we require local internal cluster validity indices. To assess the homogeneity of individual cluster and rank them accordingly, we define two measures: the intra-cluster mean variance ($\bar{\sigma}^2$) and mean joint probability density function ($\bar{P}_f$). Low value of $\bar{\sigma}^2$ indicates compactness of the cluster. The joint probability density function assumes that observations of a non-random activity should be normally distributed within its cluster around the cluster centroid with the standard deviation of the cluster. High value of $\bar{P}_f$ indicates good cohesion of the cluster based on assumption of normal distribution. The equations for the two measures are given in Equations (13) to (18). Logarithm is used in Equation (17) to compress the range of values.

$$mean\ variance\ \bar{\sigma}^2(C_j) = mean(var(C_j)) \qquad (13)$$

$$var(C_j) = \frac{1}{n_j - 1} \sum_{x_i \in C_j} \left(x_i^{(j)} - \bar{x}^{(j)} \right)^{\wedge 2} \qquad (14)$$

$$\bar{x}^{(j)} = mean(x_i \in C_j) \qquad (15)$$

where $x_i^{(j)} = [x_{i1} \cdots x_{ip}]$ is a data point in Cluster C_j with dimension p, n_j is the number of points in Cluster C_j, $\bar{x}^{(j)}$ (dimension p) is the mean of all points in Cluster C_j, ^2 is element-wise square.

$$mean\ joint\ probability\ \bar{P}_f(C_j) = mean(P(C_j)) \qquad (16)$$

$$P(C_j) = \sum_{f=1}^{p} ln\left(pdf\left(x_{if}^{(j)} \right) \right) \qquad (17)$$

$$pdf\left(x_{if}^{(j)} \right) = \frac{1}{\sigma_f^{(j)}\sqrt{2\pi}} e^{-\frac{\left(x_{if}^{(j)} - \mu_f^{(j)}\right)^2}{\left(2\sigma_f^{(j)}\right)^2}} \qquad (18)$$

where $x_{if}^{(j)}$ is the *f*th dimension of $x_i^{(j)}$, $\mu_f^{(j)}$ is the mean of values of *f*th dimension of all points in Cluster C_j, $\sigma_f^{(j)}$ is the standard deviation of values of *f*th dimension of all points in Cluster C_j.

4   Feature extraction

Feature extraction in the context of this paper is not about image processing. The raw data were coordinates of 15 joints in human skeleton provided by the application of Microsoft Kinect, a low-cost RGBD (RGB-Depth) camera, as shown in Fig. 2. Each activity example was sampled for a window of 2 seconds comprising 15 frames. We found empirically [16] that reducing the frames from full 60 frames to 15 frames did not degrade clustering performance. For each frame, the following features were extracted from the coordinates of the joint positions: four vectors describing body flexion, four vectors describing arms abduction, four vectors describing leg abduction and flexion and, two vectors describing interaction between hands and head. The vectors were formed locally (between joints) and normalized to shoulder width making them view invariant to camera and scale invariant to the size of the subject. There were 14 3-dimentional vectors giving 14×3=42 features per frame. With 42 features per frame, the total number of features was 42×15=630 features per activity observation.
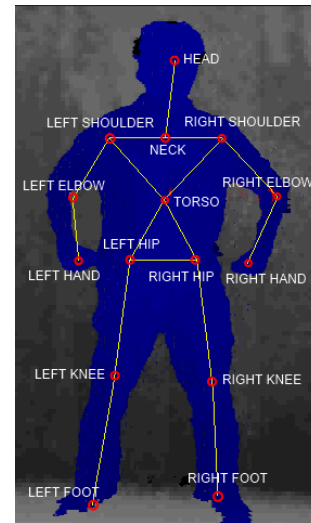
Figure 2. Human skeleton composed from fifteen (15) joint.

5   Data & experiment

5.1   Data

Currently, there are a few [6], [7] publicly available skeleton (coordinates of joints) datasets obtained from Microsoft Kinect sensor on human activities. None of these datasets have been adapted as benchmarking dataset for evaluation of human activity research works. The work presented in this paper used the dataset "Cornell Activity Dataset CAD-60" [6]. CAD-60 consisted of twelve daily activities: rinsing mouth, brushing teeth, wearing contact lens, talking on the phone, drinking water, opening pill container, cooking (chopping), cooking (stirring), talking on couch, relaxing on couch, writing on whiteboard, working on computer. The data was collected from four different subjects: two males (referred as Person 1 and Person 4) and two females (referred as Person 2 and Person 3). One of the females is left-handed (Person 3). Still (standing) and random activity samples by each subject are also included in the dataset. All of the data were collected in a regular household setting with no occlusion of body from the view of sensor. The CAD-60 dataset comprises of RGB images, depth images and skeleton data (coordinates of joint positions and orientations). We have only used the skeleton data of the joint positions in our experiment. We considered eight of them as listed in Table I (1 to 4, 6 to 9). The other three activities are not atomic in their dataset. We refer an atomic activity as one that cannot be further decomposed into sequence of smaller activities. For examples, the drinking comprised of picking up the cup and drink; rinsing mouth comprised of sipping water, gargle and spit; opening pill container comprised of lifting the pill box, twist the cap. At this stage, we are interested to discover atomic actions or lower-level activities, which will be used to discover higher-level activities eventually. We also considered the still (standing) as one activity. In total, we considered nine activities. The random activity samples were also used to test the ability of our approach to reject random activities. Two datasets were composed for each subject as following, giving a total of eight datasets in the experiment: (1) P1, P2, P3, P4: 50 observations of each of the 9 activities for Person 1, 2, 3, 4 respectively. (2) P1R, P2R, P3R, P4R: 100 observations of random activities in addition to 50 observations of each of the 9 activities for Person 1, 2, 3, 4 respectively.

Table I. List of activities

1.	brushing teeth
2.	cooking (chopping)
3.	cooking (stirring)
4.	relaxing on couch
5.	still (standing)
6.	talking on couch (sitting)
7.	talking on the phone
8.	working on computer
9.	writing on whiteboard
10.	random

5.2   Experiment

We conduct the following experiment on each of the eight datasets described in Section 5.

1. Estimate number of clusters, i.e., find k_s.
   1.1. Run K-means from $k=2$ to k_{max}. There is no concrete guideline for the choice of k_{max}, however many researchers had referred to Mardia et al. [8] as stating the rule of thumb for setting $k = \sqrt[2]{n/2}$ where n is the number of data points (observations) in the dataset. We have chosen $k_{max} = \sqrt[2]{n}$ which include the value of k suggested by the said rule of thumb. The actual value of k_{max} is not crucial as we will only accept a few highly ranked clusters. k_{max} can be up to n. However, it helps to restrict the computation time by setting reasonable value of k_{max}.
   1.2. For each value of k, K-means was run for three rounds with random initialization (seeded for comparison with other values of k) and the clustering result with lowest total sum-of-squared-Euclidean distance from all members to their centroid was taken as the result.
   1.3. Compute the global cluster validity indices for all values of k in above.
   1.4. Determine the suggested number of clusters, k_s, based on the global cluster validity test above.
2. Assess clustering quality (homogeneity) based on k_s above. Compute local cluster validity measures for each of the k_s clusters to accept or reject cluster(s).

6   Results & discussion

Table II shows the result of estimating number of clusters (value of k_s) for K-means using the five cluster validity indices for the eight datasets described in Section 5. For the purpose of comparison, we compute the overall error for each index as given in Eq. (19). The index with lowest error is considered giving best value of k_s.

$$Overall\ error\ e_t = e_{t(1)} + e_{t(2)} \qquad (19)$$

where

$$e_{t(1)} = \left(\sum_{P \in \{P1,P2,P3,P4\}} (k_s^P - k_e^P)^2 \right)^{1/2} \qquad (20)$$

$$e_{t(2)} = \left(\sum_{P \in \{P1R,P2R,PR3,P4R\}} (k_s^P - k_e^P)^2 \right)^{1/2} \qquad (21)$$

where k_s^P is the k suggested, for dataset P, k_e^P is the expected k for dataset P, $e_{t(1)}$ is the total error for datasets without random activities, $e_{t(2)}$ is the total error for datasets with random activities.

Table II. Estimated number of clusters, ks, for K-means using DB, CH, KL, Ha and Sil indices for eight datasets.

	DB	CH	KL	Ha	Sil
P1	6	8	8	8	6
P2	6	6	4	14	6
P3	6	10	20	10	9
P4	9	9	20	12	9
$e_{t(1)}$	5.2	3.3	16.4	6.0	4.2
P1R	5	3	17	11	5
P2R	17	3	11	11	4
P3R	10	3	16	12	10
P4R	3	3	17	10	3
$e_{t(2)}$	13.7	20.0	6.7	4.2	15.9
e_t	18.9	23.3	23.1	10.2	20.1

For datasets without random activities, k_e^P is 9. For datasets with random activities, we expect k_e^P to be more than 9, however we do not know the exact number. For the purpose of comparison, we have used the average value of all k_s above 9 for datasets with random activities. The value is 12.9. While Sil, CH and DB did well on datasets without random activities, they performed poorly on datasets with random activities that have high variances. The result suggests that Hartigan index was the best choice among the five indices. The values of k_s given by Hartigan index were used to cluster each dataset and the results are shown in Fig. 3 and 4. Fig. 3 shows the confusion matrices for the clustering result from the datasets without random activities. The rows are the nine activities as listed in Table I (1 to 9). The columns are the clusters. The columns are not merged and not sorted to provide a complete picture of the clustering result. Fig. 4 shows the confusion matrices for the clustering results from the datasets with random activities. Activity 10 is the group of random activities. Table III gives the homogeneity evaluation of the clusters for datasets without random activities, i.e., the results shown in Fig. 3. To explain the interpretation of the table, we take an example of the results for P1. The result shows that Cluster 3 has been ranked top by both evaluation measures. This means Cluster 3 is the most homogeneous according to these measures. Both measures had ranked Cluster 5 at second. Comparing this ranking with the corresponding confusion matrix for P1 in Fig. 3, we see that Cluster 3 and

P1

	1	2	3	4	5	6	7	8
1						48	2	
2			7					43
3			50					
4				50				
5						48	2	
6		50						
7						44	6	
8	50							
9		50						

P2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1					50									
2		1					4	45						
3		37				13								
4					27			23						
5	50													
6				50										
7														50
8		14									36			
9			30		1					8		11		

P3

	1	2	3	4	5	6	7	8	9	10
1				50						
2					1					49
3				50						
4								50		
5	50									
6		50								
7		50								
8							32	18		
9				50						

P4

	1	2	3	4	5	6	7	8	9	10	11	12
1		41		3								6
2			2						48			
3				4	46							
4								50				
5										50		
6						23				26	1	
7				50								
8							50					
9	26			1	23							

Figure 3. Confusion matrix without random activities for four subjects.

Table III. Mean variance ($\bar{\sigma}^2$) and mean joint probability ($\bar{P}_f$) for clustering result in Fig. 3, i.e., without random activities, for each subject

P1	$\bar{\sigma}^2$		$\bar{P}_f$
3	0.0087	3	347.19
5	0.0860	5	101.34
2	0.0883	1	63.40
1	0.129	2	57.23
6	0.223	8	-2.88
8	0.230	6	-29.87
4	0.298	4	-91.74
7	0.410	7	-124.28
P2	$\bar{\sigma}^2$		$\bar{P}_f$
1	0.0570	14	127.46
14	0.0642	1	125.90
5	0.0935	6	34.74
6	0.119	12	31.77
10	0.122	5	20.92
12	0.132	7	3.77
7	0.136	2	-7.01
2	0.145	10	-12.34
3	0.206	3	-33.63
13	0.333	11	-43.07
4	0.336	13	-117.00
11	0.338	4	-123.47
9	0.362	8	-148.66
8	0.452	9	-150.45
P3	$\bar{\sigma}^2$		$\bar{P}_f$
1	0.0122	1	373.36
9	0.0429	9	200.09
5	0.0582	5	143.98
7	0.0702	7	132.11
8	0.144	8	30.58
6	0.186	6	-0.26
3	0.234	4	-17.88
4	0.293	3	-34.13
2	0.320	2	-70.46
10	0.480	10	-172.85
P4	$\bar{\sigma}^2$		$\bar{P}_f$
11	0.0568	3	224.33
8	0.0786	11	162.60
7	0.114	8	103.68
6	0.117	6	72.56
2	0.174	7	30.75
1	0.179	2	16.10
5	0.199	5	-37.77
10	0.249	1	-38.34
9	0.251	9	-43.79
4	0.275	10	-64.10
3	0.358	4	-92.64
12	0.396	12	-160.00

5 concisely captured all 50 observations of Activity 6 (talking on couch) and 4 (relaxing on couch) respectively. In this confusion matrix, Cluster 4, 6 and 7 are weak clusters as they contained more than one activity. Incidentally, these clusters are ranked low in both measures in Table III. Looking at all results in Table III, we can observe that highly ranked clusters are generally homogeneous with a few exceptions. This will not be a problem if the objective is to select one or a few highly ranked clusters to perform learning. We also notice that $\bar{\sigma}^2$ ranked Cluster 3 in P4 low, while $\bar{P}_f$ ranked Cluster 3 on top. Cluster 3 is homogeneous, however, it has only two members; logically it should be rejected.

It will be more interesting to look at the results for datasets containing random activities in Table IV and Fig. 4. It is encouraging to see that clusters containing significant number of random activities have been consistently ranked at the bottom. For P1R, Cluster 3, 9 and 10 contain significant number of random activities and have been ranked at the bottom in corresponding section (P1R) in Table IV. For P2R, Cluster 7, 8, 9 and 11 are ranked at the bottom. Cluster 2 should be rejected as well, and has been ranked low just above those clusters with random activities. For P3R, Cluster 4, 5, 8, and 12 are ranked at the bottom, while Cluster 3 come slightly above them. Ideally, Cluster 3 should be ranked low as it should be rejected. Nevertheless, clusters ranked on top remain correctly identified from homogenous clusters. For P4R, Cluster 3, 5, 6, 7 and 9 are ranked at the bottom. In all datasets, the first few clusters on top are homogenous, apart from Cluster 8 in P4R. However, for Cluster 8 in P4R, having 2 random activities with 50 observations of Activity 3 (stirring) is not expected to cause problem in learning phase.

Referring to Table IV, for each of the homogeneity measure, is there any threshold we can use to identify homogeneous clusters confidently, i.e., to accept the clusters? We find the value for each measure in each dataset where the first cluster from the top is to be rejected. For P1R, the highest cluster to reject is Cluster 2 and the corresponding values for each measure are: $\bar{\sigma}^2$ less than 0.0881 and $\bar{P}_f$ higher than 13.98. For P2R, the highest cluster to reject is Cluster 2 and the corresponding values for each measure are: $\bar{\sigma}^2$ less than 0.268 and $\bar{P}_f$ higher than −112. For P3R, the highest cluster to reject is Cluster 3 and the corresponding values for each measure are: $\bar{\sigma}^2$ less than 0.255 and $\bar{P}_f$ higher than −139.15. For P4R, the highest cluster to reject is Cluster 5 and the corresponding values for each measure are: $\bar{\sigma}^2$ less than 0.458 and $\bar{P}_f$ higher than −183. To obtain a threshold value applicable to all subjects so that non-homogeneous clusters will be rejected, we require that $\bar{\sigma}^2$ to be less than 0.0881 (lowest of all) and $\bar{P}_f$ to be higher than 13.98 (highest of all). Applying these threshold to all datasets in Table IV, we identified the number of top few homogeneous clusters. The result is

shown in Table V. The $\bar{P}_f$ measure has identified more clusters to accept.

P1R	1	2	3	4	5	6	7	8	9	10	11
1	50										
2		50									
3		50									
4										50	
5					50						
6				50							
7	50										
8							27	23			
9	24			26							
10		20				40			12	28	

P2R	1	2	3	4	5	6	7	8	9	10	11
1						50					
2	50										
3	41							9			
4				50							
5							50				
6									50		
7	50										
8		50									
9				45		5					
10		1				1	35	25	13		25

P3R	1	2	3	4	5	6	7	8	9	10	11	12
1											50	
2			50									
3			50									
4						50						
5		50										
6									50			
7					50							
8	32							18				
9	50											
10			24	5	21	3		22	1		1	23

P4R	1	2	3	4	5	6	7	8	9	10
1	3					47				
2		50								
3								50		
4					50					
5			50							
6			1		49					
7	50									
8									50	
9	1			49						
10			38				6	28	2	26

Figure 4. Confusion matrix with random activities for four subjects.

7 Conclusion

In this paper, we proposed an approach to automatically discover new activities without priori. We demonstrated the feasibility of the approach through experimental investigation on daily activity datasets from third party. The results showed that Hartigan index could assist to estimate the possible number of clusters, or k value, in a pool of unlabeled observations of activities for each subject. Given the estimation of k, k clusters were obtained using K-means. The clustering outcomes were assessed using two measures of cluster homogeneity. The results showed both measures consistently ranked highly homogeneous clusters on top while ranking clusters with significant number of random activities at the bottom. It was also observed that

the mean joint probability density function $\bar{P}_f$ measure has the potential to use a threshold value to assist in deciding how many highly homogeneous clusters can be accepted for subsequent learning phase. This is an ability to discover new activities (clusters) autonomously. The model of each of the discovered new activities (clusters) can then be learned using supervised learning algorithm. Armed with this ability, an intelligent system can self-learn and perform unsupervised human activity recognition.

Table IV. Mean variance ($\bar{\sigma}^2$) and mean joint probability ($\bar{P}_f$) for clustering result in Fig. 4, i.e., with random activities, for each subject.

P1R	$\bar{\sigma}^2$		$\bar{P}_f$
4	0.00568	4	393.91
5	0.0374	5	157.31
7	0.0548	11	148.07
1	0.0623	7	142.36
8	0.0733	8	111.28
11	0.0768	1	104.60
2	0.0881	2	13.98
6	0.537	6	-236.60
3	0.599	3	-258.12
9	1.04	9	-274.26
10	1.20	10	-344.00
P2R	$\bar{\sigma}^2$		$\bar{P}_f$
1	0.0341	1	180.34
10	0.0651	10	73.80
5	0.113	5	15.84
6	0.159	6	-46.06
4	0.207	3	-46.25
3	0.240	4	-96.32
2	0.268	2	-112.31
8	0.313	8	-185.44
9	0.939	9	-295.41
11	1.15	11	-342.36
7	1.26	7	-352.73
P3R	$\bar{\sigma}^2$		$\bar{P}_f$
7	0.0249	7	258.57
2	0.0336	2	202.46
1	0.0404	1	190.59
10	0.144	10	24.35
9	0.225	11	-6.07
11	0.252	9	-115.99
3	0.255	6	-133.27
6	0.291	3	-139.15
4	0.308	4	-161.39
12	1.12	12	-317.39
8	1.18	8	-333.79
5	1.35	5	-361.30
P4R	$\bar{\sigma}^2$		$\bar{P}_f$
10	0.0482	10	157.73
4	0.131	1	-0.51
8	0.135	4	-11.41
1	0.151	8	-13.94
2	0.159	2	-23.20
6	0.367	6	-167.71
5	0.458	5	-182.96
3	0.628	3	-267.60
7	1.14	7	-325.54
9	1.32	9	-357.62

Table V. Number of accepted clusters based on threshold value for each measure.

	Number of accepted clusters	
	$\bar{\sigma}^2 < 0.0881$	$\bar{P}_f > 13.98$
P1R	6	7
P2R	2	3
P3R	3	4
P4R	1	1

8   References

[1] D. L. Davies, and D. W. Bouldin, D. W, "A cluster separation measure," in Pattern Analysis and Machine Intelligence, IEEE Transactions on, (2), 224-227, 1979.

[2] D. Wyatt, M. Philipose, and T. Choudhury, "Unsupervised activity recognition using automatically mined common sense," in Proceedings of the National Conference on Artificial Intelligence, July 2005, Vol. 20, No. 1, p. 21.

[3] J. A. Hartigan, "Clustering algorithms," John Wiley & Sons, Inc., 1975.

[4] J. K. Aggarwal, and M.S. Ryoo, "Human activity analysis: A review," in ACM Comput. Surv. 43, 3, Article 16, April 2011.

[5] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, June 1967, Vol. 1, No. 281-297, p. 14.

[6] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Human Activity Detection from RGBD Images," in Association for the Advancement of Artificial Intelligence Workshop on Pattern, Activity and Intent Recognition (PAIR), 2011, pp. 47-55.

[7] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, June 2012, pp. 1290-1297.

[8] K. V. Mardia, J. T. Kent, and J. M. Bibby, "Multivariate analysis," 1980, pp. 365.

[9] M. Stikic, D. Larlus, S. Ebert, and B. Schiele, "Weakly Supervised Recognition of Daily Life Activities with Wearbale Sensors," in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 12, December 2011, pp. 2521-2537.

[10] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," in Journal of computational and applied mathematics, 1987, 20, 53-65.

[11] T. Caliński, and J. Harabasz, "A dendrite method for cluster analysis," in Communications in Statistics-theory and Methods, 1974, 3(1), 1-27.

[12] T. Huynh, M. Fritz, and B. Schiele, "Discovery of Activity Patterns using Topic Models," in UbiComp '08 Proceedings of the 10th International Conference on Ubiquitous Computing, 2088, pp. 10-19.

[13] U. Maulik, and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," inPattern Analysis and Machine Intelligence, IEEE Transactions on, 2002, 24(12), 1650-1654.

[14] W. J. Krzanowski, and Y. T. Lai " A criterion for determining the number of groups in a data set using sum-of-squares clustering," Biometrics, 1988, 23-34.

[15] W. Ong, L. Palafox, and T. Koseki, "Investigation of Feature Extraction for Unsupervised Learning in Human Activity Detection," in Bulletin of Networking, Computing, Systems, and Software, North America, 2, jan. 2013.

[16] W. Ong, T. Koseki, and L. Palafox, "Unsupervised Activity Detection with Skeleton Data From RGB-D Sensor," to be published in Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on IEEE, 2013, in press.

[17] Y. Song, L. Goncalves, and P. Perona, "Unsupervised learning of human motion," in Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2003, 25(7), 814-827.

DGLS System: Decision Guidance for Optimal Load Shedding in Electric Power Microgrids

Chun-Kit Ngan[1] **and Alexander Brodsky**[1]
[1]Department of Computer Science, George Mason University, Fairfax, VA, USA

Abstract - *We propose and develop a decision guidance system, called DGLS, for load shedding of electric power in microgrids to minimize energy costs and maximize customers' savings while preserving the desired quality of service (QoS) in terms of power interruption. The DGLS system is designed to support energy managers to forecast power demand over a time horizon, use the predicted peak demand usage to optimize the peak demand bound for every monthly pay period, continuously monitor the hourly electricity demand, and shed load when the demand exceeds the optimal peak demand bound using a service prioritization scheme. Technically, the contributions of this paper include the design of the DGLS system, the development of a mixed integer linear programming (MILP) model for the peak demand optimization and its implementation using the IBM Optimization Programming Language (OPL), and an experimental case study for a university campus microgrid, which utilizes the peak demand optimization model and a proposed graphical methodology for making a trade-off between cost savings and power interruptions.*

Keywords: Optimization, Decision Guidance, Decision Support, Microgrids, Power Load Shedding, Peak Demand

1 Introduction

Increasing electricity demand has been widely recognized as a global trend in every business and industry. Population growth and economic development are among the key factors that lead to a higher total electricity consumption and a peak demand usage which, in turn, result in a rising energy cost to consumers. In this paper, we focus on the management of peak power demand within microgrids of commercial and industrial customers in order to minimize energy costs and maximize customers' savings while preserving the desired quality of service (QoS) in terms of power interruption.

Typically, it is considerably more expensive to generate electric power for the peak demand. In addition to a higher electricity cost, the peak demand of electric power also results in unpredictable, demand-side power fluctuations, as well as the possible misbalance between the power supply and the customers' demand, which cause power system outages.

Existing approaches to solve the power system outage by electric power companies can be roughly divided into two categories: diverse power-load shedding schemes [1, 2, 3, 4, and 5] and various time-differentiated pricing models [6, 7, 8, 9, 10, 11, 12, 13, 14]. The former approach uses the frequency magnitude, the frequency decline rate, or both of them of the power system to determine when the power load should be shed so that a complete balance between the system supply and the customers' demand can be made. Specifically, if the frequency magnitude drops below a certain threshold, the frequency decline rate reaches a certain limit, or a combination of both, a certain amount of power load is shed in order to rebalance the supply and demand.

The latter approach is to use various time-differentiated pricing models rather than a common average-pricing scheme. The average-pricing scheme charges the customers the average price over a certain period of time of power consumption. However, this pricing scheme does not incentivize the customers to shift power usage to lower demand periods, and thus reduce the total peak demand. To address this issue, various time-differentiated pricing models, such as real-time pricing (RTP) [6, 8, 9, 10, 11, 12, 14], day-ahead pricing (DAP) [7], and time-of-use pricing (TOUP) [13] have been proposed. All of these pricing models reflect the fluctuating prices to the end customers so that they pay what the electricity is worth at different periods of a day. Specifically, these time-differentiated pricing models encourage the customers to shift operations and appliances to the off-peak hours so that their electricity costs can be reduced and the occurrence of the power system outage can be prevented.

However, an important question is how the commercial and industrial customers - for example, at the George Mason University (GMU) campuses, an unusually high peak demand usage of just a few minutes may significantly increases the cost of the electric bill for the following year - should respond to those time-differentiated pricing approaches. Answering this question is exactly the focus of this paper. To mitigate the peak demand problem, our key idea is to learn an optimal peak demand bound over historical and projected electric power demands for each future pay period. This optimal peak demand bound is then used to monitor the prospective demand usage in any time interval of that future pay period.

Once the demand usage exceeds the bound, some electricity loads are shed by shutting down some electric account units so that the peak demand charge can be controlled.

To address this problem, in this paper, we propose and develop a decision guidance system [15, 16, 17, 18, 19, 20], called DGLS, for load shedding of electric power in microgrids in order to minimize energy costs and maximize customers' savings while preserving the desired quality of service (QoS) in terms of power interruption. More specifically, the DGLS system is designed to support energy managers to forecast electric power demand over a time horizon, use the predicted peak demand usage to optimize the peak demand bound for every monthly pay period, continuously monitor the hourly electricity demand, and shed load when the demand exceeds the optimal peak demand bound using a service prioritization scheme. More specifically, the technical contributions of this paper include (1) the design of the DGLS system, (2) the development of a mixed integer linear programming (MILP) model for the peak demand optimization, which is both accurate and efficient, and its implementation using the IBM Optimization Programming Language (OPL) [21, 22], and (3) an experimental case study for the GMU university campus microgrid, which utilizes the peak demand optimization model and a proposed graphical methodology for making a trade-off between cost savings and power interruptions.

The rest of the paper is organized as follows. In Section 2, we provide a descriptive overview on the DGLS system. Using the GMU energy cost problem as an example, we describe and demonstrate the DGLS optimization model and its OPL implementation in Section 3 and 4 respectively. In Section 5, we show the trade-off graph between the annual power saving and the annual power interruption, as well as explain the graph in detail on the GMU energy cost problem. In Section 6, we conclude and briefly outline future work.

2 Decision-Guided Load-Shedding (DGLS) System

To better understand the load shedding problem, we consider the real case study at the George Mason University (GMU) campuses, where there are more than 33,000 students, and the total size of all the campuses is more than 800 acres, in which the electric power demand across those expanding campuses is expected to increase. The increase in electric power consumption results in a higher electricity cost, which is composed of the two main components: (1) a total kilowatt-hour (kWh) charge, i.e., the charge for the total electricity consumption, and (2) an Electricity Supply (ES) service charge, i.e., the charge for the peak demand usage in any 30-minute interval over the past 12 months. The first total kWh charge is priced particularly high during the business office hours between 09:00 a.m. and 06:00 p.m. from Monday to Friday. The second ES service charge is a proxy for the cost of capital investment for power generation capacity, since the power company, such as Virginia Electric and Power

Company, needs to build generation, transmission, and distribution facilities that are capable of supporting the peak demand usage, even though the average power demand could be considerably lower. This ES service charge, i.e., the peak demand charge, amounts to approximately 30% of the electric bill in each monthly pay period and is determined based upon the electricity supply demand. This electricity supply demand is decided on the highest of either (*C1*) or (*C2*) according to the electric utility contract:

C1: The highest average kilowatt measured in any 30-minute interval of the current billing month during the on-peak hours of either:

(1) Between 10 a.m. and 10 p.m. from Monday to Friday for the billing months of June through September, or

(2) Between 7 a.m. and 10 p.m. from Monday to Friday for all other billing months.

C2: 90% of the highest kilowatt of demand at the same location as determined under (*C1*) above during the billing months of June through September of the preceding eleven billing months.

Thus it is possible that a high peak demand usage just for one minute of electricity consumption over the past year would result in a very significant increase in the total charge of the electric bill of the next monthly pay period. Therefore, controlling the peak demand usage is crucial for decreasing the electricity cost.

To address this peak demand problem, we propose the Decision-Guided Load-Shedding (DGLS) System, shown in Fig. 1. This system has four main components: Energy Management System (EMS), Load Shedding Priority Controller (LSPC), DGLS Optimizer, and Demand Prediction Learner (DPL). The EMS is a system of computer-aided tools to monitor, control, and optimize the performance of the microgrid. One operation of the EMS is to monitor and receive the meter/sensor data from the electric power microgrid and then sends those historical power-demand data to the DPL, which uses the received data and the Energy Manager's Facility Expansion Plan, e.g., the planned size of the future area increased at the GMU Fairfax campus in the next academic year, to generate the predicted electric power demand over a time horizon, e.g., two years. Another operation of the EMS is to issue the control command to the microgrid to shut down some electric account units to prevent the electricity demand from exceeding the peak demand bound. That control command generated and initiated by the EMS to the microgrid is based on the load-shedding command inputted from the LSPC, which receives the input data and information, i.e., the load shedding prioritization scheme from the Energy Manager and the optimal peak demand bound from the DGLS optimizer, to generate the control command. The DGLS optimizer utilizes the input information, including the QoS requirements, e.g., the annual

maximal power interruption in kWh, and the electric utility contractual terms, e.g., the ES service charge deteration (*C1* and *C2*), from the Energy Manager, as well as the predicted electricity demand over a time horizon, e.g., two years from 2012 to 2013, from the DPL, to generate a decision optimization model described in Section 3. This model is used to learn the optimal peak demand bound, which is described in Section 4, as an input to the LSPC. Note that the DGLS optimizer learns the peak demand bound monthly in advance for the next monthly pay period, and the EMS monitors the real-time demand on an hourly basis for preventing the demand from exceeding the learned, optimal peak demand bound.
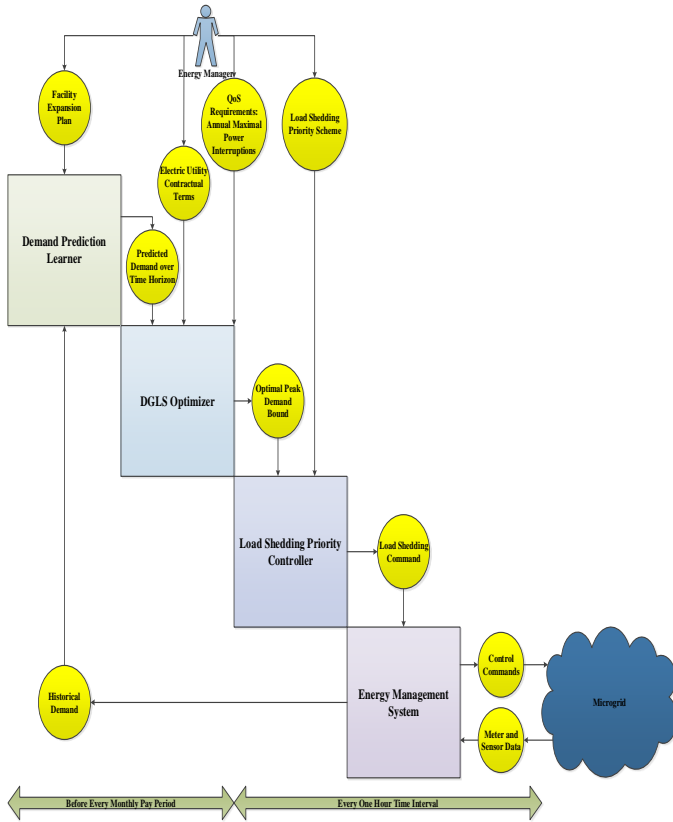
Fig 1. Decision-Guided Load-Shedding (DGLS) System.

3   The Decision-Guided Load-Shedding (DGLS) Optimization Model

To formulate a DGLS optimization model to learn an optimal peak demand bound for every monthly pay period requires three input data sets: the historical and predicted electricity demand over a time horizon, the electric utility contractual terms of each pay period payment, and the maximal power interruptions allowed per year. Using the GMU energy cost problem as an example, we explain the terminology and the optimization problem formulation used in this case study, which are shown in Table I, Table II, and Fig. 2 respectively. In addition to those, Table III includes the descriptions for all the constant values, from the electric

utility contract, used in the model for the GMU energy cost problem.

TABLE I.   INPUT DATA SET FROM THE DEMAND PREDICTION LEARNER

Input Data Item	Description
Power Intervals	• PowerIntervals is a set of powerIntervals, where powerInterval = {pInterval, payPeriod, year, month, day, hour, weekDay} is a tuple, for $-8759 \leq$ pInterval ≤ 17520, $-11 \leq$ payPeriod ≤ 24, $2011 \leq$ year ≤ 2013, $1 \leq$ month ≤ 12, $1 \leq$ day ≤ 31, $0 \leq$ hour ≤ 23, and $0 \leq$ weekDay ≤ 6. • pInterval is a hourly time interval denoted by an integer value, which indicates a historical data if pInterval is less than or equal to zero, e.g., $-8759 \leq$ pInterval ≤ 0 for the year 2011, and a future forecasted data if pInterval is greater than zero, e.g., $1 \leq$ pInterval ≤ 17520 for the year 2012 and 2013. • payPeriod is a monthly pay period denoted by an integer value, which indicates a historical data if payPeriod is less than or equal to zero, e.g., $-11 \leq$ payPeriod ≤ 0 for the year 2011, and a future forecasted data if payPeriod is greater than zero, e.g., $1 \leq$ payPeriod ≤ 24 for the year 2012 and 2013. Each payPeriod corresponds to a range of pIntervals; for example, the payPeriod = 1 consists of the pIntervals from 1 to 730.
Power Demand Array	demandKw[PowerIntervals] ≥ 0 is an array of electric power demand over the PowerIntervals.

TABLE II.   INPUT DATA SET FROM THE ENERGY MANAGER

Input Data Item	Description
Annual Maximal Power Demand Interruption	annualBound $\in Z^*$ is the annual maximal power demand interruption.

Actual Power Consumption Per Power Interval	kW[i] is the actual power consumption for each power interval that satisfies the below constraints:
	
	if (i.pInterval ≤ 0 ∨ demandKw[i] $\leq$ peakDemandBound[i.payPeriod])
	    kW[i] = demandKw[i]
	
	else if (demandKw[i] > peakDemandBound[i.payPeriod])
	    kW[i] = peakDemandBound[i.payPeriod];,
	
	where i ∈ PowerIntervals, $-8759 \leq$ i.pInterval ≤ 17520, $-11 \leq$ i.payPeriod ≤ 24, and peakDemandBound[i.payPeriod] is the decision parameter for every pay period.
	
	It means that if the time interval is historical or the actual demand demandKw[i] is less than or equal to the peak demand bound, the actual power consumption kW[i] has to be bound by demandKw[i]. However, if the time interval is positive and the actual demand demandKw[i] is greater than the peak demand bound, kW[i] has to be bound by the peak demand bound instead.

Peak Demand Usage Per Pay Period	payPeriodSupplyDemand[p] is the peak demand usage per pay period, which is defined in the below constraints according to the electric utility

Input Data Item	Description

	contract, i.e., *C1* and *C2*:
	
	 if (i.payPeriod == p ∧ i.weekDay ≥ 1 ∧ i.weekDay ≤ 5 ∧ ((i.month ≥ 6 ∧ i.month ≤ 9 ∧ i.hour ≥ 10 ∧ i.hour ≤ 22) ∨ (i.month ≤ 5 ∧ i.month ≥ 10 ∧ i.hour ≥ 7 ∧ i.hour ≤ 22)))
	     payPeriodSupplyDemand[p] ≥ kW[i]
	
	 else if (i.month ≥ 6 ∧ i.month ≤ 9 ∧ i.payPeriod ≥ p – 11 ∧ i.weekPay ≥ 1 ∧ i.weekDay ≤ 5 ∧ i.hour ≥ 10 ∧ i.hour ≤ 22)
	     payPeriodSupplyDemand[p] ≥ 0.9 * kW[i];,
	
	 where i ∈ PowerIntervals, p ∈ PayPeriods, and 1 ≤ p ≤ 24.

Generation Demand Charge	generationDemandCharge[p] = 8.124 * payPeriodSupplyDemand[p] is the ES service charge, i.e., the peak demand charge, where p ∈ PayPeriods, and 1 ≤ p ≤ 24.
Aggregated Annual Maximal Power Interruption over the future time intervals from 2012 to 2013	∑(demandKw[i] – kW[i]) ≤ 2 * annualBound is the aggregated annual maximal power interruptions for two years, where i ∈ PowerIntervals and 1 ≤ i.pInterval ≤ 17520.
Peak Demand Bound	peakDemandBound[p] cannot exceed payPeriodSupplyDemand[p], i.e., peakDemandBound[p] ≤ payPeriodSupplyDemand[p], where p ∈ PayPeriods, and 1 ≤ p ≤ 24.
Total Power Consumption Per Pay Period	payPeriodKwh[p] is the total power consumption in each pay period, i.e., payPeriodKwh[p] = ∑kW[i] * timeIntervalSize, where i ∈ PowerIntervals, p ∈ PayPeriods, 1 ≤ p ≤ 24, i.payPeriod = p, and timeIntervalSize = 1.

Total kWh Charge Per Pay Period	payPeriodKwhCharge[p] is total kWh charge per pay period, i.e., payPeriodKwhCharge[p] ≥ 0, which satisfies the below constraints according to the electric utility contract:
	
	 if (payPeriodKwh[p] ≤ 24000)
	     payPeriodKwhCharge[p] = 0.01174 * payPeriodKwh[p]
	
	 else if (payPeriodKwh[p] ≤ 210000 + extraKwhBound[p])
	     payPeriodKwhCharge[p] = 0.01174 * 24000 + 0.00606 * (payPeriodKwh[p] – 24000)
	
	 else
	     payPeriodKwhCharge[p] = 0.01174 * 24000 + 0.00606 * (186000 + extraKwhBound[p]) + 0.00244 * (payPeriodKwh[p] – (210000 + extraKwhBound[p])),
	
	 where extraKwhBound[p] = 210 * (payPeriodSupplyDemand[p] – 1000), p ∈ PayPeriods, and 1 ≤ p ≤ 24.

Total Cost Per Pay Period	The total cost per pay period is the sum of payPeriodKwhCharge[p] and generationDemandCharge[p], i.e., (payPeriodKwhCharge[p] + generationDemandCharge[p]), where p ∈ PayPeriods, and 1 ≤ p ≤ 24.

Input Data Item	Description
Total Cost of all the PayPeriods	The total cost of all the PayPeriods is the aggregations of all the total costs in each pay period, i.e., totalCost = ∑(payPeriodKwhCharge[p] + generationDemandCharge[p]), where p ∈ PayPeriods, and 1 ≤ p ≤ 24.

TABLE III.   INPUT DATA SET FROM THE ENERGY MANAGER

Constant	Description
0.9	Percentage of the highest kW of demand during the billing months of June through September of the preceding 11 billing months.
8.124	Amount ($) of Electricity Supply (ES) demand charged per kW.
24000	First ES kWh
0.01174	Amount ($) of the first 24000 ES kWh charged per kWh
186000	Next ES kWh
0.00606	Amount ($) of the next 186000 ES kWh charged per kWh
210000	Sum of the first ES kWh and the next ES kWh
0.00244	Amount ($) of the additional ES kWh charged per kWh
1000	kW of ES demand
210	kWh for each ES kW of demand over 1000 kW

Minimize totalCost

Subject To

∀i ∈ PowerIntervals,

$$kW[i] = \begin{cases} demandKw[i] \ if \ i.pInterval \leq 0 \ \lor \ demandKw[i] \leq peakDemandBound[i.payPeriod] \\ peakDemandBound[i.payPeriod] \ if \ i.pInterval \geq 1 \ \land \\ \qquad demandKw[i] > peakDemandBound[i.payPeriod] \end{cases}$$

where i.pInterval = −8759, −8758, …, 0, 1, …, 17520, *and i.payPeriod* = −11, −10, …, 0, 1, …, 24

peakDemandBound[p] ≤ payPeriodSupplyDemand[p], where p = 1, 2, …, 24

∀i ∈ PowerIntervals,

$$payPeriodSupplyDemand[p] \geq \begin{cases} kW[i] \ if \ (i.payPeriod \ = \ p \land i.weekDay \geq 1 \land i.weekDay \leq 5 \land \\ \qquad ((i.month \geq 6 \land i.month \leq 9 \land i.hour \geq 10 \land i.hour \leq 22) \lor \\ \qquad (i.month \leq 5 \land i.month \geq 10 \land i.hour \geq 7 \land i.hour \leq 22))) \\ 0.9 * kW[i] \ if \ (i.month \geq 6 \land i.month \leq 9 \land i.payPeriod \geq p-11 \land \\ \qquad i.weekDay \geq 1 \land i.weekDay \leq 5 \land i.hour \geq 10 \land i.hour \leq 22) \end{cases}$$

where p = 1, …, 24

$$payPeriodKwhCharge[p] = \begin{cases} 0.01174 * payPeriodKwh[p] \\ \qquad if \ (payPeriodKwh[p] \leq 24000) \\ 0.01174 * 24000 \\ \qquad +0.00606 * (payPeriodKwh[p] - 24000) \\ \qquad if (payPeriodKwh[p] \geq 24000 \\ \qquad \land \ payPeriodKwh[p] \leq 210000 + extraKwhBound[p]) \\ 0.01174 * 24000 \\ \qquad + 0.00606 * (186000 + extraKwhBound[p]) + \\ 0.00244 * (payPeriodKwh[p] - (210000 + extraKwhBound[p])) \\ \qquad if (payPeriodKwh[p] > 210000 + extraKwhBound[p]) \end{cases}$$

where p = 1, …, 24

$$\sum_{\substack{i \in PowerIntervals \land \\ i.pInterval \geq 1 \land i.pInterval \leq 17520}} (demandKw[i] - kW[i]) \leq 2 * annualBound$$

peakDemandBound[p] ≥ 0,
payPeriodSupplyDemand[p] ≥ 0,
payPeriodKwh[p] ≥ 0,
extraKwhBound[p] ≥ 0,
payPeriodKwhCharge[p] ≥ 0, where p = 1, 2, …, 24,
kW[i] ≥ 0, where i ∈ PowerIntervals

Fig 2. The DGLS Optimization Model for the GMU Energy Cost Problem.

From the MILP model shown in Fig. 2, we can see that *demandKw[i]* is assigned to *kW[i]* when the power interval is

in the past or *demandKw[i]* is less than or equal to *peakDemandBound[p]*, where i.pInterval = p. However, when the power interval is in the future and *demandKw[i]* is greater than *peakDemandBound[p]*, *peakDemandBound[p]* is assigned to *kW[i]*. *peakDemandBound[p]* is also bound by *payPeriodSupplyDemand[p]*, which is restricted by the electric contractual constraints, i.e., *C1* and *C2*. *payPeriodKwhCharge[p]* is determined based upon the electric utility contract of the electric bill, which is described in Table II. The total allowed power interruption over the future power intervals cannot be more than the twice of *annualBound*, where *annualBound* is the annual maximal power demand interruption over the future time intervals from 2012 and 2013. Lastly, all the decision control variables, including *peakDemandBound[p]*, *payPeriodSupplyDemand[p]*, *payPeriodKwh[p]*, *extraKwhBound[p]*, *payPeriodKwhCharge[p]*, and *kW[i]* must be restricted in sign, i.e., non-negative real values, where $1 \leq p \leq 24$ and $i \in$ PowerIntervals.

4   The OPL Implementation for the DGLS Optimization Model

The DGLS optimization model has been implemented by using the IBM OPL language. Using the GMU historical data of power usage in the past years, e.g., 2011, and its predicted electricity demand over a future time horizon, e.g., 2012 and 2013, we used the OPL language to implement and demonstrate the DGLS optimization model for the GMU energy cost problem, which is shown in Fig. 3, as an example. First, the value 24, i.e., the total 24 months from 2012 to 2013, is assigned to the variable *nbPayPeriods* in the line number 7. The value starting from 0 to 1500000000 kWh is assigned to the variable *annualBound*, that is, the annual maximal power interruption allowed in the line number 8, one by one to demonstrate the tradeoff in Section V. From the line number 11 to 19, I declare a tuple of a power interval that has the attributes, including *pInterval*, *payPeriod*, *year*, *month*, *day*, *hour*, and *weekDay*. The line number 21 declares the *PowerIntervals* that include both the past and the future power intervals. The line number 22 declares the *demandKw[PowerIntervals]* array. The line number 24, 25, 26, and 27 declare the decision parameter sets, including the *peakDemandBound[p]*, *kW[PowerIntervals]*, *payPeriodSupplyDemand[p]*, and *payPeriodKwhCharge[p]*. The line number 31, 32, 33, and 34 declare decision parameter expressions, i.e., *payPeriodKwh[p]*, *generationDemandChange[p]*, and *extraKwhBound[p]*. The total cost, which is declared on the line number 34, is minimized on the line number 36. All the constraints are declared from the line number 39 to 65.

The intuition of using the OPL language is that its optimization formulation looks like the formal mathematical model. When comparing the mathematical optimization model in Fig. 2 with the OPL formulation in Fig. 3, we realize that both models are very similar to each other. Only some notations and syntaxes are different that is shown in Table IV.

For example, instead of using the summation sign ($\sum$) in the MILP model, the OPL language uses the syntax, "sum", to perform the aggregation. Rather than using the if-then statement in the mathematics, the OPL uses the construct with the implication operation (=>).

```
1  /**********************************************
2   * OPL 12.4 Power Demand Model                *
3   * Author: Alexander Brodsky and Chun-Kit Ngan*
4   * Creation Date: May 30, 2012 at 8:28:56 PM  *
5   **********************************************/
6  float timeIntervalSize = ...;
7  int nbPayPeriods = ...;
8  float annualBound = ...;
9  range PayPeriods = 1..nbPayPeriods;
10
11 tuple powerInterval{
12   int pInterval;
13   int payPeriod;
14   int year;
15   int month;
16   int day;
17   int hour;
18   int weekDay;
19 }
20
21 {powerInterval} PowerIntervals = ...;
22 float demandKW[PowerIntervals] = ...;
23
24 dvar float+ peakDemandBound[PayPeriods];
25 dvar float+ kW[PowerIntervals];
26 dvar float+ payPeriodSupplyDemand[PayPeriods];
27 dvar float+ payPeriodKwhCharge[PayPeriods];
28
29 pwlFunction kWfunction[i in PowerIntervals] = piecewise{1 -> demandKw[i]; 0};
30
31 dexpr float payPeriodKwh[p in PayPeriods] = sum(i in PowerIntervals : i.payPeriod == p) kW[i] * timeIntervalSize;
32 dexpr float generationDemandCharge[p in PayPeriods] = 8.124 * payPeriodSupplyDemand[p];
33 dexpr float extraKwhBound[p in PayPeriods] = 210 * (payPeriodSupplyDemand[p]-1000);
34 dexpr float totalCost = sum(p in PayPeriods) (payPeriodKwhCharge[p] + generationDemandCharge[p]);
35
36 minimize totalCost;
37
38 subject to {
39   forall(i in PowerIntervals : i.pInterval <= 0) kW[i] == demandKw[i];
40
41   forall(i in PowerIntervals : i.pInterval >= 1) kW[i] == kWfunction[i](peakDemandBound[i.payPeriod]) ;
42
43   forall (p in PayPeriods) peakDemandBound[p] <= payPeriodSupplyDemand[p];
44
45   forall(p in PayPeriods)
46     forall(i in PowerIntervals : i.payPeriod == p && i.weekDay >= 1 && i.weekDay <= 5
47       && ((i.month >= 6 && i.month <= 9 && i.hour >= 10 && i.hour <= 22) ||
48       (i.month <= 5 && i.month >= 10 && i.hour >= 7 && i.hour <= 22)))
49         payPeriodSupplyDemand[p] >= kW[i];
50
51   forall(p in PayPeriods)
52     forall(i in PowerIntervals : i.month >= 6 && i.month <= 9 && i.payPeriod >= p - 11
53       && i.weekDay >= 1 && i.weekDay <= 5 && i.hour >= 10 && i.hour <= 22)
54         payPeriodSupplyDemand[p] >= 0.9 * kW[i];
55
56   forall (p in PayPeriods)
57     ((payPeriodKwh[p] <= 24000)
58       => payPeriodKwhCharge[p] == 0.01174 * payPeriodKwh[p]) &&
59     ((payPeriodKwh[p] >= 24000 && payPeriodKwh[p] <= 210000 + extraKwhBound[p])
60       => payPeriodKwhCharge[p] == 0.01174 * 24000 + 0.00606 * (payPeriodKwh[p] - 24000)) &&
61     ((payPeriodKwh[p] >= 210000 + extraKwhBound[p])
62       => payPeriodKwhCharge[p] == 0.01174 * 24000 + 0.00606 * (186000 + extraKwhBound[p])
63                   + 0.00244 * (payPeriodKwh[p] - (24000 + 186000 + extraKwhBound[p])));
64
65   sum(i in PowerIntervals : i.pInterval >= 1) (demandKw[i] - kW[i]) <= annualBound * 2;
66 }
```

Fig 3. The OPL Implementation for the GMU Energy Cost Problem.

TABLE IV.   DIFFERENCES BETWEEN MATHEMATICAL OPTIMIZATION MODEL AND OPL FORMULATION MODEL.

Mathematical Optimization Model	OPL Formulation Model
Notation: Summation Sign ($\sum$)	Syntax: sum
Example: $\sum$(demandKw[i] − kW[i]) ≤ 2 * annualBound	Example: **sum**(i in PowerIntervals : i.pInterval >= 1) (demandKw[i] - kW[i]) <= annualBound * 2
Notation: If-then Statement	Syntax: =>
Example: **if** (payPeriodKwh[p] $\leq$ 24000)	Example: (payPeriodKwh[p] <= 24000)

Mathematical Optimization Model	OPL Formulation Model
payPeriodKwhCharge[p] = 0.01174 * payPeriodKwh[p]	=> (payPeriodKwhCharge[p] == 0.01174 * payPeriodKwh[p])
Notation: Where clause Example: peakDemandBound[p] ≤ payPeriodSupplyDemand[p], **where** p ∈ PayPeriods	Syntax: forall Example: **forall** (p in PayPeriods) peakDemandBound[p] <= payPeriodSupplyDemand[p];

5   Trade-Off Graph for the Annual Power Interruption (kWh) vs. the Projected Annual Saving (USD) for the GMU Energy Cost Problem

Using the OPL language to formulate and solve the GMU energy cost problem as an example, we plotted the trade-off graph between the annual power interruption (kWh) and the projected annual saving (USD) that is shown in Fig. 4. From the graph, we see that there are three clear regions. Between 0 and 10000 kWh, the curve of the graph is increasing very slowly. It means that an annual power interruption in this region does not bring a significant amount of annual saving to the university although the power supply service to the university is kind of minimal. When the power interruption starts after the 10000 kWh, the curve of the graph starts getting steep and keeps increasing that leads to a higher amount of annual saving for the university, but the power interruption begins bringing a significant impact on the GMU campus. In the last portion of the graph, we find that even though the power interruption reaches 95000000 kWh or more, the annual saving still remains the same, i.e., $1551246.81. It means that increasing the power interruption infinitely does not result in an increasing annual saving limitlessly. Such infinite power interruption only worsens the power supply service to the entire campuses at GMU. Using this given trade-off graph, the GMU Energy Manager can determine which level of desired quality of service (QoS) in terms of power interruption on the campus can be preserved while earning a desirable annual saving.

6   Conclusions and Future Work

In this paper we propose and report on the development of DGLS, a Decision-Guidance System for Load Shedding of electric power in microgrids in order to minimize energy costs and maximize customers' savings while preserving the desired quality of service (QoS) in terms of power interruption. The DGLS system is designed to support energy managers to forecast electric power demand over a time horizon, use the predicted peak demand usage to optimize the peak demand bound for every monthly pay period, continuously monitor the hourly electricity demand, and shed load when the demand exceeds the optimal peak demand bound using a service prioritization scheme. The core technical challenge is the development of the mixed integer

linear programming model (MILP) for the peak demand optimization that is very accurate in terms of the electric contractual terms and engineering constraints, and yet efficient and scalable, which is done by the careful modeling of mainly continuous decision variables and using constructs that avoid introduction of combinatorics, e.g., explicit or implicit binary variables, into the model. The model has also been implemented and demonstrated by the IBM OPL language for the GMU energy cost problem. The future work includes the specific schemes for service prioritization, tighter integration with the energy management system, and introducing additional factors such as power curtailment contracts and local back-up generation of power.
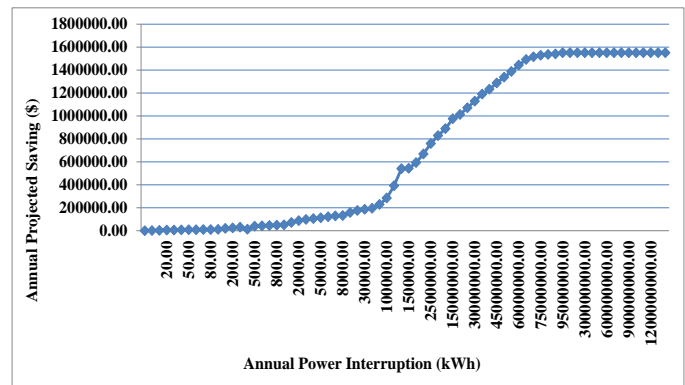
Fig 4. Trade-off Graph between the Annual Power Interruption and the Projected Annual Saving at George Mason University.

7   References

[1]   Ahsan, M.Q., Chowdhury, A.H., Ahmed, S.S., Bhuyan, I.H., Haque, M.A., & Rahman, H. (2012). Technique to Develop Auto Load Shedding and Islanding Scheme to Prevent Power System Blackout. IEEE Transactions on Power Systems, Volume. 27, Issue. 1, Pages. 198 – 205.

[2]   Mahat, P., Chen, Z., & Bak-Jensen, B. (2010). Underfrequency Load Shedding for an Islanded Distribution System with Distributed Generators. IEEE Transactions on Power Delivery, Volume. 25, Issue. 2, Pages. 911 – 918.

[3]   Ding, Z., Cartes, D.A., & Srivastava, S. (2006). New Load Shedding Scheme for Islanded Power Systems. Proceedings of the 2006 IEEE/SMC International Conference on System of Systems Engineering. Tallahassee, FL.

[4]   You, H., Vittal, V., & Yang, Z. (2003). Self-Healing in Power Systems: An Approach Using Islanding and Rate of Frequency Decline-Based Load Shedding. IEEE Transactions on Power Systems, Volume. 18, Issue. 1, Pages. 174 – 181.

[5]   Delfino, B., Massucco, S., Morini, A., Scalera, P., & Silvestro, F. (2001). Implementation and Comparison of Different under Frequency Load-shedding Schemes.

Proceedings of Power Engineering Society Summer Meeting, Volume. 1, Pages 307 – 312.

[6] Ameren Illinois Power Company. (2010). Real-time Pricing for Residential Customers. http://www.ameren.com/Residential/ADC.

[7] Centolella, P. (2010). The Integration of Price Responsive Demand into Regional Transmission Organization (RTO) Wholesale Power Markets and System Operations. Energy, Volume. 35, Issue. 4, Pages. 1568 - 1574.

[8] Mohsenian-Rad, A-H. & Leon-Garcia, A. (2010). Optimal Residential Load Control with Price Prediction in Real-Time Electricity Pricing Environments. IEEE Transactions on Smart Grid, Volume. 1, Issue. 2, Pages. 120 - 133.

[9] Burke, W. & Auslander, D. (2009). Residential Electricity Auction with Uniform Pricing and Cost Constraints. Proceedings of North American Power Symposium. Starkville, MS.

[10] Ontario Newsroom for Residents. (2009). Smarter Electricity Pricing Coming to Ontario: McGuinty Government Rolls Out Time-of-Use Rates. http://news.ontario.ca/mei/en/2009/05/smarter-electricity-pricing-coming-to-ontario.html.

[11] Holland, S., & Mansur, E. (2008). Is Real-time Pricing Green? The Envirobmental Impacts of Electricity Demand Variance. The Review of Economics and Statistics, Volume. 90, No. 3, Pages. 550 – 561.

[12] Alexander, B. (2007). Smart Meters, Real Time Pricing, and Demand Response Programs: Implications for Low Income Electric Customers. Oak Ridge National Laboratory Technical Report.

[13] Wolak, F. (2006). Residential Customer Response to Real-time Pricing: The Anaheim Critical Peak Pricing Experiment. Center for the Study of Energy Markets, University of California Energy Institute, UC Berkeley.

[14] Borenstein, S. (2004). The long-run Effects of Real-time Electricity Pricing. Center for the Study of Energy Markets, University of California Energy Institute, UC Berkeley.

[15] Alrazgan, A., Nagarajan, A., Brodsky, A., and Egge, N., (2011). Learning Occupancy Prediction Models with Decision-Guidance Query Language. Proceedings of the 44th Hawaii International Conference on System Sciences. Koloa, Kauai, Hawaii, U.S.A.

[16] Brodsky, A., Cherukullapurath, M., Awad, M., and Egge, N., (2011). A Decision-Guided Advisor to Maximize ROI in Local Generation and Utility Contracts. Proceedings of the

2nd European Conference and Exhibition on Innovative Smart Grid Technologies. Manchester, U.K.

[17] Brodsky, A., Egge, N., and Wang, X.S., (2011). Reusing Relational Queries for Intuitive Decision Optimization. Proceedings of the 44th Hawaii International Conference on System Sciences. Koloa, Kauai, Hawaii, U.S.A.

[18] Brodsky, A., Bhot, M.M., Chandrashekar, M., Egge, N.E., and Wang, X.S., (2009). A Decisions Query Language (DQL): High-Level Abstraction for Mathematical Programming over Databases. Proceedings of the 35th SIGMOD International Conference on Management of Data. Providence, RI, U.S.A.

[19] Brodsky, A., Henshaw, S.M., and Whittle, J., (2008). CARD: A Decision-Guidance Framework and Application for Recommending Composite Alternatives. Proceedings of the 2nd ACM International Conference on Recommender Systems. Lausanne, Switzerland.

[20] Brodsky, A. and Wang. X.S., (2008). Decision-Guidance Management Systems (DGMS): Seamless Integration of Data Acquisition, Learning, Prediction, and Optimization. Proceedings of the 41st Hawaii International Conference on System Sciences. Waikoloa, Big Island, Hawaii, U.S.A.

[21] The IBM Corporation. (2012). Optimization Programming Language (OPL). http://pic.dhe.ibm.com/infocenter/cosinfoc/v12r4/index.jsp?topic=%2Filog.odms.ide.help%2FOPL_Studio%2Fmaps%2Fgroupings_Eclipse_and_Xplatform%2Fps_opl_Language_1.html.

[22] Hentenryck, P.V. (1999). The OPL Optimization Programming Language. The MIT Press.

A MEMs-based Labeling Approach to Punctuation Correction in Chinese Opinionated Text

Yanqing Zhao and Guohong Fu

School of Computer Science and Technology, Heilongjiang University, Harbin 150080, China

Abstract - *This paper presents a maximum entropy models based approach to punctuation prediction and correction for Chinese opinionated texts. This study involves three parts. First, we conduct a survey of punctuation errors in Chinese opinionated texts based on a corpus of online product reviews. Then, we propose a maximum entropy sequence labeling approach to Chinese punctuation prediction. Finally, we perform punctuation error detection and correction by comparing automatically-predicted punctuations with the relevant original punctuations in opinionated texts. Our experimental results show our system is effective for most punctuation errors in Chinese opinionated texts.*

Keywords: Opinion mining, punctuation correction, punctuation prediction, maximum entropy models

1   Introduction

With the explosive growth of various opinions posted on the web over the past years, opinion mining has been attracting an ever-increasing amount of attention from the natural language processing community. However, as one typical type of user-generated contents, opinionated texts are usually informal and ill-formed with a variety of noises. In general, opinionated text contains spelling errors, special characters, non-standard word forms, grammar mistakes, misplaced punctuations, usage of multilingual words and so on [1] [2] [3]. Such noise inevitably raises a new challenge for processing opinionated texts. This paper will focus on punctuation correction in Chinese opinionated text.

As important symbols in written language, punctuation marks indicate not only the structure and organization of written language but also the intonation and pauses for reading. Therefore, adding proper punctuation marks can not only enhance the readability and understanding of texts. For Chinese, punctuation marks can also provide additional information for further language analysis of texts, such as word segmentation, phrasing and syntactic analysis [5][6][7]. On the other hand, punctuation is casually placed in informal user-generated text, which inevitably introduces numerous mistaken punctuation marks in opinionated text and thus bring a new obstacle to opinionated text analysis. At this point, punctuation prediction and correction plays an important role in many natural language processing tasks such as opinion mining, micro-blog processing, automatic summarization, information extraction, automatic speech recognition (ASR), and machine translation (particularly spoken language translation) [8] [9].

Punctuation prediction, also referred to as punctuation restoration, aims at inserting proper punctuation marks at the proper position of raw character or word streams without punctuation [5][10], while punctuation correction is to detect punctuation errors in written text and correct them. Over the past years, numerous studies have been performed on the insertion of punctuations in speech transcripts using supervised techniques. However, it is actually very difficult or even impossible to develop a large high-quality corpus to achieve reliable models for predicting punctuations in speech transcripts or ASR outputs [11].

However, it appears that no research is yet available concerning the problem of punctuation correction in Chinese opinionated text. Furthermore, most previous research on punctuation prediction focused on very shallow linguistic features such as lexical features or prosodic information (viz. pitch and pause duration [9], few studies have been done on the exploration of deeper linguistic features like syntactic structural information for punctuation prediction, particularly in Chinese [5].

In this paper, we attempt to check and correct punctuation errors in Chinese opinionated text under the framework of maximum entropy models (MEMs). To this end, we first built a corpus of product reviews and conducted a survey of punctuation errors in Chinese opinionated text. Then, we formulate Chinese punctuation prediction and correction as a multiple labeling task on a raw sequence of words without punctuation marks. Finally, we employ MEMs to explore three levels of features, namely word-level, phrase-level and functional chunk-level features for Chinese punctuation prediction and correction. Our experimental results show that the introduction of deeper linguistic features is benefits Chinese punctuation prediction and correction.

This paper is organized as follows: In Section 2 we will review in brief the related work on Chinese punctuation prediction and correction. In Section 3 will present a survey of punctuation errors in Chinese opinionated text. In Section 4 we will describe in detail our labeling approach to Chinese

punctuation prediction and correction, followed by the experimental results in Section 5. The conclusions will be given in Section 6.

2 Related work

Punctuation prediction has been well studied in the communities of ASR, and a variety of techniques have been attempted, including n-grams [10][11], maximum entropy models (MEMs) [5][12], and CRFs [9][13].

Current research focuses on seeking informative features for punctuation prediction. Huang and Zweig (2002) attempted to explore POS features and prosodic features for inserting punctuations in automatically recognized speech texts using MEMs [12]. Gravano et al. (2009) examined the effect of different orders of n-grams on performance in punctuation prediction [10]. Takeuchi et al. (2007) exploited silence information from ASR systems and head or tail phrases within sentences [11]. They showed that using head and tail phrases could result in improvement of performance in sentence boundary detection. More recently, Huang and Chen (2011) exploited different features, such as the beginning and end features of voice fragment, character features, word features, POS features, syntactic features and topic features, to label pause and stop in Chinese texts under the framework of CRFs [6].

In addition to speech transcripts and ASR outputs, a number of researchers start to study punctuation prediction via written texts. Tomanek et al (2007) employed CRFs to phrase a biological article, and then inserted punctuation to sentences during sentence segmentation [13]. Xue and Yang (2011) used MEMs to explore contextual words, POS features and syntax trees for inserting commas in Chinese texts [7]. In [3], Laboreiro and Sarmento (2010) applied support vector machines (SVMs) to perform sentence segmentation and punctuation correction on micro-blog texts [3]. The features involved in their study include characters, character types, symbols and punctuations.

From these studies, it is clear that systems with more and deeper features outperform systems only using simple features. However, most previous studies only used lexical cues for punctuation prediction. This might be that a well-annotated corpus of speech texts is not available to date.

This paper is primarily concerned with punctuation correction in Chinese opinionated text. Due to the fact that opinionated texts on the internet are mostly generated by web users and thus ill-formed with punctuation errors, we take punctuation error correction as a process of punctuation prediction. In other words, when the automatically-restored punctuation marks are different from the original counterparts, we thin the former is correct. In addition, unlike previous research, in the present study we approach Chinese punctuation prediction from the perspective of written texts. As such, we attempt to exploit multiple levels of textual features under the framework of MEMs-based sequence labeling and thus examine their role in Chinese punctuation prediction and correction.

3 Punctuation marks in Chinese

In this section we look into the types of punctuation errors in Chinese opinionated text and their distributions after a brief introduction to Chinese punctuation.

3.1 Chinese punctuation

Chinese punctuation uses a different but relatively complicated set of punctuation marks in comparison with some European languages like English. In addition to other symbols such as slash (/) and arrow (→), Chinese text commonly contains sixteen normal punctuation marks that are different in shape and usage. According to their functions, Chinese punctuation marks can be classified into two main categories, namely delimiters and indicators [5]. A delimiter is used to indicate a pause or manner of speaking while an indicator is for marking the nature or role of a statement. Delimiters can be further divided into two sub-types in terms of their location in sentences: sentence-final delimiters at the end of a sentence, including period, question mark and exclamation mark, and sentence-internal delimiters within a sentence, including comma, semicolon, colon and enumeration comma.

Table 1. Tagset for Chinese Punctuation Marks

Tag	Punctuation type	Punctuation marks
FUL	period / full stop	。
EXC	exclamation mark	！
QUE	question mark	？
COM	Comma	，
SEM	Semicolon	；
COL	colon	：
ENU	enumeration comma	、
ELL	Ellipsis	……
DAS	em dash	——
MID	middle dot	·
CON	connection mark	—, ～, --, -,
BRA	Bracket	（, ）, ［, ］, ｛, ｝【, 】
QUO	quotation mark	", ", ', ', ", ", ', ', 「, 」, 『, 』
TIT	title mark	《, 》, 〈, 〉
PRO	proper name marks	＿, ＿＿
EMP	emphasis marks	.

Table 1 defines a set of sixteen tags for Chinese punctuation marks. It should be noted that in the present study we focus on the first fourteen punctuation marks because proper name marks and emphasis marks are inconvenient to input, and are thus rare in real Chinese opinionated text.

3.2 Punctuation errors in opinion text

Three main types of punctuation mistakes commonly appear in Chinese opinionated text. They are misuse of punctuation, punctuation missing and punctuation redundancy. Table 2 illustrates some typical punctuation errors from Chinese product reviews of mobile phones.

Table 2. Typical punctuation errors in Chinese opinionated text

Types of errors	Mistaken punctuation	Standard punctuation	Translation
Punctuation misuse	开机键不好按～～～听筒质量严重不行。	开机键不好按，听筒质量严重不行。	The power button is not good to press, and the quality of earphone is extremely poor.
Punctuation missing	电池不够用，分辨率低	电池不够用，分辨率低。	The battery is insufficient, and the resolution is low.
Punctuation redundancy	性价比还可以，电池也木有，那么费电。	性价比还可以，电池也木有那么费电。	The price-performance is ok, and it does not cost so much electricity.

In order to investigate the distribution of punctuation errors in Chinese opinionated text, we have also built a corpus of reviews on mobile phone products from Jindong online shopping mall (http://www.360buy.com). This corpus contains a total of 7500 opinionated product reviews, in which all mistaken punctuation marks are manually corrected.

Table 3. Distribution of different punctuation errors in Chinese opinionated text

Type	Frequency	Punctuation errors			
		Misused punctuation	Missing punctuation	Redundant punctuation	Total
full stop	7649	323	2958	21	3302
exclamation mark	428	43	201	12	256
question mark	26	4	6	7	17
comma	6193	913	309	90	1312
semicolon	65	6	2	0	8
colon	24	0	3	27	30
enumeration comma	301	57	26	7	90
bracket	76	1	13	7	21
quotation mark	16	0	1	0	1
symbols		0	0	307	307
Total	14778	1347	3519	478	5344

Table 3 shows the distribution of different punctuation errors in the corpus. As can be seen from this table, the corpus contains a total of 14778 punctuation marks, among which 5344 punctuation marks are improperly placed in sentences. This shows in a sense that the problem of punctuation errors is very serious in Chinese online product reviews, which will inevitably bring additional obstacles to product review analysis. From Table 3, we can also observe that among the three main types of punctuation errors, missing punctuations are the most frequent punctuation errors in Chinese opinionated text, accounting for more than 65% of all punctuation errors in the corpus.

4 Chinese punctuation correction

In this section, we first reformulate Chinese punctuation prediction and correction as a multiple pass labeling task on a raw sequence of words with no punctuation, and thus employ MEMs to carry out this task.

4.1 Task formulation

Given an opinionated text, our system will take the following three main steps to perform punctuation correction:

- First, a preprocessing module removes all punctuation marks within the input opinionated

text. Thus, the original text is reduced to an unpunctuated text.

- Second, a linguistic analysis module is to perform word segmentation and part-of-speech tagging, base phrase chunking and functional chunking on the unpunctuated text.

- Third, a multiple-pass punctuation labeling module is used to tag each word in the unpunctuated text with a proper tag indicating its position pattern and the types of punctuation marks if applicable.

- Finally, a correction module is employed to perform punctuation correction by comparing the output of punctuation labeling with the original input.

4.2 Labeling scheme

To achieve punctuation labeling, we employ five tags to represent the patterns of words in punctuation prediction. Table 4 presents the detailed definition of these tags.

Table 4. Tagset for the patterns of words during punctuation labeling

Tag	Definition
B	The preceding word of the current punctuation mark
A	The following word of the current punctuation mark
O	Word that is not adjacent to the current punctuation mark
BOT	Head word of a text
EOT	Tail word of a text

To illustrate this problem, consider the following exemplar text "开机键不好按~~~听筒质量严重不行" (*The power button is not easy to press, and the handset quality is extremely bad.*), along with its word stream and the corresponding labeling representation.

(a) Original text: 开机键不好按~~~听筒质量严重不行。
(b) Unpunctuated text: 开机键不好按听筒质量严重不行
(c) Word segmentation: 开机键/不/好/按/听筒/质量/严重/不行/
(d) Punctuation labeling: 开机键/BOT-O 不/O 好/O 按/B-COM 听筒/A-COM 质量/O 严重/O 不行/EOT-FUL
(e) Punctuation correction: 开机键不好按，听筒质量严重不行。

Fig.1 An example of word-based punctuation labeling for punctuation correction

Fig.1 illustrates the process of word-based labeling for correcting punctuation errors in the above examplar sentence. Obviously, each word within the raw sequence of unpunctuated text will receive a proper punctuation tag. Here, a punctuation tag has the form of t_1-t_2, where t_1 denote the pattern of the current word in punctuation labeling, and t_2 stands for the type of the punctuation that precedes and follows the word if applicable.

4.3 MEMs for punctuation labeling

Let $W=w_1w_2…w_n$ be an input sequence of words, and $T=t_1t_2…t_n$ be a potential sequence of punctuation tags corresponding to W. From a statistical point of view, the task of punctuation labeling is to find the most likely sequence of punctuation tags $\hat{T}$ that maximizes the conditional probability $P(T|W)$ of the punctuation tag sequence given the word sequence over all of its possible punctuation tag sequences $\{T\}$. Maximum entropy modeling decomposes the conditional probability $P(T|W)$ of a sequence of punctuation tags given a sequence of words W into a product of conditional probabilities $P(t_i|h_i)$ ($1 \leq i \leq n$), namely

$$\hat{T} = \arg\max_T P(t_1 t_2 \cdots t_n \mid w_1 w_2 \cdots w_n) \approx \arg\max_T \prod_{i=1}^{n} P(t_i \mid h_i) \quad (1)$$

Where t_i ($1 \leq i \leq n$) is a candidate tag of the word w_i during punctuation labeling, and h_i ($1 \leq i \leq n$) is the history or context of word w_i. The parameters $P(t_i|h_i)$ can be estimated from the training data using Equation (2).

$$\begin{cases} P(t \mid h) = \dfrac{1}{Z(h)} \exp(\sum_{j=1}^{K} \lambda_j f_j(t,h)) \\ Z(h) = \sum_t \exp(\sum_{j=1}^{K} \lambda_j f_j(t,h)) \end{cases} \quad (2)$$

Where $Z(h)$ is a normalization term, and f_i is the jth binary feature function for the current word w_i, associated with a weight λ_j. Its value is 1 if the feature condition is satisfied by the tag-history pair and 0 otherwise.

4.4 Linguistic features

In the present study, we consider three levels of linguistics features for Chinese punctuation prediction and correction, namely words, phrases and functional chunks. It should be noted that the Tsinghua tagsets for POS, phrases and functional chunks are used in our current study, respectively. The detailed definitions for the three tagsets can be seen in (Zhou, 2004).

Word-level features. At word level, we exploit word forms and their part-of-speech (POS) tags in a window of three words, including the current word w_i, the preceding word w_{i-1} and the following word w_{i+1}, and their respective POS tags t_i, t_{i-1}, and t_{i+1}. Table 5 details the word-level

feature template for Chinese punctuation prediction and correction.

Table 5. Word-level features for punctuation labeling

No.	Feature	Definition
L1	$w_{i-1}w_i$	The preceding word and the current word
L2	$w_{i-1}w_{i+1}$	The current word and the following word
L3	$w_{i-1}t_i$	The preceding word and the current word's POS tag
L4	t_iw_{i+1}	The current word's POS tag and the following word
L5	$t_{i-1}w_i$	The preceding word's POS tag and the current word
L6	w_it_{i+1}	The current word and the following word's POS tag
L7	w_i	The current word

Phrase-level features. At phrase level, we consider some possible combinations of the current word, the preceding word, the following word and their relevant phrase tags as features for punctuation prediction and correction. The template for phrase-level is given in detail in Table 6. Where, p_i, p_{i-1} and p_{i+1} denote the category tags of the phrases containing words w_i, w_{i-1} and w_{i+1}, respectively.

Table 6. Phrase-level features for punctuation labeling

No.	Feature	Definition
P1	$w_{i-1}\ p_{i-1}w_i\ p_i$	The preceding word and its phrase tag, the current word and its phrase tag
P2	$w_ip_iw_{i+1}\ p_{i+1}$	The current word and its phrase tag, the following word and its phrase tag
P3	$w_{i-1}\ p_{i-1}t_i\ p_i$	The preceding word and its phrase tag, the current word's POS and phrase tag
P4	$t_ip_iw_{i+1}\ p_{i+1}$	The current word's POS and phrase tag, the following word and its phrase tag
P5	$t_{i-1}\ p_{i-1}w_i\ p_i$	The preceding word's POS and phrase tag, the current word and its phrase tag
P6	$w_ip_it_{i+1}\ p_{i+1}$	The current word and its phrase tag, the following word's POS and phrase tag
P7	$p_{i-1}w_ip_i$	The preceding word's phrase tag, the current word and its phrase tag
P8	$w_i\ p_ip_{i+1}$	The current word and its phrase tag, the following word's phrase tag
P9	$p_{i-1}t_ip_i$	The preceding word's phrase tag, the current word's POS and phrase tag
P10	$t_i\ p_ip_{i+1}$	The current word's POS and phrase tag, the following word's phrase tag

Functional chunk-level features. At functional chunk level, we consider some possible combinations of the current word, the preceding word, the following word and their relevant functional chunk tags as features for punctuation prediction and correction. The templates for functional chunk-level features is given in Table 7. Where, f_i, f_{i-1} and f_{i+1}

stand for the category tags of the functional chunks containing words w_i, w_{i-1} and w_{i+1}, respectively.

Table 7. Functional chunk-level features for punctuation labeling

No.	Feature	Definition
F1	$w_{i-1}\ f_{i-1}w_i\ f_i$	The preceding word and its functional chunk tag, the current word and its functional chunk tag
F2	$w_if_iw_{i+1}f_{i+1}$	The current word and its functional chunk tag, the following word and its functional chunk tag
F3	$w_{i-1}\ f_{i-1}t_if_i$	The preceding word and its functional chunk tag, the current word's POS and its functional chunk tag
F4	$t_if_iw_{i+1}f_{i+1}$	The current word's POS and its functional chunk tag, the following word and its functional chunk tag
F5	$t_{i-1}\ f_{i-1}w_if_i$	The preceding word's POS and functional chunk tag, the current word and its functional chunk tag
F6	$w_if_it_{i+1}f_{i+1}$	The current word and its functional chunk tag, the following word's POS and functional chunk tag
F7	$f_{i-1}w_if_i$	The preceding word's functional chunk tag, the current word and its functional chunk tag
F8	$w_if_if_{i+1}$	The current word and its functional chunk tag, the following word's functional chunk tag
F9	$f_{i-1}t_if_i$	The preceding word's functional chunk tag, the current word's POS and its functional chunk tag
F10	$t_if_if_{i+1}$	The current word's POS and its functional chunk tag, the following word's functional chunk tag

5   Experiments

To evaluate our approach, we have conducted several experiments on different corpora. This section will present the relevant results.

5.1   Experimental setup

We use two datasets in our experiments: the Tsinghua University treebank (Zhou, 2004) is used to examine the role of different levels of features on Chinese punctuation prediction, and the above corpus of reviews on mobile phone products extracted from Jindong online shopping mall (http://www.360buy.com) is for evaluating punctuation correction performance in Chinese opinionated text. In particular, we divide these corpora into training datasets and test datasets, respectively. Table 8 shows the basic statistics of the experimental datasets.

Table 8. Basic statistics of the experimental datasets

Corpus	Training dataset		Test dataset	
	# sent.	# punct. (Total /Average per sent.)	# sent.	# punct. (Total / Average per sent.)
Tsinghua Treebank	14248	57865 / 4.06	3751	13515/ 3.60
The opinion corpus	5095	9832 / 1.93	2548	4946 / 1.94

As can be seen from Table 8, the Tsinghua Treebank has a larger average number of punctuation marks per sentence compared with the product review corpus. The reason may be because that the Tsinghua Treebank is actually a corpus of news text. As a formal language, sentences in news text are generally longer and have more complicated structures than opinionated sentences, and thus contain more punctuation marks on average.

In addition, we employ three measures to score the performance of our system in punctuation prediction and correction, namely the precision (denoted by P), the recall (denoted by R) and the F-score.

5.2 Experimental results

We conducted two main experiments: The first experiment is designed to examine the effects of different levels of features on punctuation prediction over the Tsinghua Treebank while the second on is further to investigate the role of these features on punctuation correction using the opinion text corpus. Table 9 and Table 10 present the respective experimental results.

Table 9. Performance of Chinese punctuation prediction using different levels of features

Features	P	R	F
Word-level	0.587	0.553	0.570
Phrase-level	0.665	0.630	0.647
Functional chunk-level	0.728	0.631	0.676

Table 10. Performance of Chinese punctuation correction using different levels of features

Features	P	R	F
Word-level	0.603	0.883	0.717
Phrase-level	0.675	0.908	0.775
Functional chunk-level	0.709	0.913	0.798

From Table 9 and Table 10 we can see that word-level features can yield F-scores of 0.57 and 0.717 for punctuation prediction and punctuation correction, respectively. The relevant figures can rise to 0.676 and 0.798 after using the functional chunk-level features. This proves in a sense that

deeper linguistic features is of great value to Chinese punctuation prediction and correction.

6 Conclusions

In this paper, we proposed a MEMs-based multiple-pass labeling approach to punctuation prediction and correction in Chinese opinionated text. In particular, we have investigated the distribution of different types of punctuation errors in real Chinese opinionated text. Furthermore, we have also examined the effect of three levels of linguistic features, namely words, phrases and functional chunks on Chinese punctuation prediction and correction. We have also evaluated our approach on the Tsinghua Treebank and a corpus of product reviews from Jindong online shopping mall (http://www.360buy.com). The experimental results show that using multiple deeper features is of great value to Chinese punctuation prediction and correction.

Although the proposed method yields good results for periods and commas, the prediction of brackets, quotations and title identifier is still not satisfactory. This might be due to the data sparseness caused by the small number of these punctuations. Another possible reason is that the features in use are not effective or informative for these punctuation marks. Therefore, in the future research we plan to improve our current system by expanding the scale of the training corpus and seeking more informative features for Chinese punctuation prediction and correction.

7 Acknowledgements

8 References

[1] L. Dey, and S.M. Haque. Opinion mining from noisy text data. In *Proceedings of AND '08*, pages 83-90, 2008.

[2] L. V. Subramanian, S. Roy, T. A. Faruquie, and S. Negi. A survey of types of text noise and techniques to handle noisy text. In *Proceedings of AND '09*, pages 115-122, 2009.

[3] G. Laboreiro, and L. Sarmento. Tokenizing micro-blogging messages using a text classification approach. In *Proceedings of AND '10*, pages 81-87, 2010.

[4] Q. Zhou. Annotation scheme for Chinese treebank. Jounal of Chinese information processing, 18(4): 1-8, 2004.

[5]   Y. Guo, H. Wang, and J. V. Genabith. A linguistically inspired statistical model for Chinese punctuation generation. ACM Transactions on Asian Language Information Processing, 9(2), 2010.

[6]   H.-H. Huang, and H.-H. Chen. Pause and stop labeling for Chinese sentence boundary detection. In *Proceedings of Recent Advances in Natural Language Processing (RANLP'11)*, pages 146-153, 2011.

[7]   N. Xue, and Y. Yang. Chinese sentence segmentation as comma classification. In *Proceedings of ACL '11*, pages 631-635, 2011.

[8]   E. Matusov, A. Mauser, and H. Ney. Automatic sentence segmentation and punctuation prediction for spoken language translation. In *Proceedings of IWSLT'06*, pages158-165, 2006.

[9]   W. Lu, and H. T. Ng. Better punctuation prediction with dynamic conditional random fields. In *Proceedings of EMNLP '10*, pages 177-186, 2010.

[10] A. Gravano, M. Jansche, and M. Bacchiani. Restoring punctuation and capitalization in transcribed speech. In *Proceedings of ICASSP'09*, pages 4741-4744, 2009.

[11] H. Takeuchi, L. V. Subramaniam, S. Roy, D. Punjani, and T. Nasukawa. Sentence boundary detection in conversational speech transcripts using noisily labeled examples. International Journal of Document Analysis and Recognition, 10(3): 147-155, 2007.

[12] J. Huang, G. Zweig. Maximum entropy model for punctuation annotation from speech. *Proceedings of ICSLP' 02*, pages 917-920, 2002.

[13] K. Tomanek, J. Wermter, and U. Hahn. Sentence and token splitting based on conditional random fields. *Proceedings of PACLING '07*, pages. 49-57, 2007.

Forecasting meteorology models implemented by the feed-forward architecture of neural networks

J. Moreno, J. Serrano, and J. Corredor
Engineering and Achitecture Department
Universidad de Pamplona, Villa del Rosario, Colombia.
{jorge.morenoj, jcserrano, john.corredor}@unipamplona.edu.co

Abstract— *Forecasting meteorology models has become a main tool, in order to obtain energy exploitation, in turn ecologically. Currently Artificial Intelligence (AI) has changed the approach to solve the problems of prediction, among others. The change in approach is due to the AI that has modern learning techniques with mathematical models closer to reality. Our approach is to implement feed-forward neural networks architecture into wind forecasting models. The objective of our work is to predict the mean monthly wind speed behavior for year 2012 and beyond, as close to reality in the city of San José de Cúcuta-Colombia.*

Keywords: neural networks, feedback architecture, wind energy, forecasting meteorology

1. Introduction

Currently the countries are researching how the renewable energy sources enhance their contribution (clean, green energy), due to the high demand for energy worldwide. One of the alternative energy sources more use, is wind energy, which represents 10% of energy in Europe [6]. Wind energy differs from conventional energy sources due its stochastic nature, because the wind speed is considered one of the most difficult meteorological parameters to modeling [5]. Forecasting meteorological play a key role in the challenge of balancing supply and demand on electrical consumption systems. Neural networks are a great tool in science to make predictions, offering robust solutions and easy implementation [2].

In order to obtain a better prediction and less uncertainty level, we implement a forecasting model using feedforward neural network architecture. The meteorological data diversity is obtained using different neural network configuration. For prediction, we have a set of twelve neural network configurations (one by month) [6]. We propose use two data types for feed of neural networks, provided by Institute of Hydrology, Meteorology and Environmental Studies (IDEAM - Colombia). In the medium term, our forecasting meteorology models implemented by the feedforward architecture of neural networks, will be able to predict the monthly average wind speed in the city of San José de Cúcuta-Colombia, for 2012 and next years.

This work is organized as follows. Section 2 presents in detail the characteristics of neural networks and state of art, describing its main components. Section 3 describes the process performed to input data of the neural network to complete the information (incomplete data given by IDEAM - Colombia). Section 4 presents methodology used to design the neural network, and also presents the experiments conducted to evaluate the performance of the different approaches proposed. Section 5, show results of prediction the monthly average wind speed in the city of Cucuta-Colombia, for 2012. Section 6 closes the paper, showing some conclusions and future work.

2. Artificial Neural Networks

The Artificial Neural Networks (ANNs) are known as the "*universal approximators*" and "*computational models*" with particular characteristics such as the ability to learn or adapt, to organize or to generalize data. The basis of neural networks is the neuron, this represents the smallest unit implemented. The artificial neurons are interconnected through a synaptic weight (which is a numeric value). The synaptic weight is sent to the nucleus of the neural network to perform calculations inside. The input value of neuron *i* (**net *i***) to the activation function is described in 1, as the sum of the synaptic weights multiplied by the input neurons.

$$net_{=} \sum_{i=1}^{N} W_i j * Y_j \qquad (1)$$

There are a number of papers related to the weather forecast using neural networks, then we describes the most significant articles.

In Lei Ma et al [4], they study the different models that are implemented for the prediction of wind speed, their advantages and drawbacks. Through their observations, we have observed advantages of using artificial intelligence in this research.

In [6], the authors emphasize the importance of generating multiple neural network to achieving better prediction

results, and highlights the value of having a variety of data. Similar our work in the creation of 12 neural networks to improve the system accuracy.

In [4], the authors implemented neural networks to predict the wind speed in a wind park, based on a regional mesoscale model in order to increase accuracy in the short term and thus reduce the problem of accuracy of the neural network, as the time range to predict increases. With this article, the training data has been organized such that the neural network could have a great degree of precision, even though the network have to predict the average speed on a wide scale of time..

R.E Abdel et. al. [7] highlights the importance of optimal treatment of input data in the neural network, in order to achieve lower noise and error that the neural network could generate. With this point of view, were designed different neural networks using different types of input data to reduce noise and the corresponding errors.

E. Rivera et al. [1] they implement data about a large margin of time (data up to 7 years antique) and tested with different neuron network settings, changing both the number of layers as the number of neuron in the input layer. Through this approach climate data, we obtained from the city of Cucuta, with an antiquity of 17 years and prioritized change the number of neurons in the input layer, to test different versions of the neural network in order to obtain a better prediction

G. Wang et al. [2], have implemented a modification of neural networks for prediction, specifically the EMD-FNN model (empirical mode decomposition based on neural networks) also evaluated different configurations of data, varying the delta delay on each. Based on the different configurations tested, was taken into account varying the time delta networks in order to obtain a variation in the number of elements.

3. Pre-processing information.

We use in this paper four types of data: monthly average wind speed, average monthly temperature, distance traveled daily wind, mean daily temperature[2]. These data were supplied by the IDEAM-Colombia, from 1995 until 2012, but with the drawback of being incomplete. To complete and optimize the data, we used two techniques, statistical and artificial intelligence[1].

3.1 Statistical technique

A statistical technique was implemented in order to complete daily data regarding the temperature and distance traveled of winds. This technique consists of the average of next day, previous day, same day from previous month and the same day of next month the date desired to completing. Figure 1 and equation 2 represents the statistical technique.

$$n = \frac{18 + 21 + 13 + 18}{4} \qquad (2)$$

Fig. 1: Statistical technique in order to complete daily data

Number	Input	Hidden Layout	Average Error
1 neural	28	1	**0,277**
2 neural	28	2	0,731
3 neural	28	3	0,550
4 neural	28	4	0,562
5 neural	28	5	**0,258**
10 neural	28	10	0,274
25 neural	28	25	0,404
50 neural	28	50	0,369

Table 1: Average errors obtained in data pre-processing.

3.2 Neural Networks

We implemented a neural network in order to obtain the missing monthly average wind speed data. We performed different versions of neural network, changed the neural input numbers and the numbers of neural in hidden layer. In Table 1 the input layer have been tested with 28 Neural Networks and the average error obtained is show.

As discussed previously, we implemented a neural network to complete the missing data from 1995 to 2005. In order to complete the data, we have used data from 88 months distributed as follows: 64 for the training stage and 24 for validation stage.

In the training stage is made different versions of neural networks by changing the number of neurons in the hidden layer. Table 1 shows the average errors obtained. Average errors are calculated with Equation 3, n number of data used in the validation stage, *expected output* is the average wind speed obtained by the neural network and *real output is* the

average wind speed real.

$$Av.error = \frac{\sum_{i=1}^{n}(expectedOutput - realOutput)}{n} \quad (3)$$

From the experience acquired in this research, neural networks with high degree of precision, containing 28 neurons in the hidden layer. Because there are no standards for neurons in the hidden layer, different amounts were tested up to 50, a number that was considered sufficient to solve the problem. Table 1 shows that there is not a direct relationship between the number of neurons in the hidden layer and the average error.

From Table 1 shows that *1 neural* and *5 neural* have the lowest average error. We choose *1 neural* because the difference between the two neural average error is not very high (*1 neural* and *5 neural networks*). In addition, the computational cost to process data with *1 neural network* is much lower.

In order to observe better the behavior about wind speed, we organize the data in two groups. Figures 2 and 3, show the two groups, X-axis represents the wind speed, Y-axis represents the months of the year. Additionally Figures 2 and 3 show the average wind speed at 5mps lower and upper respectively.
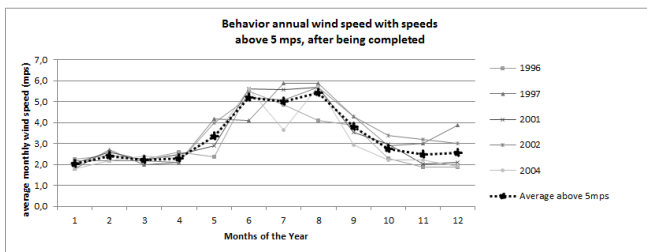
Fig. 2: Behavior annual wind speed with speeds above 5 mps, after being completed.
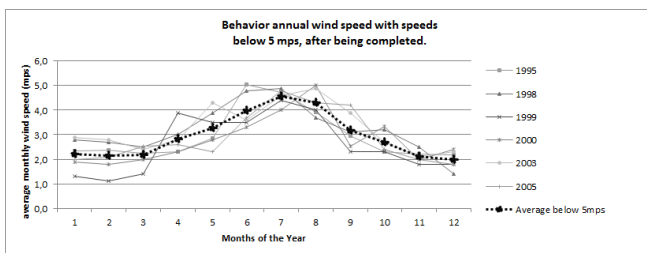
Fig. 3: Behavior annual wind speed with speeds below 5 mps, after being completed.

Type	Input	Output
Daytime temperature	x	
Wind distance	x	
Monthly average wind speed		x

Table 2: Distribution of the input and output data.

4. Methodology

4.1 Process

In order to obtain the prediction, we proceed to design and modeling the neural networks, with the previously processed data. The process flow is shown in Fig. 4.
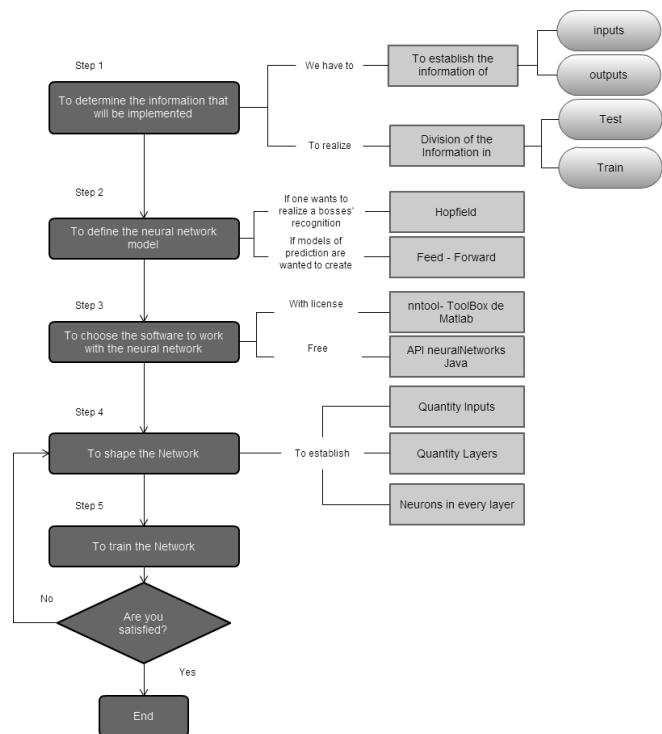
Fig. 4: Process flow: modeling neural networks.

4.2 Step 1

With the purpose of having a higher accuracy, we have organized the data input and output of the neural network [7]. Table 2 describes how data is organized.

In order to train the neural networks, data were used corresponds for the months: January, February, March, April, June and July from 1995 to 2012. In turn, we have taken into account for the years 1995-2011 from May to December, this because only data monthly average wind speed for such dates were available.

Month	Days	Inputs
January, March, May, July, August, October, December	31	62
April, June, September, November	30	60
February	28	56

Table 3: Number of input, organized by month.

4.3 Step 2

To perform a prediction model by implementing neural networks, is chosen Feed-Forward architecture, model widely used due to its excellent ability in learning from experience [4].

4.4 Step 3

For this research we have chosen MATLAB simulation software, specifically *toolbox nntool*.

4.5 Step 4

This step begins to the neural network model, which defines three guidelines: number of inputs, number of layers and neurons in each layer.

The following equation (4) describes the number of inputs that neural network should have.

$$inputs = d * i \qquad (4)$$

In (4), d corresponds to the numbers of days of the month, and i numbers of data type. With equation (4) we have completed the missing data for the Table 4.

There is no rule that describes the number of layers to be used in a neural network. However, it is acceptable to a network with three layers, with a feed-forward architecture [2].

After several cycles of training, the following Equation 5 was used to obtain the error rate of the neural network.

$$[h!]error = |\frac{Real - expected}{expected} * 100| \qquad (5)$$

The error rates are shown in Table 4, where each row represents the month of the year, and each column the number of neurons in the hidden layer.

5. Prediction

Figure 5 shows the prediction of 2012, X-axis represents the wind speed, Y-axis represents the months of the year. We can observed the behavior as binomial distribution very similar to those above and below average 5mp (Figures 1 and 2 respectively).

Number of neural in hidden layer				
1	2	3	4	5
Jan. 26.67	66.67	**0.57**	60.00	53.55
Feb. 16.40	14.95	**1.05**	27.78	30.88
Mar. 16.40	26.55	28.62	36.00	**5.41**
Apr. **6.12**	98.98	8.37	52.73	42.61
May 26.40	7.25	19.83	20.09	**5.31**
Jun. 28.33	13.92	17.11	10.27	**1.82**
Jul. 15.65	10.80	**4.13**	26.06	16.19
Aug. **2.55**	20.46	17.45	14.12	51.24
Sep. 16.13	41.59	32.14	48.04	**3.99**
Oct. 4,24	53.90	**3,41**	22,70	12,85
Nov. 21.72	**0.01**	31.88	98.86	31.06
Dec. **0.57**	7.44	12.49	14.74	4.72

Table 4: Errors generated by the neural network in the prediction stage.
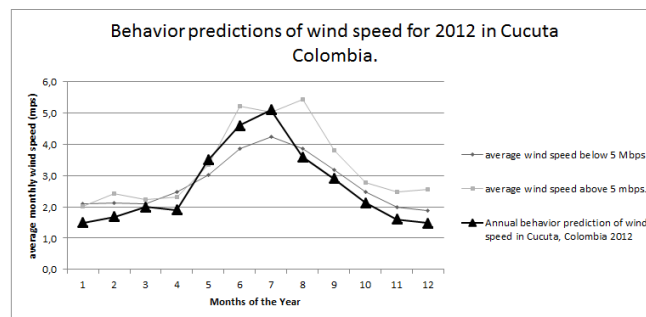
Fig. 5: Behavior prediction wind speed for year 2012 in Cucuta-Colombia

6. Conclusion

In the particular case of Cucuta-Colombia, we decided to start the analysis with neural network with 31 inputs, and only one type of data (wind distance). However, when moving to neural network models with 62, 60 and 56 inputs and 2 inputs data type (wind distance and temperature daily) the performance and accuracy of the neural networks increase.

We also noted that according to Table 1 is not necessary to have large numbers of neurons in the hidden layer, in order to increase performance and accuracy of the neural

network. By increasing the number of neurons in the hidden layer increases the computing needs without increasing the accuracy of neural networks in prediction (Precision errors are obtained below 6.2% in Table 1 of neural networks with hidden layer composed of up to 5 neuron networks).

This research is a very useful contribution to business and local government to make use of alternative energy sources. The prediction of the behavior of the average wind speed is very successful, and this tool can be used successfully to predict the behavior of the wind for the next few years. With precision stability of neural networks, the developed model is an effective tool for the operation and planning of wind parks [3].

7. Acknowledgments

References

[1] R.E. Abdel-Aal, M.A. Elhadidy, and S.M. Shaahid. Modeling and forecasting the mean hourly wind speed time series using gmdh-based abductive networks. *Renewable Energy*, 34(7):1686 – 1699, 2009.

[2] E. Cadenas and W. Rivera. Short term wind speed forecasting in la venta, oaxaca, mexico, using artificial neural networks. *Renewable Energy*, 34(1):274 – 278, 2009.

[3] Z. Guo, W. Zhao, H. Lu, and J. Wang. Multi-step forecasting for wind speed using a modified emd-based artificial neural network model. *Renewable Energy*, 37(1):241 – 249, 2012.

[4] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan. A review on the forecasting of wind speed and generated power. *Renewable and Sustainable Energy Reviews*, 13(4):915 – 920, 2009.

[5] I. Ramirez-R and L. Fernandez-J. An advanced model for short-term forecasting of mean wind speed and wind electric power. *Control and intelligent systems*, 32(1):21–26, 2004. eng.

[6] S. Salcedo-Sanz, A. Pérez-Bellido, E. Ortiz-García, A. Portilla-Figueras, L. Prieto, and F. Correoso. Accurate short-term wind speed prediction by exploiting diversity in input data using banks of artificial neural networks. *Neurocomputing*, 72(46):1336 – 1341, 2009. Brain Inspired Cognitive Systems (BICS 2006) / Interplay Between Natural and Artificial Computation (IWINAC 2007).

[7] S. Salcedo-Sanz, A. Pérez-Bellido, E. Ortiz-García, A. Portilla-Figueras, L. Prieto, and D. Paredes. Hybridizing the fifth generation mesoscale model with artificial neural networks for short-term wind speed prediction. *Renewable Energy*, 34(6):1451 – 1457, 2009.

Genetic Algorithm with Double Chromosome for Prediction of Protein Concentration on Wheat

Kelton de Souza Santiago,
Anderson Silva Soares,
Telma Woerle de Lima
and Gustavo Teodoro Laureano
Federal University of Goias.
Goiania, Brazil
Email: anderson@inf.ufg.br

Clarimar José Coelho
Pontifical University Catholic of Goias.
Goiania, Brazil

Arlindo Rodrigues Galvão Filho
Department of System and Control
Technological Institute of Aeronautics
São José dos Campos, Brazil

Abstract—One of the main problems of quantitative analytical chemistry is to estimate the concentration of one or more species from the values of certain physicochemical properties of the system of interest. For this it is necessary to construct a calibration model, ie, to determine the relationship between measured properties and concentrations. The multivariate calibration is one of the most successful combinations of statistical methods to chemical data, both in analytical chemistry and in theoretical chemistry. Among used methods can cite Artificial Neural Networks(ANN), the Nonlinear Partial Least Squares(N-PLS), Principal Components Regression(PCR), Multiple Linear Regression(MLR) and Genetic Algorithm (GA). This paper investigates the effect of the genetic algorithm based on double chromosome (GADC) to the samples and variables selection applied to the multivariate calibration problems. The obtained results were compared with the well-known algorithms for samples and variable selection Kennard-Stone, Partial Least Square and Successive Projection Algorithm.

I. INTRODUCTION

The term multivariate calibration refers to the construction of a mathematical model to estimate a quantity of interest on the basis of measured values of a set of explanatory variables. Among the traditional technics for construct this model, we can cite the Multiple Linear Regression (MLR) where the data are modelled using linear predictor functions, and unknown model parameters are estimated from the data. Given a data set

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ & \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{bmatrix}, \quad (1)$$

a linear regression model assumes that the relationship between the dependent variable y_n and the p-vector of regressors x_n is linear. Thus the model takes the form

$$y = x_0 b_0 + x_1 b_1 + \ldots + x_{J-1} b_{J-1} + \epsilon \quad (2)$$

or in vectorial notation,

$$Y = X\beta + \epsilon \quad (3)$$

with $x = [x_0 \; x_1 \ldots x_{J-1}]$ is the vector of measured values, $\beta = [b_0 \; b_1 \ldots b_{J-1}]^T$ is the vector to be determined and ϵ is a part of random error.

The MLR method is the simplest form to determine the coefficients vector β. In this case the MLR is given by Equation (4).

$$\beta = (X^T X)^{-1} X^T Y \quad (4)$$

and new concentrations can be estimate from β like as

$$\hat{Y} = X\beta \quad (5)$$

and the prediction ability can be measured by Root Mean Square Error of Prediction (RMSEP) in Equation (6)

$$\text{RMSEP} = \frac{\sum_{i=0}^{N} (\hat{y}_i - y_i)^2}{N} \quad (6)$$

where $\hat{y}$ is the predicted value obtained by Equation (5), y is the real value of the concentration and N the total number of samples.

The simple question is, which samples should be use for calibration in Equation (4) that minimizes the Equation (6). Many algorithms were proposed for this task. Among several we can to cite the Kennard-Stone (KS), proposed by Kennard and Stone [2] and their variant, sample set partitioning based on joint X-Y distances, SPXY developed by Kawakami et al [3]. Additionally we also can to use a random division or used the classical cross validation method. However the random division have problems of replicability and the cross validation has a high computational cost.

Still on the Equation (4), note that it have a portion $(X^T X)$ that your result must have full column rank p, otherwise, we have a condition known as multicollinearity in the predictor variables caused by inversion instability. If the model has more variables than equations, the equation system is ill-conditioned. As a small example consider the Equation (7)

$$y = b_1 * x_1 + b_2 * x_2 \quad (7)$$

If $x_2 = 2 * x_1$, the model can be rewritten like

$$y = b_1 * x_1 + b_2 * (2 * x_1) = (b_1 + 2 * b_2) * x_1 \qquad (8)$$

From Equation (8) we can to conclude that there isn't a unique value for b_1 and b_2, problem known as ill-conditioned. In this sense, search algorithm like as genetic algorithm can to be used to find a subset of variables that minimize the multicollinearity among the variables and consequently the prediction error of Equation (6).

The proposed genetic algorithm adopts a double chromosome structure where the first chromosome contains the variables selected and the second chromosome contains the samples to be used in Equation 4. The basic idea is evolve the chromosome double in parallel in order to select samples and variables simultaneously. As a case study we used a real world application problem involves the protein content prediction on wheat samples with 775 variables and 683 samples.

II.   SAMPLE SELECTION ALGORITHMS

The focus of the sample selection algorithm is to decide which sample to store for generalization. Storing many samples can result in storage requirement and slow running and this leads to overfitting when predicting.

This section approaches the theoretical basis regarding the classic algorithms used for selecting samples which also were adopted for tests in this research. They are: Random selection, Kennard-Stone (KS) and the partitioning sample set based on the xy distance (SPXY) algorithm.

The random selection method uses pseudo-random number generators to select samples for the calibration and validation sets. It is the most simple method to perform samples selection.

Proposed by Kennard and Stone, KS algorithm is well known between the analytical chemists to perform samples selection [3], [4]. Typically, this algorithm is applied to perform the selection of samples to compose the calibration set, since it carries the selection of samples greater variability. The selection criterion is the distance between the samples.

The last algorithm for sample selection was proposed in 2005, the Sample Set Partitioning Based on Joint X-Y distance (SPXY) is a variant of KS algorithm [5]. SPXY increases the distance defined by Kennard-Stone calculating a distance to the dependent variable y for the sample in question. The algorithm SPXY is used to separate the set of samples in calibration set and validation set [5].

III.   VARIABLES SELECTION ALGORITHMS

One of the major difficulty of multivariate analysis consists of selecting a combination of variables that lead to model optimization. One of the practical problems is to identify how many and what wavelengths should be chosen, especially when high spectral overlap occurs. Several mathematical algorithms have been developed in an attempt to avoid this problem [6].

The variable selection methods search to produce more simple or parsimonious models. The search for the subset of variables consists of a combinatorial optimization problem driven by an objective function. Restrictions on combinations and cost functions define the strategy of the selection algorithm. Despite several proposals of variable selection algorithms reported in literature [7]–[11], this is still a topic of discussion in chemometrics and related fields.

The successive projections algorithm (SPA) was proposed in 2001 by [12], with the aim of selecting variables to build multivariate models using UV-VIS spectrometer measures. However, over the past few years the SPA has been widely used in multivariate calibration, classification, selection of samples, calibration transfer, involving modeling structure activity (QSAR) and selection of wavelet coefficients in the field [12].

The essence of SPA consists in performing operations on the projection X_{cal} calibration matrix ($K_c x J$) whose rows and columns correspond to K_c calibration samples and J, the spectral variables [12].

The partial least squares regression (PLS) [1] is a method for regression on factors whose objective is the prediction of a set of output variables Y based on the observation of a set input variable X in the absence of a theoretical method. It is intended for a large number of input variables compared to the number of samples.

The construction of a PLS model requires a set of samples along with the value of dependent variables. Thus, X is the matrix containing the specimen into the rows and the Y matrix containing the values for prediction in their rows, the PLS regression takes as models simultaneously latent variables inherent both X and Y. These factors are then used to define a subspace X that best suits modeling of Y.

IV.   GENETIC ALGORITHMS

Genetic algorithms (GA) are a global search heuristic inspired on the natural evolution of species and in the natural biological process [13].

Basically, a GA creates a population of possible solutions to the problem being solved and then submit these solutions to the evolution process. Genetic operators are applied to transform the population in every generation, in order to created better individuals. The main operators responsible for the population diversification well known in the literature are crossover (or recombination) and mutation [13].

The main advantages of GAs are their robustness and applicability in a wide variety of problems. GAs requires no knowledge or information of the surface gradients defined by the objective function and the search performance undergoes little or no effect on discontinuities or surface complexity.

However, GAs have some drawbacks such as the difficulty to find the accurate global optimum, require a large number of fitness ratings functions, and also a great possibility of settings that can complicate the problem treated resolution.

V.   GENETIC ALGORITHM BASED ON DOUBLE CROMOSOME (GADC)

The genetic algorithm proposed on this paper (GADC) adopts all main characteristics of a typical GA. Since this genetic algorithm makes the selection of variables and samples, a representation were adopted which helps in simplifying

the process, it's called double chromosome. Where the first chromosome contains the selected variables and the second chromosome contains the division of the samples in calibration and validation sets.

Algorithm 1 GADC

$t \leftarrow 0$

InitialPopulation($Pop(t)$) {Creates a initial population with random 0s and 1s for the two chromosomes}

Evaluates($POP(t)$) {Evaluates the individuals using the fitness function RMSEP}

while some stop criteria **do**

    $POP' \leftarrow$ ParentSelection($POP(t)$) {Selects the best individuals for reproduction}

    Reproduction(POP') {Generates new individuals using crossover and mutation operators in POP'}

    Evaluates(POP')

    $POP(t+1) \leftarrow$ Selects($POP(t), POP'$)

    $t \leftarrow t+1$

end while

Algorithm 1 presents the basic structure of GADC. Follows we will describe the main functions used by GADC to perform the evolutionary process. First of all, function *InitialPopulation* creates the initial solutions by a random process that fills the first chromosome with zeros and ones as much as the number of variables and the second chromosome in the same way, but considering the number of available samples to separate between calibration and validation sets.

Function *Evaluates* evaluates each individual in this way: the ones in the first chromosome indicates the variables selected and the ones in the second chromosome indicates the samples that will include in the calibration set. So, the the regression model by multiple linear regression (MLR) is builded using the variables and samples indicated by chromosome. The zeros in second chromosome indicates the samples not will be include in the calibration set, but will compose the validation set in order to calculate the fitness by Equation 6.

The parents used in the reproduction process are selected by function *ParentSelection*. This method selects the 20% best solutions to participate of the reproduction as one of the parents and the other parent is randomly choose from the remain individuals.

The reproduction process is made using uniform crossover and flip mutation. In the uniform crossover the double chromosome are considered as a unique chromosome. The mask vector of the uniform crossover is created with same size of the individual and randomly populated with binary values. This mask vector indicates that when its value is one, the son receives the allele from *parent_1*, and if the value of mask vector is 0, the new individual receives the allele from *parent_2*. The mutation probability is adopted only to select a individual for mutation. After the individual is choose a randomly position is mutate using the flip mutation. Once again, we considered the two chromosomes as a unique form.

In order to compose the next generation function *Selects* uses the percent of best solutions from the actual population and the other individuals come from the new population obtained by the reproduction operators.

VI. Materials and Methods

The data set for multivariate calibration study is the same used by [14], that consists of 755 visible near infrared spectra of whole-kernel wheat samples, which were initially used as shoot-out data in the 2008 International Diffuse Reflectance Conference (once http://www.idrc-chambersburg.org/shootout.html), and protein content is chosen as the property of interest. The following tests were performed to evaluate the proposed method GADC.

- The Pseudo-random selection algorithm was applied using a function to randomly separate data into calibration, validation and prediction sets with 300, 300 and 175 samples, respectively.

- The Kennard-Stone (KS) algorithm [3] was applied to the derivative spectra to separate data into calibration, validation and prediction sets with 301, 237 and 237 samples, respectively. The same parameters were applied also to SPXY algorithm [5].

- The SPA and PLS variable selection algorithms were applied in conjunction with pseudo-random number generator (PNG), KS and SPXY algorithms.

- GADC uses a set of 500 samples to make an effectively division between calibration and validation sets. The prediction set number is equals 275. The GADC function receives as parameters the X and Y matrices, the number of generations and the population size.

All tests were carried out by using a desktop computer with an Intel®Core(TM) i3-2100 processor (3.1 GHz) and 4 GB of RAM memory and Matlab 7.13. The tests that involved randomness were performed exhaustively in order to obtain a standard deviation.

VII. Results

Table I presents the results obtained using the algorithms KS and SPXY to calibration sample selections only, without variable selection. These values are the RMSEP in the prediction set. The SPXY algorithm had a less RMSEP than KS algorithm.

TABLE I.   PREDICTION RESULTS FOR KS AND SPXY ALGORITHM FOR SAMPLE SELECTION WITHOUT VARIABLE SELECTION.

	KS	SPXY
RMSEP	2.8270	1.4567

Table II presents the results obtained using algorithms for sample and variable selection PNG-SPA and PNG-PLS. As can be seem the use of variable selection reduce the prediction error when compared with previous results. Despite the sample selection have been taken at random method, the variable selection algorithms reduced the prediction error.

Table III shows the results for KS-SPA, SPXY-SPA, KS-PLS and SPXY-PLS algorithms. The minimum RMSEP was obtained combining the SPXY sample selection and PLS algorithms. However, remember that PLS uses all original variables to build new transformed variables. Therefore, in practice, it requires all variables.

TABLE II.   PREDICTION RESULTS FOR PNG-SPA AND PNG-PLS ALGORITHM. IT WAS TESTED 50 TIMES AND CALCULATED ITS STANDARD DEVIATION

	PNG-SPA	PNG-PLS
Average Number of Variables	24	22
Average RMSEP	0.2373	0.2070
Minimum RMSEP	0.2171	0.1777
Maximum RMSEP	0.2517	0.2245
Sdv RMSEP	0.0097	0.0356

TABLE III.   PREDICTION RESULTS FOR KS-SPA, SPXY-SPA, KS-PLS AND SPXY-PLS ALGORITHMS FOR PROTEIN CONTENT IN THE WHEAT DATA SET

	KS-SPA	SPXY-SPA	KS-PLS	SPXY-PLS
Number of Variables	38	22	14	20
RMSEP	0.2491	0.2368	0.2071	0.1973

Table IV presents the GADC prediction results using double chromosome for variables and sample selection. Table shows parameters used by GADC like variables number, generations and population size and mutation rate. Each parameter set was tested 50 times and calculated its standard deviation.

TABLE IV.   PREDICTION RESULTS FOR GADC ALGORITHM FOR PROTEIN CONTENT IN THE WHEAT DATA SET

	GA (50x)			
	Conf. 1	Conf. 2	Conf. 3	Conf. 4
Average Number of variables	97	95	98	98
Average Number of samples in calibration set	324	312	296	306
Mutation rate	0.1	0.1	0.2	0.2
Next parent rate	0.1	0.2	0.1	0.2
Population size	50	50	100	100
Number of generation	100	100	100	100
Maximum RMSEP	0.0981	0.0992	0.0880	0.0957
Minimum RMSEP	0.0587	0.0689	0.0619	0.0616
RMSEP Sdv	0.0075	0.0071	0.0055	0.0075

Table IV shows that for different configurations the results are similar in RMSEP, number of variables and number of samples in calibration set. In this sense we choice to use the configuration 1 for comparison with classical algorithms. Table V shows a comparative of minimum RMSEP obtained by all algorithms. As can be seem, the prediction error using the model obtained by GADC was 70% better than the best classical algorithm (SPXY-PLS with 0.1973). Figure 1 shows the real protein content versus the predicted protein content. As can be seem, the predicted value is close to real values.

TABLE V.   COMPARATIVE OF MINIMUM RMSEP OBTAINED BY THE TESTED ALGORITHMS.

	RMSEP (minimum)
KS	2.8270
SPXY	1.4567
PNG-SPA	0.2066
KS-SPA	0.2018
SPXY-SPA	0.1957
PNG-PLS	0.1711
KS-PLS	0.2071
SPXY-PLS	0.1973
GADC	0.0587

VIII.   CONCLUSION

The aim of this paper was to establish an genetic algorithm with double chromosome structure (GADC) to compare to
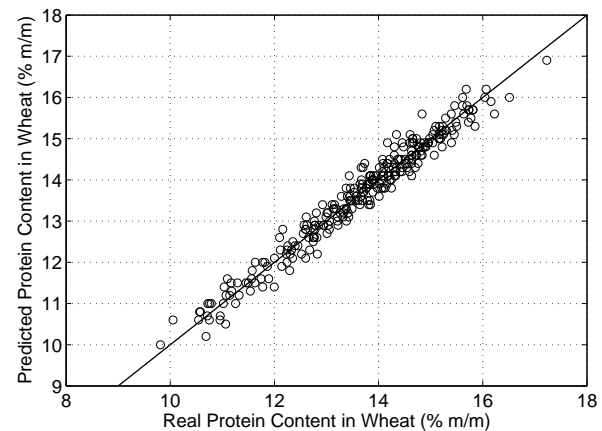
Fig. 1.   Comparison between the real value of protein content and predicted value using the model obtained by AGDC.

the results of the most popular techniques for sample and variable selection problem in multivariate calibration. The KS and SPXY algorithms presented results close to one another, although the SPXY have fared somewhat better. With variable selection algorithms the results the prediction error decrease. Our results show that the evolutionary algorithm with double chromosome leads to significantly better results compared to the others algorithms tested.

REFERENCES

[1] S. Wold, H. Martens, H. Wold. *The multivariate calibration problem in chemistry solved by the pls method*. In: Ruhe, A.; Kagstrom, B., editors, PROC. CONF. MATRIX PENCILS, p. 286-293, Heidelberg, 1983. Springer Verlag.

[2] P. J. Gemperline, J. R. Long, V. G. Gregoriou *Nonlinear multivariate calibration using principal components regression and artificial neural networks*. Anal. Chem., 1991, 63 (20), pp 23132323.

[3] R. W. Kennard, L. A. Stone. *Computer aided design of experiments*, Technometrics, 11:137, 1969.

[4] M. Daszykowski, B. Walczak, D. L. Massart. *Representaive Subset Selection*, Analytica Chymica Acta, V.468, 91-103, 200

[5] R. K. H. Galvo, et al. *A method for calibration and validation subset partitioning*, Talanta, 67:736, 2005.

[6] H. A. D Filho. *Desenvolvimento de tcnicas quimiomtricas de compresso de dados e de reduo de rudo instrumental aplicadas a leo diesel e madeira de eucalipto usando espectroscopia NIR*. Tese de Doutorado. UNICAMP. Campinas, Maro de 2007.

[7] J. M. Michael, L. C. Gerard, S. Clifford. *Variable Selection in Multivariate Calibration of a Spectroscopic Glucose Sensor*. Texas A & M University, Biomedical Engineering Program, 233 Zachry Engineering Center (M.J.McS., G.L.C.) and Department of Statistics (C.S.), College Station, Texas 77843. Volume 51, Number 10, 1997.

[8] M. Forina, et al. *Selection of useful predictors in multivariate calibration*. Anal. Bioanal. Chem. 380: 397, 2004.

[9] Y. V. Heyden, J. P. M. Andries, M. Goodarzi. *Variable Selection and Reduction in Multivariate Calibration and Modelling*. LC-GC Europe; Dec 2011, Vol. 24 Issue 12, p642.

[10]   P. S. Itnia, et al. *Multivariate Calibration by Variable Selection for Blends of Raw Soybean Oil/Biodiesel from Different Sources Using Fourier Transform Infrared Spectroscopy (FTIR) Spectra Data*. Energy and Fuels 2008, 22, 20792083.

[11]   C. M. Andersen, R. Bro. *Variable selection in regression - a tutorial*. J. Chemom. Special Issue Article, 2010 .

[12]   M. C. U. Arajo, et al. *The successive projections algorithm for variable selection in spectroscopic multicomponent analysis*. Chemometrics and Intelligent Laboratory Systems. 57: 65, 2001.

[13]   D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. New York: Addison-Wesley (1989).

[14]   R. K. H. Galvo, et al. *Multi-Core Computation in Chemometrics: Case Studies of Voltammetric and NIR Spectrometric Analyses*. J. Braz. Chem. Soc. , Vol. 21, No. 9, 1626-1634, 2010.

SESSION

FUZZY LOGIC AND APPLICATIONS + COGNITIVE SCIENCE AND SYSTEMS + QUANTUM LOGIC AND COMPUTING

Chair(s)

TBA

Microgenetic Critic for Situation Assessment Supporting Abduction and Surprise

Ronald L. Hartung and Kirk A. Weigand
[1] The Design Knowledge Company, Fairborn, Ohio, USA
[2]Air Force Research Laboratory, Wright-Patterson AFB, Ohio, USA

Abstract - *One of the core problems in cognitive systems is recognizing a situation and grounding that recognition to a sensed representation occurring in the world. This recognition also requires representations that can be analyzed and critiqued. This critic function challenges symbolic approaches, which become limited by the fuzziness of underlying meanings as contrasted to the crispness of symbols. Symbolic representations also lack the ability to translate sensor data to more abstract concepts while still preserving the underlying relationships. The proposed microgenetic abductive system is inspired by humans' adaptive solution to this representation problem through their cognitive evolutionary development. This paper describes a system and approach designed to advance abductive situation recognition by bridging form sub-symbolic input to concepts that allow critical analysis of surprising events.*

Keywords: abduction, cognitive system, sub-symbolic representation, world representation, situation recognition

1   Introduction

This work comes from a project to apply research from cognitive science, logic and process philosophy to solve problems in software security. In particular, abduction and surprise have been a major focus of the study. [1]. The ability to recognize surprising situations and to abductively address problem solving in complex domains is an attribute found among human experts. These human abilities are both elegant and flawed in that they usually work yet sometimes fail as well. Abductive inference's ability to find the best explanation of many possible explanations developed from a long evolutionary heritage that enabled the human species to survive. Human's ability to adapt appears to come from learning embedded in long and short term processes of growth: 1) Development of the human species (phylogenesis), 2) development of individual humans (ontogenesis) and 3) development of the mind over the life of an individual (microgenesis). [2][3][4]

On a pragmatic bent, this framework attempts to engineer a working artificial intelligence system that straddles the divide between symbolic, linguistic and semantic models in Artificial Intelligence (AI) and bottom-up artificial neural network and automata systems. However the focus of the paper is on the component to recognize situations and also

evaluate the situation in an effort to "detect when the situation is not what it seems;" that is, to encounter surprise and adapt in an efficacious manner that serves to benefit the agent's purpose with respect to the novel states encountered from the surprising situation.

2   Background

The proposed system is inspired by cognitive science but also results from an engineering approach. Inspiration from cognitive science stems from an ongoing debate between symbolic and connectionist schools of thought on the most productive scientific path for building cognitive models. Gopnik [5] and Elman [6] call for a systematic understanding of development in cognitive science to help resolve the Nature versus Nurture argument found at the roots of the connectionist versus symbolic debate. Gopnik calls for a bridging of the symbolic systems with connectionist systems using evidence from her studies of ontogenetic development. Microgenetic theory offers a top-down coherent way that humans may learn during moment-by-moment human thinking [2][3]. Gardenfors' theory of concept geometry offers a grounding for these desiderata [7].

3   Situation Recognition System

The situation recognition system is composed of three components, as shown in Figure 1.

Figure 1
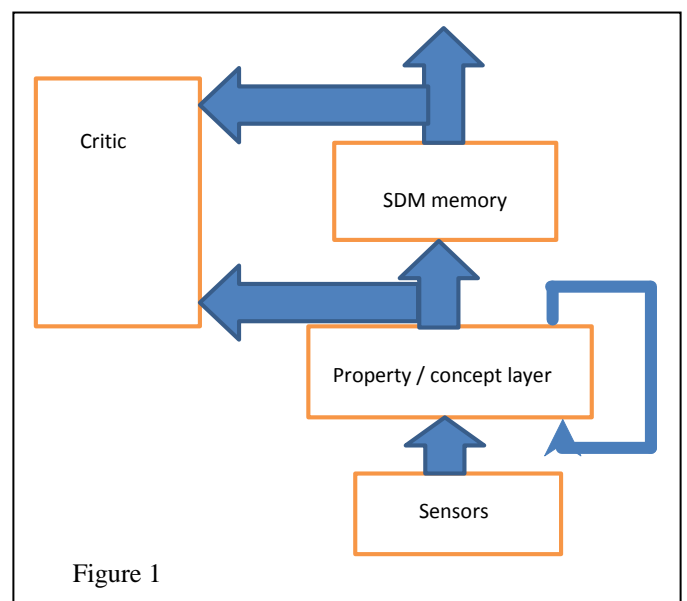
The property and concept layer is constructed using conceptual spaces and is described first. The SDM (sparse distributed memory) is described next and is used to match and recognize. The critic, still in its infancy, is described last.

3.1.1   Conceptual Spaces

The model describes a process that takes sensory input to concepts, based on Gårdenfors' conceptual spaces [7]. The terms defined in the model are summarized as follows. The sensory system has some simple properties. Input signals are treated as relative. The important feature required by Gårdenfors' model is that similarity and betweeness are persevered.   That is classical geometry, not precise measurement, is the essence of the output of the sensory system.

Sensors are grouped into related sets called domains. Gårdenfors' example of a domain is color, which allows the component sensor to measure hue, chromaticity and brightness as defined by the NCS color system [8]. These three dimensions produce a space which represents the domain.

Properties are partitions of a domain. For example, red is represented by a region of the color domain. A significant property of these regions is their convex shape. Convex shapes are easier to learn. Gårdenfors denotes these as natural properties. One possible method for the partitioning is Voronoi tessellation; there are other interesting possibilities as well such as Kohonen maps, and perhaps deep connectionist learning neural systems.

Properties can be combined into concepts by the same mechanism as properties are formed from domains. Concepts, unlike properties, are built by combining sensors from multiple domains, properties and other concepts (see figure 2).
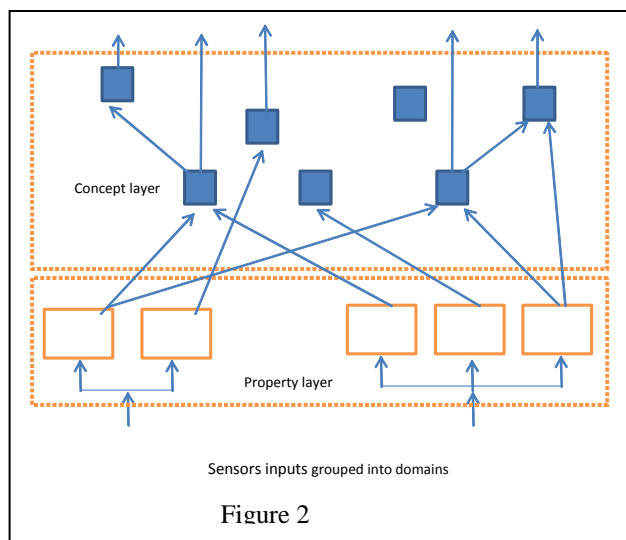
Figure 2

The model allows the construction of new domains. These are derived from the domains directly related to the senses. There are several methods used. Domains represent abstract

constructs similar to symbolic representations in AI and cognitive modeling and thus serve as the top-down anchor in this framework. Higher order domains can be constructed from a set of domains. A simple example is the longerThan relation constructed as a partition of the space constructed by two length domains. Clearly following this can produce considerably higher dimension spaces. For example, a space for representing faces or persons will have a large number of dimensions.

Another approach is to produce a new partition on an existing domain. That is, to take a subspace of the domain and to partition that domain into properties. The example used is skin color. White and red have different meanings when applied to skin color than when applied to some other context such as a ball. Here, similarity is used to construct a new subdomain.

Gårdenfors uses the example of Newtonian mass as an abstract domain; while it is similar to weight, it is not something directly measured by the sensory system. This shows an approach to systems being grounded in the world by sensory systems, yet also being able to construct abstract models by analogy to physical sensory systems. There is no direct sensory input into Newtonian mass, yet it can still be represented as a domain.

An interesting and useful aspect of this approach is the structured sub-conceptual level. This captures the fluidness of human concepts and presents an approach that can be used to support analogy and to guide induction [7]. The fluid aspects of concepts can be achieved in several ways. Sensitivity to context can be used to downgrade the importance of some of the sub-conceptual properties and sub-concepts that support a given concept. This can be used to select a new concept or to look at the comparison of two concepts in the current environment (defined by the sensory input and goals). This is the key to binding symbolic concepts to the world by a connectionist sub-symbolic structure of meaning.

The approach also enables a bi-directional processing technique – concept to properties and properties to concepts. This is largely the function of the critic. The processing arrows are omitted from the diagrams for simplicity. On initially encountering an input from the sensory system, the flow is from inputs to properties to concepts. However, analysis can also work back from concepts to properties and can do sensitivity analysis by looking at the strength of property and concept selectors. This provides a very meaningful fuzzy membership for properties and concepts.

The input to the system will come through a collection of sensors. The sensors are specific to the problem domain. For example in the reverse engineering domain, the sensors will prehend the aspects of code that a human must build from lower level primitives. This will short cut the construction of concepts and properties that take place in a human perception and cognition during reverse engineering. Care must be taken in the construction of the sensor system. Raising the sensor level too high, on a scale of abstraction, will reduce the ability to make distinctions that will be needed to form a flexible system. Biological inspiration is taken from the evolution of

the human sensory and cognitive system as tuned to survival on the savannas, and later developed the ability to solve more conceptual problems.

3.2   Sparse Distributed Memory

The next layer is a memory built on the sparse distributed memory (SDM) design from Pentti Kanerva [9]. This is extended with the work found in [10]. The Learning Intelligent Distributed Agent (LIDA) system has made extensive use of SDM for a variety of memory types [11]. The memory in this model needs to fulfill three functions: situation recognition, sequence storage and action. Only situation recognition is considered in this paper. Situation recognition follows Kanerva's original scheme of applying a large address space so that the memory easily extracts similar situations. The memory stores the address in the word being stored and that provides the auto associative memory.

The use of the SDM to detect similarity in situations depends on the relationship between the access radius and the locations that are similar to the input being considered. The details can be found in the reference [9]. If the concept is stored in the memory as a single bit, then the access radius will capture locations with enough common concepts. However, a more interesting approach is to store situations with a strength value for the concept. The issue is then to maximize the use of the access radius. The way to do this is to Huffman encode the strength so the distance in coding units and the strength indication is aligned.

While the SDM is very good at, associatively, selecting the best match, in this application the additional step needed is to consider alternatives based on marginal, weakly selected concepts.

3.3   The critic function – abduction's entry

The overarching project has been to look at abduction and sensemaking problem solving. This is discussed in [1]. The present level of development of the critic is still weak in abductive ability [12][13], but it is a step toward stronger abduction and also a step toward expert sense making. The sense making problem is often a straight forward recognition of a situation that has been encountered before. The challenge is recognizing when the situation is almost the one seen before, but misses some critical aspect.

While the system is designed to recognize situations and direct actions by setting goals, the critic has an oversight function with the ability to interfere. It has a focus on surprise and abduction. This is implemented in two modes – by exploration of alternatives and by reflection during backtracking on failure. As a practical note, there are two caveats on the current state of the critic. The description provided here is our current view and is still under development to strengthen the critic. Also, as we apply this system to different problems, the role of the critic can be restricted. Some domains demand a critic with restricted actions in order to assure the system behaves in a controlled fashion. In this case the critic can log its observations for latter study. This mode is useful in new domains.

Surprise can occur is two ways. There is the obvious mode when a goal, selected by the system, fails to be obtained. Such a failure triggers backtracking. The other is the preemptive detection of "not quite right" situations, which is covered first.

The SDM is used to find situations which match the current set of concepts and properties encountered in the environment. This is done by associative matching in the SDM. In order to consider detection of "not quite situations," the agent must consider how these situations can arise. There appear to be two main cases that are tractable. The first examines the situational assessment for marginal values. The strength of Gårdenfors' system is its geometrical formulation, using relative values, not absolute numerical values. Thus in classifying properties and concepts, we can use the geometrical notions to mark properties that are close to classifying as a different property. This marginal boundary analysis can be used to look at other possible goals in contrast to the initial situation. Since this can be a costly activity, it is triggered by an exanimation of the "strength of evidence." This is done by considering first the marginal properties and concepts in the situation and the concepts or properties that are in the found situation, but not in the input (missing evidence). If nothing appears to be marginal, the analysis is not triggered. Secondly, during the actual situation, the critic analyzes the concepts and properties expressed by the input to the sensors. Here, the system looks at concepts and properties not included in the situation produced from the SDM. Alternatives are generated by suppressing some of the concepts in the found situation and probing the memory for other possible situations. This mode of analysis is inspired by process philosophy's explication of consciousness as amplified by inclusion of counterfactuals (D.R. Griffin, 2009).

On failure of a goal, the system backtracks and performs the same kinds of analysis describe above. However, the failed goal provides additional information, that of falsifying that choice of the goal as a counterfactual representation. This forces reconsideration of that goal and to a lesser degree the prior goals that supported the choice. The "blame" for the bad choice is passed back and is applied with a reduced degree of weight as the backtracking proceeds backward.

Now turning to abduction, Hoffmann uses two dimensions to construct a 3 by 5 matrix of abductive types. The simpler forms of abduction are implemented implicitly by the matching in the SDM. Since we are looking for the closest match to a situation, evidence for the situation that is missing will be assumed. The more interesting case is how to apply shifts in the structure of a representation. This is made possible by a meta-information level over Gårdenfors' system. By producing a meta-level over the domains, one can rank similarity measures between domains inductively. This similarity can then be used to shift concepts. Analogical reasoning can be used to modify a situation by substituting concept A for a concept B, but only when their underlying domains have relative similarity.

4 Conclusions

Our work is an intriguing approach to situation recognition. The current state of the system does not yet apply learning to the problem space. This is clearly the next step. Gårdenfors' work does address learning and this is an additional function to add to the critic. In addition the application of abduction in this work falls in the lower reaches of Hoffmann's taxonomy. Improvement will require extension into three reaches. First of all, the use of tool manipulation [Magani 2009] is heavily involved in the more advanced abductive processes. This allows us to bring in logic, mathematics and external tools like software systems. Second, the similarity and metaphor provided by conceptual geometry can be used for theoric shifts. Third, elicited expert intuition can bootstrap microgenetic development of the concept geometry.

5 Acknowledgement

6 References

[1] Ronald Hartung, Kirk Weigand "Abduction's Role in Reverse Engineering Software", NAECON, 2012

[2] J.W. Brown, "Microgenetics theory: Reflections and Prospects," Neuropsychoanalysis, 3: 61-74, 2001

[3] J.W. Brown, "What is Consciousness?" Process Studies, 41:1 21-41, Spring/Summer, 2012

[4] M. Pachalska & B. D. MacQueen,"The Microgenetic Revolution in Contemporary Neuropsychology and Neruolinguistics," in Process Approaches to Consciousness in Psychology, Neuroscience and Philosophy, Editors, M. Weber & A. Weekes, SUNY Press, New York, 2009

[5]Aliason Gopnik, http://thesciencenetwork.org/programs/cogsci-2010/alison-gopnik, 2012

[6] Jeff Elman, CogSci http://thesciencenetwork.org/programs/cogsci-2010/10-years-of-rumelhart-prizes-a-symposium, 2010

[7] Peter Gårdenfors, "Conceptual Spaces, the geometry of thought", A Bradford book, MIT Press, Cambridge Mass, 2000

[8] Colin Ware, " Information Visualization", Elsevier, p. 121, ISBN 9780123814654, 2012

[9] Pentti Kanerva, "Sparce Distributed Memory", A Bradford Book, MIT Press, Cambridge Mass,1998

[10]Javier Snaider, Stan Franklin,"Extended Sparse Distributed Memory and Sequence Storage" Cognitive Computation, 4(2), 172-180. doi: 10.1007/s12559-012-9125-8, 2012

[11] U. Faghihi, & Stan Franklin, "The LIDA Model as a Foundational Architecture for AGI" In P. Wang & B. Goertzel (Eds.), Theoretical Foundations of Artificial General Intelligence (pp. 105-123). Paris: Atlantis Press, 2012

[12] Hoffmann, Michael, H. G. "Theoric Transformations" and a New Classification of Abductive Inferences, Transctions of the Peirce Society, Vol 46 No. 4 pg 570 – 590 2011

[13] L. Magnani, "Abductive Cognition The Epistemological and Eco-Cognitive Dimensions of Hypothetical Reasoning" Springer Verlog ISBN 978-3-642-03630-9, 2009

[14] D.R. Griffin, "Consciousness as Subjective Form: Whitehead's Nonreductionist Naturalism" in Process Approaches to Consciousness in Psychology, Neuroscience and Philosophy, Editors, M. Weber & A. Weekes, SUNY Press, New York, 2009.

Cognitive Control of Artificial Life Forms Utilizing Artificial Procedural Memories

James A. Crowder, John N. Carbone, Roderic Paulk

Raytheon Intelligence, Information, and Services

16800 E. Centretech Parkway, Aurora, Colorado 80011

Abstract - *Memories involve the acquisition, categorization, classification, and storage of information. The purpose of memory is to provide the ability to recall information and knowledge as well as events. Through our conceptual recollection of the past we are able to communicate with other people and understand events based on past experiences. The purpose of this paper is to describe a new cognitive architecture for artificial intelligence controlled devices that incorporates artificial procedural memory creation and recall. This cognitive architecture provides a scalable framework for episodic memory creation as an entity experiences events, and, over time, develops procedural memory "scripts" that allow repetition of tasks it has "learned" to accomplish, while effectively managing it's internal knowledge economy. The system will utilize a temporal-calculus driven spatial map to store spatio-temporal information, as well as the procedural memories. This provides the system with the cognitive abilities to approach task selection and reason via both experiential and spatial learning.*

In order to test and evaluate the viability of artificial memory creation and retrieval, a series of artificially intelligent, cybernetic life forms (cyber insects) were created and tested utilizing a combination of analog and digital neural frameworks allowing procedural memory creation, storage, and retrieval as the methodology for autonomous control of the artificial life form. Presented here are initial design and discussion of artificial analog and digital neural structures along with a discussion of the use of artificial procedural memories, used as autonomous cognitive control of the artificial life form. Lastly, future research needs are proposed for improving the continuous balance of knowledge storage and knowledge necessity relative to prime directives and learned objectives in Self Evolving Life Forms (SELF).

Keywords: Artificial Life Form, Procedural Memory, Neural Networks.

1. Introduction

To examine plausibility and feasibility of self-evolving life forms, we have undertaken a series of experiments over the past 10 years to develop and test small, artificially intelligent, cybernetic entities with varying abilities to think, learn, and self-evolve, at low-brain functional levels. These artificial life forms were created to learn and act like insects, with rudimentary cognitive functions to establish whether artificial cognitive architectures are realizable at a most simplistic level. The current 2nd instantiation, of these cybernetic insects, which we have named 'Zeus,' after the Greek god, whose name means "living one," utilizes a simplistic analog neural network for information transfer throughout his internal effector network, and utilizes low-level digital cognitive framework that affects learning and self-evolving. Controllers were utilized containing EEPROM, RAM, and Flash memory in order to facilitate learning and storage of learned behavior in as low a Size, Weight, and Power (SWaP) footprint as possible[1]. Basic effector control commands are stored in EEPROM. As Zeus learns, information is stored in RAM until determining that a behavior has been 'adequately' learned, and then stored as a series of commands (procedural memory) into flash memory.

2. Analog Neural Structures

For his analog neural network, we utilize an adaptation of the information continuum equation [Crowder 2010a], which is shown below:

---

[1] AVR ATTINY24 and ATTINY44 Microcontrollers are used, with the ATTINY 24 as the baseline.

$$C \frac{du(x,y,t)}{dt} = -\frac{1}{R}u(x,y,t) + \int\int_{x\;y} w(x,y)z(x,y,t)dxdy + I(x,y,t)$$

where:

u represents the unit node of the system,
x represents the preprocessed input to node u,
y represents the output from node u,
w represents the relative contextual information threads and association weight of *u* with its surrounding nodes, including a decay factor for each relative information thread that describes the relative contextual decay over time, where:

$$w = \sum_{j=1}^{M} \frac{1}{r_j} T_j KD_j W_j$$

   where:
   T represents the Contextual Information Thread j derived from Fuzzy, Self-Organizing Contextual Topical Maps
   KD represents Knowledge Density j of Information Thread T
   W represents Weighting for Contextual Thread j, and

$$\sum_{j} W_j = 1$$

   I represents the processing activity for node *u*,
   z represents the learning functionality for node *u*,
   1/R represents the decay rate for node u^2, and
   C represents the capacity of node *u*.

The adaptation, shown below [Hopfield 1984], describes the dynamic equation for Zeus' analog neurons [Hutchinson, et. al. 1988].

$$C_i \dot{u}_i(t') = -\frac{1}{R_i}u_i(t') + \sum_{j=1}^{N} T'_{ij} f_j\left(u_j\left(t' - \tau'_j\right)\right) \; i = 1,\ldots,N$$

Where:
   C_i is the ith neuron analog input capacitance
   R_i s the resistance to the rest of the analog neural network at the input to neuron *i*:

---

^{2} In this case, the decay represents the information's relative value over time.

$$\left(\sum_{j}\left[T'_{ij}\right]\right)^{-1}$$

f_i is the transfer function of the ith neuron

Within Zeus' analog neural network, the digital cognitive system monitors the strength of the analog neurons to determine when the strength of learning has progressed to the point where the learning should be "committed to memory" within the digital cognitive system (RAM). Polyn and Kahana [Polyn and Kahana, 2008] suggest that recall of known item representations is driven by an internally maintained context representation. They described how neural investigations had shown that the recall of an item represented in the mind is driven by an internally maintained context representation that was previously integrated information with a time scale. Therefore, when a series of analog neurons is sufficiently strengthened over time, and have been committed to digital memory, such that they create a series of commands or learned behaviors that can be considered a "procedural memory," these are stored in Flash Memory with a tag that corresponds to the learned activity or behavior (e.g., turn left). The next time Zeus' sensors relate information such that he needs to move left, this procedure is recalled from memory and activated; meaning, he doesn't have to think about how to turn left, he turns left automatically. This is analagous to instinctive driving of a car after we have learned to drive, however, at a much lower cognitive level than a human but enough to allow Zeus to learn and evolve.

3. Self-Evolution utilizing Procedural Memories

With Zeus, the goal was to add cognitive skills one at a time, perform tests and determine whether he was able to integrate these together within his limited cognitive framework. Once Zeus has reached a significant cognitive skill level, a new cybernetic "bug" artificial life form will be implanted with Zeus' cognitive skills;

present at the beginning of activation. We will determine whether the new artificial life form has an easier or more difficult time integrating the cognitive skills together, and whether the skill sets developed one at a time and at the same cognitive level as Zeus. We expect to also gain valuable insight into artificial life form initialization and sequencing.

Zeus was first initialized into existence in early September, 2012. He learned to walk, turn, integrate his sensors, plan his movements, and execute his plans, hence, demonstrating autonomous planning, sensory integration, and autonomous decision making, none of which is specifically part of his initial programming. He was only enabled with the skills to learn, think, store, and recall memories, provided to him initially. He now carries 25 different procedural memories in the form of a series of commands learned for a particular action.

3.1 Test Scenarios

Zeus has objective functions he endeavors to drive to zero that are part of his basic "instincts." He must learn to use his available sensors and effectors to reduce these objective functions. The set of learning tests that have been and will be performed on Zeus to test his analog and digital neural networks and cognitive algorithms are:

1. **Learning to walk**: In his initial state, Zeus understands the concept of movement, but does not have the knowledge of how to walk. He must learn to move his effectors in order to move effectively. First, initially learning to walk using his six legs, and then to turn left and right.
2. **Learning to find darkness**: One of Zeus' objective functions is "fear of light." He must learn to use his light sensors and compute the differential between the two sensors to establish the direction of movement required to lower the objective function.
3. **Power Regeneration**: Another objective function to be added is the notion of a

"hunger instinct," which to Zeus means low power. He carries solar cells to charge up, or feed. He must learn to balance the objective functions for hunger (find light and charge) vs. the objective function for fear of light (find darkness).

4. **Proximity Sensing**: Zeus carries infrared transmitter/receivers to sense when he is close to another object. One of his objective functions is to avoid closeness to other objects (receiving reflected infrared raises the objective function). He must balance 1) the need to find darkness, which may be found under other objects, 2) the need to not get too close to objects, and 3) the need to get close to a light source to "feed."

Understanding how Zeus manages and balances these objective functions and how procedural memories are created, stored, and recalled could provide valuable insight into cognitive control of autonomous life forms for use in autonomous deep sea, space, or land-based applications.

4. Procedural Implicit Memory

Procedural Implicit Memories allow previously learned tasks to be performed without specific "conscious" memory recall/reconstruction of how to perform the task [Crowder, Raskin, and Taylor 2012]. Procedural memories tend to be inflexible, in that they are tied to the task being performed. For example, when we decide to ride a bike, we don't unconsciously recall/reconstruct memories of how to drive a car, we recall/reconstruct unconscious Procedural Memories of how to ride a bike. In Zeus, tasks that are learned and our deemed 'important' to capture for future use will have Procedural Memories stored as steps, or "procedures" that are required to perform the same task in the future. In his work on Procedural Memory and contextual Representation, Kahana showed that retrieval of implicit procedural memories is a cue-dependent process that contains both semantic and temporal components [Kahana, Howard, and Polyn 2008]. Creation of

Procedural Memories is tied not only to task repetition but also to the richness of the semantic association structure [Landauer and Dumas 1997].

Earlier work by Crowder, built on Landauer's Procedural Memory computational models and Griffith's topical models [Griffiths and Steyvers 2003], theorized about the creation of artificial cognitive procedural memory models based on Knowledge Relativity Threads [Carbone, 2010] to create the semantic associations [Crowder and Carbone 2011a] and work in Fuzzy, Self-Organizing, Semantic Topical Maps [Crowder 2010 & 2011] counted on the topical model needed to create long-term procedural memories. These Knowledge Relativity Thread based models were derived from combining cognitive psychological, space-time, strong, weak, and quantum mechanical, concepts, along with Topical Maps which are based on early work by Zadeh. Zadeh [Zadeh 2004], which described tacit knowledge as worldly knowledge that humans retain from experiences and education, and concluded that current search engines, with their remarkable capabilities, did not have the capability of deduction, that is the capability to synthesize answers from bodies of information which reside in various parts of a knowledge base. More specifically, Zadeh describes fuzzy logic as a formalization of human capabilities: the capability to converse, reason and make rational decisions in an environment of imprecision, uncertainty, and incompleteness of information. In their work in cognition frameworks, Crowder and Carbone [Crowder and Carbone 2011b & 2012a] also expand on the work of Tanik [Ertas and Tanik 2000] in describing artificial procedural memories as procedural knowledge gained through cognitive insights, based on fuzzy correlations.

5. Creation and Retrieval of Artificial Procedural Memories

Continued investigation, utilizing the work of Kahana [Kahana, Howard, and Polyn 2008] in associative episodic memories [Crowder 2001], led to the development of a cognitive perceptron framework for creating, storing, and retrieving artificial implicit memories [Crowder and Friess 2010 & Crowder, Raskin, and Taylor 2012] (see Figure 5.7). Based upon this work, a systems and software architecture specification was developed for an artificial cognitive framework utilizing the cognitive perceptrons [Crowder, Scally and Bonato 2012].

The main hypothesis here is that the procedural memory scripts can be detected and acquired with the combination of rule-based computational semantic techniques enhancing the artificial life form's understanding of repeatable and useful procedures. The objectives of utilizing artificial procedural memories for cognitive control of artificial life forms is to, 1) identify potential procedural memories utilizing a combination of rule-based techniques, combined with machine learning techniques, and 2) to develop the principles of comparison and comprehension of commands required for creation of procedural memories (see Figure 1).

Crowder, in conjunction with Carbone and Friess, in researching artificial neural memory frameworks that mimic human memories, are creating computer architectures that can take advantage of Raskin and Taylor's Ontological Semantic Technology [Raskin, Taylor, and Hemplemann 2010 & Taylor and Raskin 2011] and create an artificial procedural memory system that has human reasoning capabilities and mimics the fuzzy and uncertain nature of human cognitive processes. This new focus for Crowder [Crowder, Raskin, and Taylor 2012] is to create processes necessary for the creation, storage, retrieval, and modification of artificial procedural memories (see Figure 2).

6. Conclusions

Testing on Zeus has shown that self-evolution through creation, storage, and retrieval of artificial procedural memories can provide an

effective and efficient mechanism for autonomous control of artificial life forms. There is a significant amount of research, development and testing required in general, and specifically over the next year, including expansion of Zeus' neural framework to include a more comprehensive Possiblistic, Abductive Neural Network to allow hypothesis generation enhancing the speed and the quality of his cognitive skillset; results will be published in the near future. Our continuous research in this area in developing higher fidelity cognitive functions and exploring new types of cognitive testing are important to establish the breadth of self-possibilities. We will continue to explore not only these low level brain functions, but also much higher brain-functions as well.
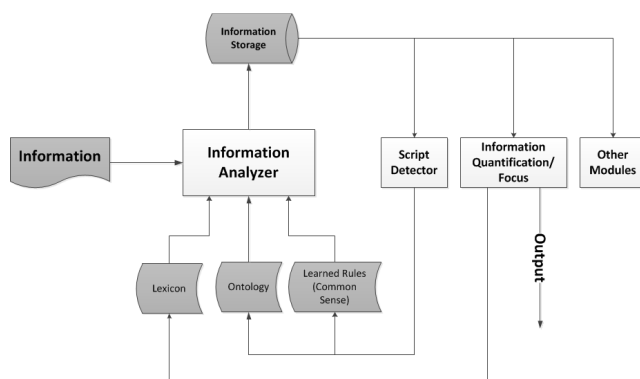
Figure 1 – Artificial Procedural Memory Creation
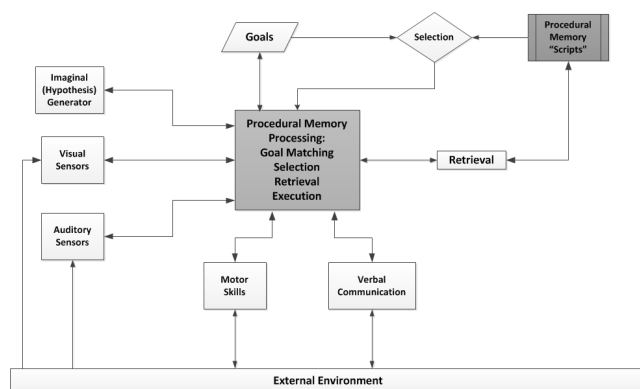
Figure 2 - Artificial Procedural Memory Retrieval

One question to be answered for comprehensive autonomous systems is, "How much initial information, or memories, should be provided?"

Based upon our initial results, we speculate confidently that since learning is stochastic in nature and depends on current understandings, the more initial information or memories (Cognitive Ontology concepts) an entity begins with, the more it will influence the artificial life form's learning direction and its ability to "intuit" about its environment. Consequently, it becomes imperative to manage content quantity and quality efficiently within systems of widely varying resource constraints. Therefore, future efforts will include some research into synergistic algorithms within the Activity Based Intelligence (ABI) domain. ABI employs normalization methods of activity patterns (AP), which show possibilities for improving balance between knowledge storage, necessity, prime directives, and learned objectives. Additionally, research in improving condensation of decision quality content via anomaly detection in Big Data environments shows applicability to smart aggregation of knowledge in constrained environments.

Lastly, we have been exploring the question of how to test cognitive systems. Given that these systems will learn, think, reason, and self-evolve, we believe that standard system testing is inadequate to understand whether a system is "working properly." Hence, we believe it is imperative to understand self-evolving learning processes by testing them in a context of "cognitive" and "psychological" paradigms. Therefore, to obtain a comprehensive understanding of proper system functionality we believe it is necessary to test analogously to determining whether a human is "functioning normally." Finally, we believe this leads us to a field of study envisioned long ago, "Artificial Psychology [Crowder and Friess 2012]."

7. References

1. Crowder, J. 2001. Integrating an Expert System into a Neural Network with Genetic Programming for Process Planning. NSA Technical Paper TIT_01_01_013_2001_001.

2. Crowder, J. 2010. Operative Information Software Agents (OISA) for Intelligence Processing. Proceedings of the 12th annual International Conference on Artificial Intelligence, Las Vegas, NV.

3. Crowder, J. 2011. Cognitive architectures for real-time integrated system health management. Proceedings of the 8th Annual Workshop of Structural Health Monitoring, Stanford University

4. Crowder, J. and Carbone, J. 2011a. Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. Proceedings of the 13th annual International Conference on Artificial Intelligence, Las Vegas, NV

5. Crowder, J. and Carbone, J. 2011b. Hybrid Neural Architectures for the ELYSE Cognitive System. Journal of Aerospace Computing, Information, and Computing 10.

6. Crowder, J. and Carbone, J. 2012. Reasoning Frameworks for Autonomous Systems. Proceedings of the AIAA Infotech@Aerospace 2012 Conference, Garden Grove, CA.

7. Crowder, J. and Friess, S. 2010. Artificial Neural Diagnostics and Prognostics: Self-Soothing in Cognitive Systems. Proceedings of the 12th annual International Conference on Artificial Intelligence, Las Vegas, NV.

8. Crowder, J. and Friess, S. 2012. Artificial Psychology: The Psychology of AI. Proceedings of the 3rd Annual International Multi-Conference on Complexity, Informatics and Cybernetics, Orlando, FL.

9. Crowder, J., Raskin, V., and Taylor, J. 2012. Autonomous Creation and Detection of Procedural Memory Scripts. Proceedings of the 13th Annual International Conference on Artificial Intelligence, Las Vegas, NV.

10. Carbone, J. 2010, A Framework for Enhancing Transdisciplinary Research Knowledge, Texas Tech University

11. Ertas, A. and Tanik, M. 2000. Transdisciplinary Engineering Education and Research Model. Transactions of the SDPS, Vol. 4, No. 4, pp. 1-11.

12. Griffiths, T. and Steyvers, M. 2003. Prediction and Semantic Association. In: Halmberg KJ and Steyvers M. (eds.) Advances in Neural Information Processing Systems, 15, pp. 11-18.

13. Hopfield, J. 1984. Neurons with Graded Response Have Collective Computational Properties like Those of Two-State Neurons. In Modeling and Analysis in Biomedicine, ed. Nicolini, C., World Scientific Publishing, New York, NY.

14. Kahana, M., Howard, M., and Polyn, S. 2008. Associative retrieval processes in episodic memory. In H. L. Roediger III (Ed.), Cognitive psychology of memory. Vol. 2 of Learning and memory: A comprehensive reference, 4 vols. (J. Byrne, Editor). Oxford: Elsevier.

15. Landauer, T. and Dumas, S. 1997. Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge. Psychol. Rev. 194:211-240.

16. Hutchinson, J., Koch, C., Luo, J., and Mead, C. 1988. Computing Motion using Analog and Binary Resistive Networks. CWpirtcr, Vol. 21:3, pp. 52-63.

17. Kosko, G., "Fuzzy Cognitive Maps," *International Journal of Man-Machine Studies*, 24: 65-75.

18. Raskin, V., Taylor, J. M., & Hemplemann, C. F. 2010. Ausocial engineering. New Security Paradigms Workshop, Concord, MA.

19. Taylor, J. M., & Raskin, V. 2010. Fuzzy ontology for natural language. 29th International Conference of the North American Fuzzy Information Processing Society, Toronto, Ontario, Canada.

20. Zadeh, L. 2004. A note of Web Intelligence, World Knowledge and Fuzzy Logic. Data and Knowledge Engineering, vol. 50, pp. 291-304.

A New Method for Prediction of School Dropout Risk Group Using Neural Network Fuzzy ARTMAP

Valquíria R. C. Martinho[1]**, Clodoaldo Nunes**[2]**, Carlos R. Minussi**[3]
[1]Electro-Electronic Department, Federal Institute of Mato Grosso - IFMT, Cuiabá, MT, Brazil
[2]Infortmatcs Department, Federal Institute of Mato Grosso - IFMT, Cuiabá, MT, Brazil
[3]Electrical Engineering Department, University Estadual Paulista - UNESP, Ilha Solteira, SP, Brazil.

Abstract - *Dropping out of school is one of the most complex and crucial problems in education, causing social, economic, political, academic and financial losses. In order to contribute to solve the situation, this paper presents the potentials of an intelligent, robust and innovative system, developed for the prediction of risk groups of student dropout, using a Fuzzy-ARTMAP Neural Network, one of the techniques of artificial intelligence, with possibility of continued learning. This study was conducted under the Federal Institute of Education, Science and Technology of Mato Grosso, with students of the Colleges of Technology in Automation and Industrial Control, Control Works, Internet Systems, Computer Networks and Executive Secretary. The results showed that the proposed system is satisfactory, with global accuracy superior to 76% and significant degree of reliability, making possible the early identification, even in the first term of the course, the group of students likely to drop out.*

Keywords: higher education, school dropout, prediction dropout, artificial neural networks (ANN), Fuzzy ARTMAP neural network.

1 Introduction

Historically, school dropout is one of the most complex and crucial problems in education, causing social, economic, political, academic and financial damage to all the people involved in the educational process, from the students to the governmental and promotional agencies that long for efficient strategies to reduce the indexes of school dropout, since the measures adopted up to now did not have the desired effect.

In relation to higher education, school dropout is an international problem. Although its indexes show considerable variations among different nations, they show that in fact school dropout is present and strikes more and more a higher number of higher educational institutes (HEI) worldwide.

It is worth mentioning the United States - USA, with a dropout rate in colleges and universities of around 40%, representing a decline in the index of students graduated in higher education [1]. Conversely, China and India empower higher education, increasing the conclusion index [2]. Between these extremes lies Brazil, presenting a mean dropout index of about 20% [3].

Even taking into account all the differences and specificities of the (HEI) of different nations, the difficult task of solving the evasion problem is still common ground between them [4].

From this perspective, prevention and intervention programs are developed and structured taking into account the results of researches that identify the possible causes that generate the phenomenon of evasion. However, such measures could be more fruitful if there was prior knowledge of the students prone to evasion. And, for this, the development of methods, instruments or systems capable of previously making this identification is necessary.

To meet this need an intelligent, ambitious and innovative system was developed, for the prediction of risk groups of student dropout in presential higher education courses [5], using artificial intelligence techniques, the Fuzzy ARTMAP Neural Network [6-8]. This network has a structure in which the training is carried out in a supervised and self-organized way, with the possibility of continued learning [6].

This paper aims at presenting and making the developed intelligent system available as a possibility of identifying, in a proactive, continued and accurately the students of the traditional presential education, prone to evasion in higher education. And also to disseminate their fruitful results that contributed to the development of prevention and intervention programs, in order to improve retention of those students identified in the institution [5].

Further, this paper is structured as follows: Section 2 – the publications considered more relevant to this proposal. Section 3 - the ART and Fuzzy ARTMAP networks and their training algorithms. Section 4 - the study area, the construction of database and the development of the intelligent system in question. Section 5 - the implementation of the system. Section 6 - the results and analyses of the tests performed. Section 7 - the most relevant conclusions of the experiment.

2   Brief review

Most of the works analyzed, considering the causes of evasion in higher education, are critical and theoretical productions and present as results the factors that most influenced school evasion in the Brazilian higher education, its consequences and possible ways to overcome them [9]. The educational situation in Brazil is perceived through statistical data of private and governmental schools and research by governmental organs [10]. In a more global view, besides theoretical models with explanatory schemes about the causes of evasion [11], there is a comparative study among countries with poor education and with an insignificant evasion index [12] as well as a study with students that constantly shift from one school to another, showing similar causes for the evasion to occur [13].

In relation to the analysis and prediction of school evasion using intelligent system, productions where Data Mining techniques were used [14-16] and Artificial Neural Networks [17-20], e.g., Multi-layers Perceptron Network (MLP). In all of them, the results obtained were: the identification of causal descriptive patterns that lead the students to leave school and the verification of the efficiency and performance of the techniques used. The variables that can influence the students' evasion were also investigated in distance-learning courses.

This study [21] investigates group risk of students prone to evasion in e-learning courses, uses the combination of three machine learning techniques, including, Fuzzy ARTMAP neural network. This one is the most significant for the elaboration and development presented here.

3   ART and Fuzzy ARTMAP neural networks

The Artificial Neural Networks (ANN) [19] are computational tools that emulate the human brain and learn with the experience, trying to model and simulate its learning process, organizing its neurons in such a way that they will be capable of processing the information.

A typical neural network consists of several neurons, arranged in adjoining layers, connected by synapses (communication channels) associated to certain weights attributed to connections among the neurons, where all the knowledge of a ANN is stored. They always have an input and an output layer, with the possibility of having between them a varied number of layers, called hidden or intermediate layers.

The use of neural networks offers some benefits and capabilities that in synthesis are: non-linearity, input-output mapping, adaptability ("stability-plasticity" dilemma) [6], response to evidence, contextual information, fault tolerance, uniformity of analysis, neurobiological analogy.

The ART network systems are able to solve the "stability-plasticity" dilemma. They are plastic because they are able to learn to adapt to a changing environment and, at the same time, preserve their previously learned knowledge while maintaining their ability to learn new patterns, therefore they are stable.

In the Fuzzy ARTMAP model, two ART modules are interlinked through an inter-ART module, called Field Map. This module has a self-regulatory mechanism called match tracking that seeks for "marriages" or combinations among the categories of ART_a and ART_b modules, aiming to increase the generalization level and reduce the network error [6].

The Fuzzy ART neural network uses the theory of the fuzzy sets [22], employing the minimum operator ($\wedge$) AND Fuzzy, enabling the treatment of patterns of binary and analogical input, in an interval [0, 1], and increasing the generalization ability of the network.

3.1   Algorithm of an Fuzzy ART neural network

The algorithm of an Fuzzy ART neural network consists, essentially in the sequence described below [22]:

3.1.1   Normalizing of input data

The input data are represented by the vector $a = [a_1\ a_2\ a_3\ \ldots\ a_M]$. Normalizing of this vector should be according to the (1) [22]:

$$\bar{a} = \frac{a}{|a|}\quad,\quad \text{where}:\qquad |a| = \sum_i^M a_i \qquad (1)$$

where:

$\bar{a}$: normalized input vector;

$|a|$: norm of the input vector a;

a_i : element of he input vector a with index i.

3.1.2   Coding of the input vector

The complement coding is performed according to (2), to preserve the scope of information [22]:

$$\bar{a}_i^{\,c} = 1 - \bar{a}_i \qquad (2)$$

where:

$\bar{a}_i^{\,c}$: complementary element of the element of the normalized input vector;

$\bar{a}^{\,c}$: complementary vector of the normalized input.

Thus, the network input vector will be vector I, presented in (3) [22] .

$$I = \begin{bmatrix} \bar{a} & \bar{a}^{\,c} \end{bmatrix} \qquad (3)$$

$$I = \begin{bmatrix} \bar{a}_1 & \bar{a}_2 \ldots \bar{a}_M & \bar{a}_1^{\,c} & \bar{a}_2^{\,c} & \bar{a}_M^{\,c} \end{bmatrix}$$

3.1.3   Activity vector

The activity vector of the recognition layer F_2 is indicated by $y = [\ y_1\quad y_2\quad y_3\ \ldots\ y_N\ ]$, being N the number of categories created in F_2. Thus, one has [22]:

$$y_j = \begin{cases} 1, & \text{if the neuron } j \text{ of } F_2 \text{ is active, if } j = J \\ \\ 0, & \text{otherwise, if } j \neq J \end{cases}$$

3.1.4 Parameters of the Fuzzy ART neural network

Three parameters are essential in the processing of the Fuzzy ART network, they are [22]:

- Choice parameters (α) : $\alpha > 0$;
- Training parameters (β) : $\beta \in [0, 1]$;
- Vigilance parameters (ρ) : $\rho \in [0, 1]$.

3.1.5 Initialization of the weights

Initially, all the weights have values equal to 1, as in (4) [22]:

$$w_{j1}(0) = w_{j2}(0) = \ldots = w_{NM}(0) = 1 \qquad (4)$$

3.1.6 Choice of a category

Considering an input vector I in F_1, the choice of the category j in F_2 attends to the choice function T_j defined in (5) [22]:

$$T_j(I) = \frac{|I \wedge w_j|}{\alpha + |w_j|} \qquad (5)$$

where:

$\wedge$: operator AND-Fuzzy, defined by (6):

$$(I \wedge w)_i = \min(I_i, w_i) \qquad (6)$$

The system chooses the category that corresponds to the active J neuron, according to (7) [22]:

$$J = \arg \max_{j=1,\ldots,N} T_j \qquad (7)$$

If more than one neuron with maximum activation exists, the chosen category will be the one which has the smallest index j.

3.1.7 Resonance or Reset

The resonance occurs if the vigilance criterion, (8), is met.

$$\frac{|I \wedge w_j|}{|I|} \geq \rho \qquad (8)$$

If the vigilance criterion, (8), is not met, the reset occurs. In the reset, the J neuron of the F_2 is excluded of the searching process. So, a new category is chosen through the application of the (7) so that the resonance process will be performed. This procedure is performed until the network finds a category that fulfills (8).

3.1.8 Learning (Updating the weights)

After the input vector I has completed the resonance state, the training and learning process occurs and, consequently modifying the weight vector, given by (9) [22].

$$w_J^{\text{new}} = \beta(I \wedge w_J^{\text{old}}) + (1 - \beta) w_J^{\text{old}} \qquad (9)$$

where:
J : active category;
w_J^{new} : updated weight vector;
w_J^{old} : weight vector regarding the previous updating.

If the training parameter $\beta = 1$, there is the fast training.

3.2 Algorithm of an Fuzzy ARTMAP neural network

The algorithm for the processing of the Fuzzy ARTMAP Neural Network [7], occurs in the following way:

3.2.1 Input data

The input vectors of the Fuzzy ARTMAP network are represented by:

- $a = \begin{bmatrix} a_1 & a_2 & \ldots & a_P \end{bmatrix}$: ART$_a$ input, data sampling;

- $b = \begin{bmatrix} b_1 & b_2 & \ldots & b_P \end{bmatrix}$: ART$_b$ input, desired output;

where:
P : the number of subvectors of the a and b vectors.

3.2.2 Weight matrices

The weight matrices associated to the ART$_a$ (matrix w^a) and ART$_b$ (matrix w^b) modules, as well as to the Inter-ART (matrix w^{ab}) module, are initiated with values equal to 1, since all the activities are inactive.

3.2.3 Parameters of the Network

The parameters used in the processing of the Fuzzy ARTMAP network are the same used in the Fuzzy ART network. However, each module ART$_a$ and ART$_b$ receives a specific pattern. Besides these, there is the vigilance parameter of the Inter-ART module, being: $\rho_{ab} \in [0, 1]$.

3.2.4 Match Tracking

In the Fuzzy ARTMAP neural network the modules ART$_a$ and ART$_b$, are processed and, after resonance is confirmed in each one of them, one has:

- active category for the module ART$_a$: J

- active category for the module ART$_b$: K

After the confirmation of the resonance in each module, the test match tracking is performed, given by (10) [7]:

$$\left| x^{\mathrm{ab}} \right|_i = \frac{\left| y_i \wedge w^{\mathrm{ab}_J} \right|}{\left| y_i \right|} \tag{10}$$

By the vigilance criterion, we have that [7]:

- If, $\left| x^{\mathrm{ab}} \right|_i \geq \rho_{\mathrm{ab}}$ - the training pair should be confirmed;

- If, $\left| x^{\mathrm{ab}} \right|_i < \rho_{\mathrm{ab}}$ - another index *J*, that satisfies the vigilance parameter should be found.

Otherwise, small increases are successively made to the vigilance parameter of the module ART_a, until the vigilance criterion is satisfied.

3.2.5 Learning (Updating Weights)

Only after the state of resonance occurs, the process of training and learning is performed, modifying the weight vector given by (11) and (12), respectively, modules ART_a and ART_b and, (13) and (14) module Inter-ART [7].

$$w^{\mathrm{a}}\, J^{\mathrm{new}} = \beta(I \wedge w^{\mathrm{a}}\, J^{\mathrm{old}}) + (1-\beta)w^{\mathrm{a}}\, J^{\mathrm{old}} \tag{11}$$

$$w^{\mathrm{b}}\, K^{\mathrm{new}} = \beta(I \wedge w^{\mathrm{b}}\, K^{\mathrm{old}}) + (1-\beta)w^{\mathrm{b}}\, K^{\mathrm{old}} \tag{12}$$

$$w^{\mathrm{ab}}\, JK^{\mathrm{new}} = 0,\ \mathrm{para}\ k = 1,2,...,N,\quad k \neq K \tag{13}$$

$$w^{\mathrm{ab}}\, JK^{\mathrm{new}} = 1 \tag{14}$$

4 Methodology

This study was conducted under the Federal Institute of Education, Science and Technology of Mato Grosso - IFMT. The universe of interest are the students enrolled in the Colleges of Technology (CT) in Automation and Industrial Control, Control Works, Internet Systems, Computer Networks and Executive Secretary at IFMT, attending presential courses in the morning, afternoon and evening. The choice is justified in view of the high dropout rates, verified by previous statistical studies, noting that CT Automation and Industrial Control, reached a dropout rate of 62.46% from 2004 to 2010 [5].

In the implementation and pilot test of the intelligent system proposed, the neural network was fed with data belonging to all the students enrolled in the CT, from 2004 to 2009, making a total of 1650 samples for the training phase, constituting the basis historical data. For diagnosis 499 samples, of data from the students enrolled in 2010 and 2011 were used [5].

The database for prediction of the risk group prone to evasion consists of the students' characteristics such as demographic factors, and factors internal and external to the school. These characteristics were lifted from data from the selection processes at IFMT, the Q-Selection, which stores the answers of the socioeconomic questionnaire filled by the students on the day they enroll for the selection examination and the Q-Academic, system of integrated academic management, where all the academic history of the IFMT students is concentrated [5]. It is that the database does not contain the names of the students, which are identified, only by numbers.

The input vector of the Fuzzy ARTMAP neural network is composed by 16 parameters considered as significant for the school dropout prediction and the output of the network constituted by two classes, evasion and non-evasion. The input-output vector pairs are represented in the binary coding, being the input vector composed by 41bits and, the expected response represented by 1 bit. A summary of the input and output variables of the neural network can be visualized in Table I.

TABLE I. Composition of input and output vectors.

	Position	Name	Abbreviation	Size
Characteristics of the Subvectors of *a* and *y*				
	a_1	Gender	Gen	1 bit
	a_2	Age Group	Ag	3 bits
	a_3	Ethnicity	Etn	3 bits
	a_4	Marital Status	MSt	3 bits
	a_5	People/House	P/H	3 bits
	a_6	Family Income	FI	3 bits
	a_7	Has a Computer	Comp	1 bit
	a_8	Parents' Education	PE	3 bits
	a_9	School of Origin	SO	3 bits
	a_{10}	Self-Evaluation	SEv	3 bits
	a_{11}	Where From	WF	1 bit
	a_{12}	Distance School-Residence	DistSR	3 bits
	a_{13}	Means of Transport	MT	3 bits
	a_{14}	Work	Wk	3 bits
	a_{15}	Study Shift	SS	2 bits
	a_{16}	Students/Classroom	S/C	3 bits
Output Vector (y)	*y*	Non-Evasion	NEv	1 bit
		Evasion	Ev	

(Left vertical label: Variable of the Input Vector (a) of the Network)

5 Fuzzy ARTMAP neural system proposed for the evasion prediction

The data that involve the study about evasion, are sometimes, complex, subjective, non-linear, inter-related and keep in themselves the specificities inherent to the different levels of teaching, courses and institutions that one can analyze, thus choosing ANN, as among its potentialities there is the possibility of processing problems where complex and unknown relations are involved among different sets of data

and, also adjust the relations of non-linearity between the input and outup variables [5]. More specifically, the Fuzzy ARTMAP network, where the training is carried out in a supervised and self-organized way, with possibility of continued learning,   as implemented in [23]. Its application potential aims at solving several problems of classification and of approach of non-linear functions and showing prompt reply.

The input of the Fuzzy ARTMAP network proposed is represented by vector *a* (input of the module ART_a) and its desired output, in the training phase, represented by

vector b (input of the module ART_b), being these ones described in the following way:

$$a = [a_1\ a_2\ a_3\ \ldots\ a_{16}]$$
$$b = [\ b\ ]\quad,\quad \text{where: b = "0" ou "1"}$$

The subvectors a_1, a_2, a_3, . . . a_{16} of the vector a (Table I) are row vectors which contain the binary representation of the students' characteristics. Each bit corresponds to one component of the corresponding vector.

The network output is represented by the activity layer vector F_2 (y) and provides answers in the binary coding with 1 bit, being that code "1" corresponds to students' evasion and code "0" to non-evasion, defined as follows:

$$y = [\ y\ ]\ (\textit{Fuzzy}\ \text{ARTMAP network output})$$

The model proposed in this study consists of an intelligent system (flowchart shown in Fig. 1) for the study of students' evasion in the IFMT, using an Fuzzy ARTMAP Neural network [6-8], Logic Fuzzy [24] and/or Dempster-Shafer's Theory of Evidence - TDS [25].
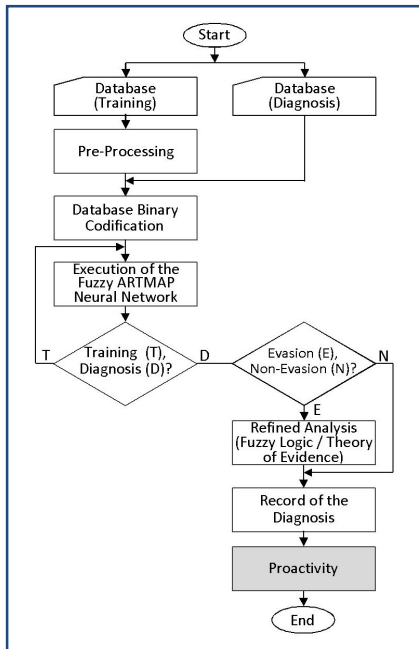
Fig. 1. Flowchart of the neural system proposed to perform the prediction of the evasion group risk.

The information of the database is pre-processed and converted into a binary database. The essentially binary conception is considerably worthwhile, because the neural network presents a more efficient behavior (prompt and better quality of answers) and allows the extraction of knowledge in a continuous way (continued training), seeking for a better adaptation to the conditions of the institution and improvement with time.

In the phase of the neural analysis, if the answer is negative in relation to evasion, no action is adopted; just the register of the mentioned information is performed. If the answer of evasion is positive, the following step corresponds to a better discrimination about the quality of information (fine analysis) based on the use of Fuzzy module [24] and/or of the Dempster-Shafer's Theory of Evidence [25]. Later, solutions that aim to revert students' evasion will be proposed (proactive action).

6 Application and analysis of the results

The intelligent system, using a Fuzzy ARTMAP, Neural Network proposed to make the prediction of the risk group of students prone to evasion, was simplemented and tested with a database composed by 1.650 rows e 42 columns in the training phsase of the network. A sample with 499 lines and 41 columns in the phase of validation and diagnosis, about 30% of the training samples. Each line represents the inputs standard vector and its corresponding desired output, in the training. The data of the columns from 1 to 41 represent the attributes correspondent to vector a, input of the module ART_a. In column 42 are represent the desired outputs, vetcor b (input of the module ART_b) of the Fuzzy ARTMAP neural network.

The parameters used in the database processing are specified in Table II.

TABLE II.   Specification of the parameters: Fuzzy ARTMAP neural network.

Parameters and References Values	
Parameters	*Values*
Choice parameter ($\alpha > 0$)	0.001
Training rate ($\beta \in [0,1]$)	1.0
Vigilance parameter module ART_a ($\rho_a \in [0,1]$)	0.2
Increasing in the vigilance parameter ρ_a (ε)	0.05
Vigilance prameter module ART_b ($\rho_b \in [0,1]$)	0.999
Vigilance parameter module inter-ART_{ab} ($\rho_{ab} \in [0,1]$)	0.7
Vigilance parameter in the match tracking ($\rho_{amat} \in [0,1]$)	0.75

After the network training five simulations were performed, based on data for the diagnosis, for the validation of the model proposed, being that, in one of them the samples were processed in a naturally random way and the others in a randomized way.

The results of the processing were compared and analyzed, using a criterion, called "voting criterion" [7], "0" or "1" of higher incidence for each of the inputs. The result of higher incidence constitutes the output of the neural network.

Later, comparing the output from the network with the real situation of each sample of the group of students analyzed, it was possible to investigate the coincidence of the evasion ("1") and non-evasion ("0") among the samples processed and the reality.

After concluding the phases of the processing of database through an Fuzzy ARTMAPNeural Network and respective analyses necessary to the understanding of the behavior in relation to students' evasion and non-evasion, the results were compiled and, briefly, shown in Table III.

TABLE III. Quantitative and perceptual results of the diagnosis of school evasion prediction .

Diagnosis of School Evasion	Quantitative and Percentages Values: Output of Network					
	Evasion		Non-Evasion		Total of Samples	
	Number	%	Number	%	Number	%
Samples	90	100	409	100	499	100
Corrects	88	97.8	295	72.1	383	76.7
Errs	2	2.2	114	27.9	116	23.3

The reading, interpretation and data analysis in Table III show that:

- of 499 samples, 90 of them corresponded to the evaded students and, 409 students who had concluded or attending a course, that is, not-evading.

- of 90 samples of evasion, the proposed system identified 88 evasion possibilities and ignored 2, with a margin of success of 97.8%.

- among the 409 samples of non-evasion, the Fuzzy ARTMAP network proposed recognized 295 samples in this situation and did not hit the target in 114, getting it right in 72.1% of the cases.

- it reached the global accuracy of 76.7%, finding correctly 383 samples of a total of 499.

The quantitative and percentage results of the previous diagnosis of the students with possibility of evasion can be perceived, more clearly in the graphs of Fig. 2 and Fig. 3, respectively.
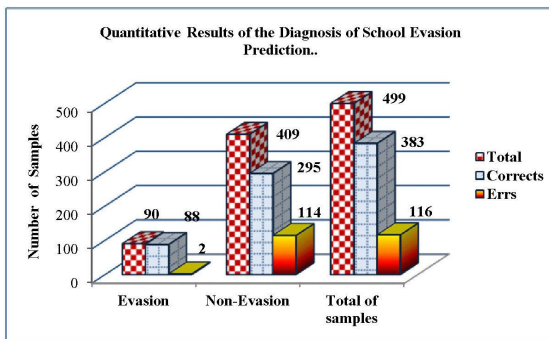
Fig. 2. Qualitative result of prediction of school evasion.
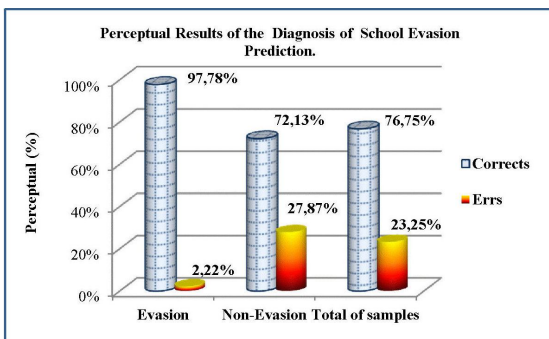
Fig. 3. Percentage result of school evasion.

Considering the experiment done and consistency of the results obtained, it can be inferred that the intelligent system, using Fuzzy ARTMAP, neural network proposed to identify the students prone to evasion, is a model with a significant degree of reliability and expresses accurately the situation in which the students analyzed are.

Conclusion

This study presented an innovative method to identify, in a proactive, continued and accurate way, the students considered to belong to the risk group of school dropout, using Fuzzy ARTMAP neural network.

The analysis of the results showed that the proposed system is satisfactory, with global accuracy superior to 76%, and with a significant degree of reliability, making possible the early identification, even in the first term of the course, the group of students likely to drop out. The anticipated identification of this group of students enables the institutional education, alongside the multidisciplinary team to adopt strategic, proactive and individualized measures with the aim of reducing or even mitigating the students' evasion.

It has been noted the consistency of the results, the accuracy and the efficiency of the system. However, one is dealing with the prediction of actions that come from the deliberations and decisions of human beings. Therefore, one recognizes the limitations of the methodology and its possible flaws, since the predictions fall beyond the complete determinism being evasion the result of a stochastic process.

After the conclusion of this draft experiment, it has been verified that the method can be extended and recommended for implementation to other groups where considerable levels of evasion were observed, adjusting it to the specificities of each group to be investigated.

The system proposed, using the Fuzzy ARTMAP Neural Network has the advantage of the possibility of working with a set of complex data, as well as the insertion of new training patterns without the necessity of restarting the process, in view of its plasticity, therefore allowing for continuous learning.

Hence, based on the results of the experiment, it is evident that the method proposed is a powerful, robust and innovative tool for the prediction of risk groups of student dropout, in higher education presential courses. It fills the gap existent in the worldwide scientific community productions, regarding the issue in question, and contributes with something that it is useful to society. Thus, with proactive action, to get a student with the potential to become a dropout to be successful is a noble mission, because they are dreams that become reality.

Acknowledgment

academic systems database available and CAPES for financial support.

References

[1]   T. Ross, G. Kena, A. Rathbun, A. Kewalramani, J. Zhang, P. Kristapovich, and E. Manning, "Higher Education: Gaps in Access and Persistence Study " U.S. Department of Education, Washington DC, Statistical Analysis Report NCES 2012-046, Aug 2012.

[2]   N. Gier, "The Crisis in Idaho Higher Education: A Lack of Faculty Autonomy as well as Funding," in Idaho State Journal, ed. Idaho - USA, 2009.

[3]   National Institute for Educational Studies and Research Anísio Teixeira - INEP, "Census of higher education: 2010 - technical summary," INEP, Brasília - DF, 2012. (in Portuguese).

[4]   UNESCO, "Higher Education in the Twenty-first Century: Vision and Action," World Conference on Higher Education, Paris, 5-9 October 1998.

[5]   V. R. C. Martinho, "Intelligent System for Prediction of Group Risk Evasion Student," (PhD in Electrical Engineering), PosGraduate Program in Electrical Engineering, Electrical Engineering Department, Campus of Ilha Solteira, UNESP, Univ Estadual Paulista, University Estadual Paulista - UNESP, Ilha Solteira - SP, 2012. (in Portuguese).

[6]   G. A. Carpenter, S. Grossberg, and K. Iizuka, "Comparative performance measures of fuzzy ARTMAP, learned vector quantization, and back propagation for handwritten character recognition," in: Neural Networks, 1992. IJCNN., International Joint Conference on, 1992, pp. 794-799 vol.1.

[7]   G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen, "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps," Neural Networks, IEEE Transactions on, vol. 3, pp. 698-713, 1992.

[8]   G. A. Carpenter and S. Grossberg, "A self-organizing neural network for supervised learning, recognition, and prediction," Communications Magazine, IEEE, vol. 30, pp. 38-49, 1992.

[9]   C. A. S. Baggi, "Evasion and institutional evaluation: a bibliographical discussion," Dissertation (Master of Education), Centre for Applied Social and Human Sciences, University of Campinas - PUCCamp, Campinas - SP, 2010. (in Portuguese).

[10]   R. L. L. Silva Filho, P. R. Motejunas, O. Hipólito, and M. B. d. C. M. Lobo, "The dropout in the Brazilian higher education," Research notebooks, vol. 37, pp. 641-659, Set/Dez 2007. (in Portuguese).

[11]   V. Tinto, "Dropout from Higher Education: A Theoretical Synthesis of Recent Research," Review of Educational Research, vol. 45, pp. 89-125, Winter 1975.

[12]   N. Viana and A. Rullán, "Reflections on dropouts in Finland and Puerto Rico," in Analytical Archives of Educational Policies vol. 18, ed. Arizona - North America: Arizona State University, 2010. (in Spanish), p. 33.

[13]   J. M. Gasper, S. DeLuca, and A. Estacion, "Switching Schools: Revisiting the Relationship Between School Mobility and High School Dropout," American Educational Research Journal, vol. 49, pp. 487-519, 1 Jun 2012.

[14]   S. Pal, "Mining Educational Data Using Classification to Decrease Dropout Rate of Students," International Journal of Multiisciplinary Sciences and Engineering, vol. 3, pp. 35 - 39, MAY 2012.

[15]   M. Jadrić, Ž. Garača, and M. Ćukušić, "Student Dropout Analysis with Application of Data Mining Methods," Journal of Contemporary Management Issues, vol. 15, pp. 31-46, Jun 2010.

[16]   S. L. Souza, "Evasion in higher education: a study using data mining as a tool for knowledge management in a database on the undergraduate engineering," Dissertation (Masters in Engenahria), Coordination of Postgraduate Engineering - COPPE, Rio de Janeiro Federal University - UFRJ, Rio de Janeiro - RJ, 2008. (in Portuguese).

[17]   A. Cripps, "Using artificial neural nets to predict academic performance," presented at the Proceedings of the 1996 ACM symposium on Applied Computing, Philadelphia, Pennsylvania, USA, 1996.

[18]   S. T. Karamouzis and A. Vrettos, "An Artificial Neural Network for Predicting Student Graduation Outcomes," in: World Congress on Engineering and Computer Science - WCECS - 2008, San Francisco, USA, 2008, pp. 991-994.

[19]   S. Haykin, Neural Networks: Comprehensive Foundation: Prentice Hall, 1999.

[20]   S. Herzog, "Estimating student retention and degree-completion time: Decision trees and neural networks vis-à-vis regression," New Directions for Institutional Research, vol. 2006, pp. 17-33, Oct 2006.

[21]   I. Lykourentzou, I. Giannoukos, V. Nikolopoulos, G. Mpardis, and V. Loumos, "Dropout prediction in e-learning courses through the combination of machine learning techniques," Journal Computers & Education, vol. 53, pp. 950-965, Nov 2009.

[22]   G. A. Carpenter, S. Grossberg, and D. B. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system," Neural Network, vol. 4, pp. 759-771, 1991.

[23]   S. C. Marchiori, M. C. G. da Silveira, A. D. P. Lotufo, C. R. Minussi, and M. L. M. Lopes, "Neural network based on adaptive resonance theory with continuous training for multi-configuration transient stability analysis of electric power systems," Applied Soft Computing, vol. 11, pp. 706-715, 2011.

[24]   L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338-353, 1965.

[25]   J. A. Barnett, "Computational methods for a mathematical theory of evidence," presented at the Proceedings of the 7th international joint conference on Artificial intelligence, Vancouver, BC, Canada, 1981.

Designing a Hybrid Neuro-Fuzzy System for Classifying the Complex Data, Application on Cornea Transplant

Mojtaba Sedigh Fazli[1], Keykhosrow Keshavarzi[2], Saeed Setayeshi[3]

Abstract— **Artificial Neural Networks are one of the Best tools for classification of complex sets of patterns. It's a crucial issue which could assist physicians to make correct decisions. In this article, a specific application of AI is applied for a biomedical engineering purpose. The aim is to find the best classifier for discriminating Lasik eyes from non- Lasik ones, using neural networks. Previously Porkar and Sedigh Fazli [8] have been showed that how HMM can optimize the traditional statics method and now for obtaining more optimized systems, we continued our researches on using Intelligent neural networks supported by fuzzy logic .Two models are applied: One is MLP which is a base model on neural network and the other classifier is a hybrid neuro-fuzzy model called LoLiMoT. This process seems to be more accurate compared to statistical ones.**

Keywords: *Neural networks, hybrid neuro-fuzzy model, MLP, Locally linear model tree, classification, corneal topography, Cornea transplant*

1. INTRODUCTION:

Some 46,000 people have cornea transplants each year. This is a sight-saving surgery [5].Corneal surgery includes a complicated process; including two major steps: the first step consists of another process called donation process. It means that, in this process the donor will dedicate his cornea to the eye bank for transplanting the cornea to candidates. In the 2nd phase, the eye surgery specialist will do the cornea transplant. Nowadays, one of the most important issues that are useful in many applications is to find the best fitting line and curve among the data. In fact, the aim of this work is to find limits for every category of data, helping us to find the order of data and to do a highly-accurate classification. Using the fitting line and curve has a vast usage, for instance one of the new applications of this method is in quantitative analysis of data. For example, it has already applied to determine the stickiness of proteins

of DNA [1]. In another article, there's a hint for the role of artificial neural system for dispersal data, this method is useful in data fitting [2] also previously some other intelligent systems has been used of image data classification [3].

2. LITERATURE REVIEW

2.1 Cornea:

The schema of an eye and its parts are illustrated in figure 2-1, the cornea can be seen in that figure.
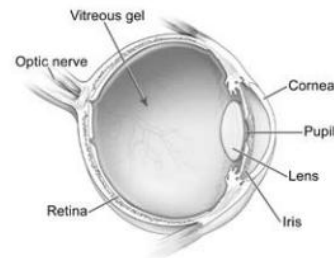
Fig. 1.1 schema of an eye

The cornea is the clear, dome-shaped tissue covering the front of the eye. It is about the size of a dime and the thickness of a credit card. The cornea is kept moist and nourished by a thin layer of tears. It is kept smooth by the blinking of the eyelids. If the cornea becomes distorted in shape, scarred, or hazy (opaque) from disease or injury, the light rays passing through it are distorted and the vision is reduced. In some cases, a corneal transplant may be necessary to replace the diseased or injured cornea with a healthy, clear cornea to restore good vision [5].

2.2 Corneal surgery challenge:

There is a big challenge in cornea surgery which is called the appropriation of donated cornea. Generally, there are some conditions which should be avoided in donor cornea which are listed as follows [6]:

1) Viral transmission of disease
2) Transmission of bacterial or fungal infections
3) Transmission of malignant disease
4) Transplantation of corneal disorders or those corneas which pose a risk to the success of the surgery.

[1] *Doctorate Student, University of Montesquieu Bordeaux IV, Bordeaux, France, Mojtabafazli@yahoo.com*
[2] *Associated Professor, Iran Medical Science University, Tehran, Iran, Dr.keshavarzi@gmail.com*
[3] *Associated Professor, Amir Kabir University of Technology, Tehran, Iran, Setayesh@aut.ac.ir*

In the 4th category, the main sub-set includes Lasik eyes [7]. It means that the eyes which already had a Lasik operation are not appropriate for the cornea transplantation and in this case, the achievement would be only a big pain and some other side-effects for cornea transplant candidate. Also, it's not so easy for transplant experts to find out if the donor cornea is a Lasik one or not? [7]

To successfully implement these standards, Eye Banks must have consistent policies for the examination and the documentation of a prospective donor's available medical records, medical history or investigation of cause of death. One of the basic problems in graft surgery is to examine the quality of the dedicated corneal, since if the quality is low, corneal graft won't have a significant effect on the sight of the person. For instance, a factor which is very important here is to have no previous surgery on eyes and specially Lasik surgery; besides that not having illnesses like corneal hump are the parameters which are important for corneal gets grafted. One of the challenging problems in this project is to lose eye pressure after death which has a great effect on topography. Usually after death all parts of the body including eye get swelled which has an effect on the thickness of corneal and the topography. Thus, when the system gives us the results, they would have some problems [6].

2.3 Line fitting and classification [11]:

In a 2 dimensional space, It assumes that, there is a line like "m" and it's defined as (x_i, y_i), I =1… m. For finding the best line we need to find the best a,b in the function as follows:

$$F(x) = ax + b \quad (1)$$

A sample for the best fitting for some specific points is illustrated in figure 2-2. In order to find the best line it needs to calculate the y distance of each point from the curve and the curve which has the minimum sum of the distance will be selected as the best fitting one, these distances are shown in figure 2-2 part B.
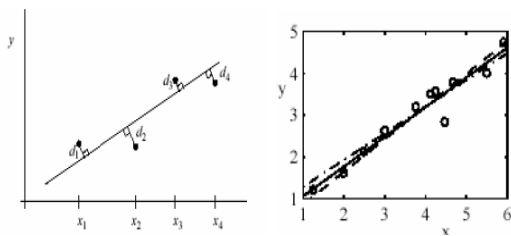
Figure 2-2: A. Sample of line fitting. B. distance of points to best line

2.4 Curve Fitting Using HMM [8]:

Previously, it has been shown that, in designing a classifier using Markov models for each stage it uses the line gained in the previous stage by Markov (Which is the major feature of Markov chain). And it could strongly fit some complex categories by Porkar, Sedigh Fazli and colleagues [8]. There is a sample of results for four points in the figure 2-3.
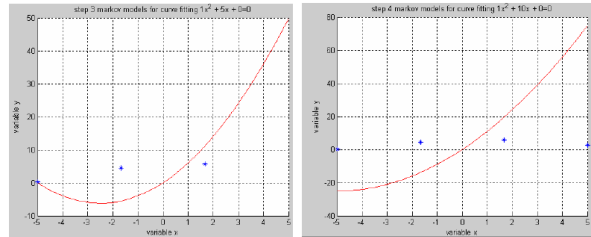
Figure 2-3. Curve fitting with HMM (A. stage 3 B. stage 4)

2.5 MLP model

MLP model is the general and the most famous model of neural networks. The architecture is illustrated as follows:
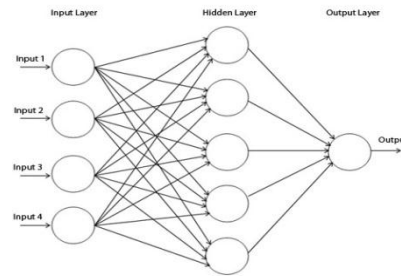
Figure 2 -4: MLP network Architecture

2.5.1 Training Process:

Training algorithm which is used here is supervised learning model. Here, the input layer units distribute input signals to the network. Connection weights modify the signals that pass through it. Hidden layers and output layer include a vector of processing elements with an activation function which is usually the sigmoid function. So that, the output of each processing unit for the forward pass will be defined as follows [12]:

$$S_i = \sum_{j=0}^{n} W_{ij} * u_j \quad (2)$$

$$u_i = f(S_i) \quad (3) \text{where} \quad f(x) = \frac{1}{(1 + e^{-x})} \quad (4)$$

In the backward phase, algorithm makes use of the error back-propagation algorithm for weights adjustment through Gradient Descent approach.

2.6 LoLiMoT neuro-fuzzy model

2.6.1 Neuro-fuzzy modeling [12, 13, 14]

Here the major idea is to divide the input space into smaller linear subspaces with fuzzy validity functions $\varphi_i(u)$ used to apply the hybrid locally linear neuro fuzzy model for function estimation. These functions describe the validity of each linear model in its region [13]. The validity function applied here is the normalized Gaussian function, which is defined as:

$$\mu(x) = \exp(-\frac{(x-c)^2}{2\sigma^2}) \quad (5)$$

Where c is the center and s is the standard deviation of the Gaussian. The whole model is a neuro-fuzzy network with one hidden layer and a linear neuron in the output layer which simply calculates the weighted sum of the outputs of locally linear models (LLMs) as:

$$\hat{y}_i = \omega_{i_0} + \omega_{i_1} u_1 + \omega_{i_2} u_2 + \ldots + \omega_{i_p} u_p$$

$$\hat{y} = \sum_{i=1}^{M} \hat{y}_i \phi_i(\underline{u})$$

$$(6), (7)$$

Where $\underline{u} = [u_1 \ u_2 \ \ldots \ u_p]^T$ is the model input, M is the number of LLM neurons, and W_{ij} denotes the LLM parameters of the ith neuron[13]. The validity functions are chosen as normalized Gaussians; normalization is necessary for a proper interpretation of validity functions:

$$\phi_i(\underline{u}) = \frac{\mu_i(\underline{u})}{\sum_{j=1}^{M} \mu_j(\underline{u})}$$

$$(8)$$

$$\mu_i(\underline{u}) = \exp\left(-\frac{1}{2}\left(\frac{(u_1 - c_{i1})^2}{\sigma_{i1}^2} + \ldots + \frac{(u_p - c_{ip})^2}{\sigma_{ip}^2}\right)\right)$$

$$(9)$$

Each Gaussian validity function has two sets of parameters, centers (C_{ij}) and standard deviations (δ_{ij}) which are the 2M. Optimization or learning methods are used to adjust two sets of parameters: the rule-consequent parameters of the locally linear models (W_{ij}) and the rule premise parameters of validity

functions $(C_{ij}$ and $\delta_{ij})$. A least squares optimization method is used to adjust the parameters of local linear models (W_{ij}), and a learning algorithm (described below) is used to adjust the parameters of validity functions $(C_{ij}$ and $\delta_{ij})[14]$. Global optimization of linear parameters is simply obtained by the least squares technique. The complete parameter vector contains $M(p + 1)$ elements:

$$\underline{\omega} = \begin{bmatrix} \omega_{10} & \omega_{11} & \cdots & \omega_{1p} & \omega_{20} & \omega_{21} \\ & & & \cdots & \omega_{M0} & \cdots & \omega_{Mp} \end{bmatrix}^T$$

$$(10)$$

And the associated regression matrix X for N measured data samples, is:

$$\underline{X} = [\underline{X}_1 \ \underline{X}_2 \ \ldots \ \underline{X}_M] \quad (11)$$

$$\underline{X}_i = \begin{bmatrix} \phi_i(\underline{u}(1)) & \cdots & u_p(1)\phi_i(\underline{u}(1)) \\ \phi_i(\underline{u}(2)) & \cdots & u_p(2)\phi_i(\underline{u}(2)) \\ \vdots & & \vdots \\ \phi_i(\underline{u}(N)) & \cdots & u_p(N)\phi_i(\underline{u}(N)) \end{bmatrix}$$

$$(12)$$

Thus:

$$\hat{\underline{y}} = \underline{X}.\hat{\underline{\omega}} \quad ; \quad \hat{\underline{\omega}} = (\hat{\underline{X}}.\underline{X} + \alpha\underline{I})^{-1} \underline{X}^T \underline{y} \ : \ \alpha \ll 1$$

$$(13)$$

Where α is the regularization parameter for avoiding any near singularity of matrix X^TX. The structure of LLNF is shown in Figure 2-5. The remarkable properties of locally linear neuro-fuzzy model, its transparency and intuitive construction, lead to the use of least squares technique for rule antecedent parameters and incremental learning procedures for rule consequent parameters. All the possible divisions in the p dimensional input space are checked and the best is performed. For more detail refer to [12, 13, 14].
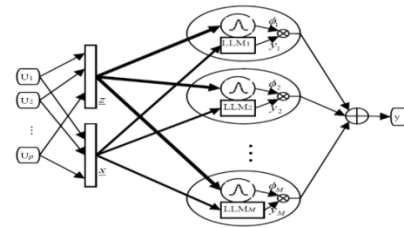
Figure 2-5: Structure of locally linear neuro-fuzzy model

2.6.2. Learning Algorithm

Locally Linear Model Tree (LOLIMOT) is a progressive tree construction algorithm that partitions the input space by axis bisection in all directions of input space. The LOLIMOT algorithm is described in five steps according to [14]:

1. Start with an initial model: Start with a single LLM, which is a global linear model over the whole input space with $\varphi_1(u) = 1$, and set M = 1. If there is a priori input space partitioning, it can be used as the initial structure.

2. Find the worst LLM: Calculate a local loss function, for example, mean square error (MSE), for each of the i = 1,..., M, LLMs and find the worst performing LLM.

3. Check all divisions: The worst LLM is considered for further refinement. The hyper rectangle (more than a three-dimensional rectangle or cube) of this LLM is split into two halves with an axis orthogonal split. Divisions in all dimensions are tried, and for each of the p divisions, the following steps are carried out. First, construct the multidimensional membership functions for both generated hyper rectangles and construct all validity functions: In part a, only the membership function of LLM that is split would change and the membership function of other neurons do not change, but all of the validity functions change that must be updated for all LLMs. Second, estimate the rule-consequent parameters for newly generated LLMs and third, calculate the loss function for the current overall model.

4. Find the best division: The best of the p alternatives checked in step 3 is selected, and the related validity functions and LLMs are constructed. The number of LLM neurons is incremented M = M + 1.

5. *Test the termination condition*: If the termination condition is met, then stop; otherwise, go to step 2. The termination condition reaches to a predefined error between output (y) and LLNF output with M neuron $(\hat{y})$, that is, when the condition $\| y - \hat{y} \| \ll \varepsilon$ is satisfied. In practice, we used a predefined number of neurons to LOLIMOT, plotted the error as a function of this number, and kept increasing the number of neurons until satisfactory performance was obtained. Details can be found in a work by Nelles [14].

Figure 2-6 illustrates the operation of the LOLIMOT algorithm in the first four iterations for a two-dimensional input space. In iteration 1, a global linear model is fit to data. Then for refinement, input space is split into halves, and a local linear model is fit in each hyper rectangle. In iteration 2, first, the best possible splitting method is selected (e.g., in Fig.2-5,

iteration 2 splitting along the u2 axis is assumed to be better), then in the selected model, the worst LLM should be used for further refinement (shaded rectangle or 2-1, for instance), and the algorithm continues with a default number of LLMs.
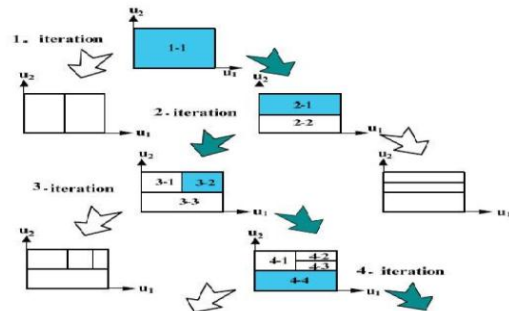
Figure 2-6: Operation of the LOLIMOT algorithm in the first five iterations for a two dimensional input space.

2.7 Corneal Topography

2.7.1 Medical analysis of the different pictures from the systems [9]:

The most important image is the one used for analysis of the surface of the corneal, called axial map or pathfinder. This picture, in general, shows the thickness of different parts of the corneal and in this task there had been more work on these two pictures to result preferences [10]. Also Keratometery view image gives some information related to the thickness of different parts of the corneal, the important point here is that in this picture all three circles and their thickness are important for graft surgery but for Lasik surgery only two of them are considered [8]. Other types of the pictures are numerical views which show the density of different parts thickness and we can realize how good the corneal is for the surgery through using this picture, to some extent [10].Photo Keratoscope cannot be sued for diagnosing Lasik. However, severe astigmatism and any surgery on corneal would be obvious and the effect is shown through some circles gathered in the direction of one axis. The final type of picture discussed here is profile view which can be useful for diagnosing as it shows cornea's curve beside the thickness [10] in figure 2-7 some types of the pictures are shown.
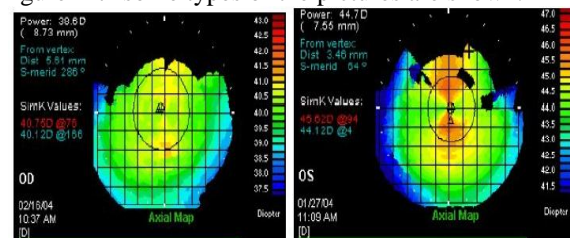
Figure 2-7-A. Numerical View   2-7-B. Keratometery View

2.8 Lasik

Nowadays, it's very popular to do the Lasik or laser surgery on eyes to improve some eye disabilities such as astigmatism and boosting patients' eyesight and vision, but as it was discussed before, the Lasik surgery will cause some abnormalities on corneal graft which will make the graft unsuitable for corneal transplant.

3. METHODOLOGY AND MODEL

3.1 Data

All the data are original data gathered from Iran Eye Bank which is the most reputed source in Iran. They generously gave us 254 samples of data including 197 non-Lasik eyes and 54 Lasik eyes. All of them were collected from Humphrey device. The simulations in each stage have been conducted through Matlab software R2010b edition.

3.2 Objectives and materials:

Here the main objective is to find out whether a donor cornea is a Lasik one or not? Here we focused on only Lasik and non-Lasik eyes due to the fact that we had data from these two categories .it was necessary to use some corneas topographical images which all were provided by Orbscan and Humphrey topography machines. Orbscan provides 2 major images from each eye: one is related to the front layer of cornea's graft and the 2nd one belongs to the back layer of cornea's graft. So, totally there are 4 images from everyone's eyes in Orbscan topography. It will help the experts to identify the keratoconus, it is of primary importance because the eyes with keratoconus abnormality are not suitable for corneal transplantation. Also its application is on diagnosis of hyperopia, Myopia and some other abnormalities on eyes. Humphrey is also presents some useful images. In this research the priority is to use the Pathfinder, axial and photokeratoscope images.

3.3 Process (Model):

For solving this problem, as it was mentioned in our previous work [8], we have 2 different Phases:

1. Preprocessing (image processing)
2. Data application in hybrid model

It could be urged that our solution to overcome this problem is illustrated in the following chart:
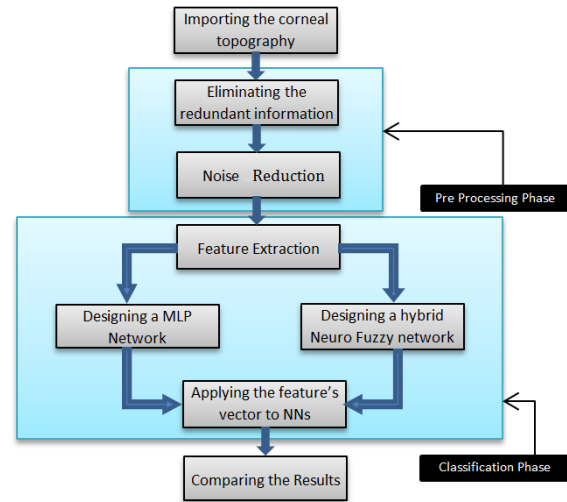
Figure 3-3: Process steps chart Including 2 Major parts (Image processing and the classification phase)

3.3.1 Preprocessing phase:

Preprocessing phase consists of an image processing phase. First, we have to select an image for classification. Humphrey's device images have been selected for image processing phase. The image which has been used for current research is named *Pathfinder*. As figure 2-7 shows, the image of corneal topography has redundant details. So, firstly it needs to eliminate some redundant parts of images. To standardize our work and to gain this objective, firstly some preprocesses has been conducted on the main image to extract a standard 159*159 image of corneal topography. The result is illustrated in Figure 3-2.
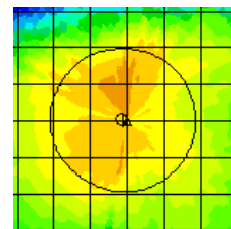
*Figure 3-2 Result of first step image processing on topography image 159*159 pixels dimensions.*

As it is obvious in figure 3-2, there are still some redundant lines and curves which can cause some predicaments in classification, thus there is a need to do more image processing to eliminate redundant information. In order to achieve this purpose, an image processing method with the help of average neighbor color is used to find the filled black point. For obtaining the best possible result median filter

has been used too, however, to obtain better results, a little change in popular median filter, which has an ordering color capability, has been done. Darker colors more than thresholds were eliminated from the list and median colors have been taken from primary color image. The outcome of this method is illustrated in Figure 3-3.
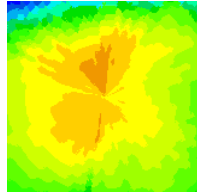
Figure3-3: The image obtained from 2nd image processing phase by eliminated lines.

Finally, for simplification the next step will be done on average image which is obtained from three major colors in RGB system, and a sample for that image is demonstrated in figure 3-4:
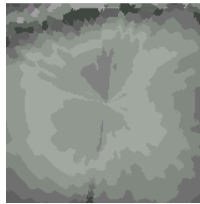
Figure3-4: Final image is now ready to start the work

3.4 Neural Network Models:

Now, we entered a new phase, called classification phase. In this new phase, two neural networks have been tested: the first one is a feed forward MLP model including 3 layers; in the first layer it includes 4 neurons and in the 2nd layer in includes 6 neurons and finally in last layer there is only one neuron (the activation function which is used here is sigmoid function). In the 2nd neural network, we tried to use a hybrid neuro-fuzzy model and we realized that the LoLiMoT is the best one. Thus, LoLiMoT with specifications, which were discussed before in literature review, has been used for pattern classification.

3.5 Feature Extraction:

In this research, to reach a better convergence to appropriate answer in classification, it was preferred to apply some individual features of each image instead of importing all image data. Since, applying all information in image data causes a divergence from the correct answer. After some continuous studies on all possible features, following ones have been selected [8, 6]:

a) Fast Fourier Transform (FFT)
b) Cross Correlation
c) d_{max}
d) d_{min}

Since in verified sample space, patterns are distributed complicatedly, using the mentioned two features cannot lead to an appropriate classification. Here using 2 features: FFT and Correlation, led to discriminating the classes but still the classes are too close to each other, therefore it will make some difficulties on classification. Here the solution is to extract some special features which individually belong to this specific application. These features come from the data obtained from Orbscan device which dominate some thickness values in 5 points of cornea. These two features are defined as follows:

$$d_{max} = V_m - V_{fmax} \quad (14)$$

$$d_{min} = V_m - V_{fmin} \quad (15)$$

Where V_m is the value in the middle of cornea and V_{fmin} is the minimum value of the four values besides cornea and Vfmax is the maximum value of the four values besides cornea.

For implementation of classifier, the patterns which are abstracted from Lasik corneas are gathered in one matrix and the non-Lasik ones in another matrix, thus the non-Lasik patterns for training set could be shown as follows:

$$X_N = (\bar{X}_{N1}\bar{X}_{N2} \dots \bar{X}_{NNn}) \quad (16)$$

$$X_N = \begin{bmatrix} X_{c1} & X_{c2} & \dots & X_{cNn} \\ X_{f1} & X_{f2} & \dots & X_{fNn} \\ X_{max1} & X_{max2} & \dots & X_{maxNn} \\ X_{min1} & X_{min2} & \dots & X_{minNn} \end{bmatrix} \quad (17)$$

Where X_N is non-Lasik cornea's patterns matrix and X_{ci}, X_{fi}, X_{maxI} and X_{minI} belong to the correlation value, FFT value, d_{max} value and d_{min} value for i^{th} pattern, respectively. So each column of the above matrix (each $\bar{X}_{Ni}$ is a vector which indicates a training pattern; also N_n shows the number of non-Lasik patterns. Similarly, the Lasik training patterns is presented in XL matrix:

$$X_L = (\bar{X}_{L1} \bar{X}_{L2} \dots \bar{X}_{LNL}) \quad (18)$$

$$X_{L}=\begin{bmatrix} X_{c1} & X_{c2} & ... & X_{cNl} \\ X_{f1} & X_{f2} & ... & X_{fNl} \\ X_{max1} & X_{max2} & ... & X_{maxNl} \\ X_{min1} & X_{min2} & ... & X_{minNl} \end{bmatrix} \quad (19)$$

Where N_L shows the number of Lasik patterns.

For generalization and simplification, the classifying system is designed in a way that it receives a matrix as input patterns and will determine the class of each pattern, so the Pattern and the Pattern class matrices could be defined as follows:

$Pattern = (X_N \; X_L)$ (20)

Then:

$Pattern = (\bar{X}_{N1}\bar{X}_{N2}... \; \bar{X}_{NNn}\bar{X}_{L1}\bar{X}_{L2}...\bar{X}_{LNL})_{(NN \; NL)}(21)$

$Pattern \; Class = (1 \quad 1 \; .... \; 1 \quad 2 \quad 2 \; .... \; 2)_{(NN \; NL)}$ (22)

In pattern class array, Number "1" shows the non-Lasik cornea's class and Number "2" determines the Lasik cornea's class.

4. RESULTS:

After doing all the process which was mentioned above, the data which include 4 selected features are applied on 2 different neural networks.

In the following table, the values of each extracted feature for around 18 images are demonstrated. But due to the limitation in paper only extracted features for the first 18th images are shown:

	Correlation	FFT	d_{max}	d_{min}
1	0.1674	195	73	106
2	0.0317	244	102	69
3	0.0206	255	31	95
4	0.0298	184	61	133
5	0.0821	191	88	185
6	0.0703	190	74	117
7	0.0314	185	107	151
8	0.0621	158	129	162
9	0.2104	240	104	172
10	0.0373	159	57	116
11	0.0277	241	62	124
12	0.259	234	66	128
13	0.2814	145	64	106
14	0.1172	252	28	114
15	0.0455	169	90	140
16	0.1783	160	51	95
17	0.2582	124	53	78
18	0.1187	147	81	126

Table 4-1: values of extracted features for first 18th patterns

Both simulations are conducted with 254available patterns including 221 training samples which are separated into 2 parts: one includes 185 non-Lasik patterns while the second one contains 47 Lasik cornea's patterns. The remaining 33 samples are

considered for the test set which includes 7 Lasik patterns and 24 non-Lasik patterns. The first one is done through MLP and the 2nd one is classified through LoLiMoT. The results are illustrated in the following figure   :
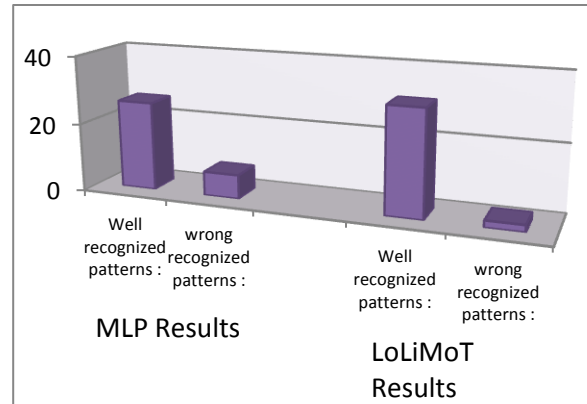
Figure 4-1: Results for MLP and LoLiMoT for 33 test patterns

As it illustrated in figure 4-1, the MLP performance is around 76.76% and 7 samples are recognized wrongly, LoLiMoT performance is around 94% and 31 out of 33 test patterns have been recognized correctly and only 2 of them were classified wrongly, but the good news is that all the Lasik samples are recognized correctly as Lasik patterns; while 2 non-Lasik patterns recognized wrongly as Lasik one.

In order to evaluation of gained results, these results are compared to our previous experience with our colleagues on HMM application on the same subject. So that all results are gathered and illustrated as follows:

MODEL	KNN	HMM	MLP	LoLiMoT
PERFORMANCE	75.00%	86.00%	76.76%	94.00%

As it is illustrated, the best classifier for this specific purpose is a classifier including LoLiMoT which is a Neuro-Fuzzy Hybrid system.

5. CONCLUSION:

In This paper, we proposed a novel method for the classification of corneal topography applied in cornea transplantation by using a Hybrid neuro-fuzzy model. As we saw, the results were more accurate than other systems such as HMM or a pure neural network. From a theoretical point of view, it presents a strong

classifier for complex distributed patterns. LoLiMoT proves that how an intelligent technique could optimize the answers in complex problems.

Our major limitation here was the number of samples, however this lack of data is already covered through nice cooperation of Iran Eye Bank, it seems that still by increasing the data size, we can achieve more promising results. Also, in the next step, it is proposed to work on other eye diseases through this expert system which can help the physicians to make accurate decision in their diagnosis.

REFERENCES

[1] Susan E.Shadle , Douglas F.Allen , Hong Guo , Wendy K. Pogozelski, John S. Bashkin and Thomas D. Tullius" *quantitative analysis ofelectrophoresis data : novel curve fitting methodology and itsapplication to the determination of a protein-DNA binding constant*" john Hopkins university , 1996.

[2] M. Hoffmann "*scattered data fitting by artificial neural network*" university of eger, 2002.

[3] T.Lexiao, Z. Dequan, Z. Conghui. "*Image Classification Based on the Combination of Text Features and Visual Features*", International Journal of Intelligent Systems, Vol. 28, Number 3, pp. 242-256, March 2013.

[4] M.R. Forster "*time series and curve–fitting:how are they related?*" January 1999.

[5] A. Sugar, "*Understand Corneal transplant Surgery*", Kellogg Eye Center, Michigan, 2008

[6] "*Factors influencing the suitability of organ-cultured corneas for transplantation*", *Invest Ophthalmol Vis Sci*38:16-24, 1997.

[7] Laura J. Rongé, "*LASIK: Beyond Refractive Surgery, Part I*", EyeNet Magazine Online, August 2001.

[8]P.Prokar, M.Sedigh Fazli, et al "*Use HMM and KNN for classifying corneal data*",International Journal of Soft Computing and Software Engineering [JSCSE], ISSN: 2251-7545,2013

[9] 'Humphrey atlas corneal topography system with mastervue software', 1998.

[10] M. Sedigh Fazli, S.Setayeshi "*Medicine Irregular Pattern Classification, Using Statistical Method Based On Neural Network Model*", University of Science & Research, Artificial Intelligence & Robotic Engineering Department, November 2005.

[11] Gerald Recktenwald, "*least squares fitting of data to a curve*"Portlanduniversity November 2001.

[12] D.E. Rumelhart, G.E. Hinton, R.J. Williams, "*Learning internal representations by error Propagation*" in: D.E. Rumelhart, J.L. McClelland (Eds.) Parallel Distributed Processing: Explorations in the Microstructures of Cognition, MIT Press, Cambridge, 1986, pp. 318-362.

[13] M.N. Maralloo, A.R. Koushki, C. Lucas, M.M. Pedram, "*Mutual Information Based Input Selection in Neuro-Fuzzy Modeling for Long Term Load Forecasting*", in: Computer Science & Information Technology, Yerevan, Armenia, 2009, pp. 209-213.

[14] O. Nelles, "*Nonlinear system identification*", Springer Verlag, Berlin, 2001.

An Automated Deduction of a "Dishkant-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 1

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Dishkant" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Dishkant implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Dishkant-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
    x = c(c(x))                    # label("AxL1").
    x v y = y v x                  # label("AxL2").
    (x v y) v z = x v (y v z)      # label("AxL3").
    (x ^ y) ^ z = x ^ (y ^ z)      # label("AxL4").
    x v (x ^ y) = x                # label("AxL5").
    x ^ (x v y) = x                # label("AxL6").

% Ortholattice axioms
    c(x) ^ x = 0                   # label("AxOL1").
    c(x) v x = 1                   # label("AxOL2").
    x ^ y = c(c(x) v c(y))         # label("AxOL3").

% Definition of Dishkant implication ([22])
    i2(x,y) = c(c(y)) v (c(y) ^ c(x))
                                   # label("Df: i2").

% Definition of Dishkant-implication-restricted join
    u2(x,y) =  i2(c(x),y)          # label("Df: u2").

% Definition of Dishkant-implication-restricted meet
    int2(x,y) = c(i2(x,c(y)))      # label("Df: int2").

% Definition of x commutes with y
    C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
          # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Dishkant-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. c(x) denotes the orthocomplement of x. v denotes lattice join; ^ denotes lattice meet.

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
    x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)   (x v (y ^ z) =
          (x v y) ^ (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

        C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int2 and u2, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i2 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 2 denotes Dishkant-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH1.2, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH1.2 is a consequence of OML:

```
============================== PROOF =================================

% Proof 1 at 2.39 (+ 0.08) seconds: "Foulis-Holland Theorem 1.2".
% Length of proof is 82.
% Level of proof is 15.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> int2(x,u2(y,z)) = u2(int1(x,y),int2(x,z)) # label("Foulis-Holland
Theorem 1.2") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
12 C(c1,c3) # label("Foulis-Holland Theorem 1.2").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
```

```
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
27 i2(x,y) = c(c(y)) v (c(y) ^ c(x)) # label("Df: i2").  [assumption].
28 i2(x,y) = y v c(y v x).  [copy(27),rewrite([14(3),23(4),14(3),14(3)])].
37 u2(x,y) = i2(c(x),y) # label("Df: u2").  [assumption].
38 u2(x,y) = y v c(y v c(x)).  [copy(37),rewrite([28(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
47 int2(x,y) = c(i2(x,c(y))) # label("Df: int2").  [assumption].
48 int2(x,y) = c(c(y) v c(c(y) v x)).  [copy(47),rewrite([28(3)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int2(c1,u2(c2,c3)) != u2(int1(c1,c2),int2(c1,c3)) # label("Foulis-Holland Theorem
1.2") # answer("Foulis-Holland Theorem 1.2").  [deny(4)].
78 c(c(c(c3 v c(c3 v c(c2))) v c(c1 v c(c3 v c(c3 v c(c2))))) != c(c(c3) v c(c1 v c(c3))) v
c(c(c1) v (c(c2 v c(c1)) v c(c(c3) v c(c1 v c(c3))))) # answer("Foulis-Holland Theorem
1.2").
[copy(77),rewrite([38(4),48(9),15(18),46(24),15(27),48(33),15(36),38(40),14(49),15(48),16
(48)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
109 x v (c(x v c(y v x)) v z) = y v (x v z).
[para(76(a,1),16(a,1,1)),rewrite([16(2)]),flip(a)].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
138 c(x v y) v c(x v c(x v y)) = c(x).
[para(84(a,1),76(a,1,2,1,2)),rewrite([15(5),133(11)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
165 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
193 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
195 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([193(5)])].
196 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
210 x v (y v (z v c(x v y))) = 1.  [para(195(a,1),16(a,1)),flip(a)].
220 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
223 x v c(y v (z v c(x))) = x.  [para(16(a,1),151(a,1,2,1))].
225 x v (y v c(z v c(x))) = y v x.  [para(151(a,1),89(a,1,2)),flip(a)].
236 c(x) v (y v c(x v z)) = y v c(x).  [para(133(a,1),89(a,1,2)),flip(a)].
257 c(x) v (c(y v x) v z) = c(x) v z.  [para(220(a,1),16(a,1,1)),flip(a)].
260 c(x) v (y v c(z v x)) = y v c(x).  [para(220(a,1),89(a,1,2)),flip(a)].
305 x v (y v c(z v (u v c(x)))) = y v x.  [para(223(a,1),89(a,1,2)),flip(a)].
708 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),225(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
741 x v c(x v c(y v c(x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),196(8)])].
771 c3 v c(c1 v c3) = c3 v c(c1).  [para(708(a,1),108(a,1,2,1)),rewrite([15(10)])].
988 c(c1 v c3) v c(c3 v c(c1)) = c(c3).
[para(708(a,1),138(a,1,1,1)),rewrite([708(12),771(10)])].
1025 c1 v c(c1 v c3) = c1 v c(c3).
[para(988(a,1),225(a,1,2)),rewrite([15(10)]),flip(a)].
```

```
1087 c1 v c(c1 v c(c3)) = c1 v c3.  [para(1025(a,1),108(a,1,2,1)),rewrite([15(10)])].
1096 c(c1 v c3) v c(c1 v c(c3)) = c(c1).
[para(1025(a,1),138(a,1,1,1)),rewrite([1025(12),1087(12),15(10)])].
1166 c(x) v (y v (z v c(x v u))) = y v (z v c(x)).
[para(16(a,1),236(a,1,2)),rewrite([16(9)])].
1399 c(c3) v c(c1 v c(c3)) = c(c1) v c(c3).
[para(1096(a,1),257(a,1,2)),rewrite([15(5)]),flip(a)].
1409 c(c(c3 v c(c3 v c(c2))) v c(c1 v c(c3 v c(c3 v c(c2)))) != c(c(c1) v c(c3)) v
c(c(c1) v (c(c2 v c(c1)) v c(c(c1) v c(c3)))) # answer("Foulis-Holland Theorem 1.2").
[back_rewrite(78),rewrite([1399(29),1399(42)])].
3701 x v c(y v c(x v y)) = x.
[para(741(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
3711 x v c(y v c(y v x)) = x.   [para(15(a,1),3701(a,1,2,1,2,1))].
3720 x v c(y v x) = x v c(y).
[para(3701(a,1),151(a,1,2,1)),rewrite([15(5),236(5)]),flip(a)].
3723 x v c(x v c(y)) = x v y.
[para(3701(a,1),109(a,1,2)),rewrite([3720(3),3720(7),3720(10),14(8),305(10)])].
3932 c(c(c1) v c(c3)) v c(c(c1) v (c(c2 v c(c1)) v c(c(c1) v c(c3)))) != c(c(c1) v c(c2 v
c3)) # answer("Foulis-Holland Theorem 1.2").
[back_rewrite(1409),rewrite([3723(7),15(3),3723(12),15(8),3720(12),15(7)]),flip(a)].
3995 x v (y v c(x v c(z))) = y v (z v x).  [back_rewrite(165),rewrite([3720(3)])].
4051 c(c(c1) v c(c2 v c3)) != c(c3 v (c(c1) v c(c2))) v c(c(c1) v c(c3)) #
answer("Foulis-Holland Theorem 1.2").
[back_rewrite(3932),rewrite([3995(21),15(16),16(16),3720(15),15(15)]),flip(a)].
4097 x v (y v c(z v x)) = x v (y v c(z)).
[para(210(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1166(6)]),flip(a)].
4105 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3711(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),260(8),15(4),133(4)]),flip(
a)].
4568 c(x v c(y v z)) = c(z v (x v c(y))) v c(x v c(z)).
[para(225(a,1),4105(a,1,2,1)),rewrite([14(2),15(4),4097(4),14(10)]),flip(a)].
4569 $F # answer("Foulis-Holland Theorem 1.2").  [resolve(4568,a,4051,a)].

============================= end of proof =========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH1.2 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. Note that some of "logical" proof lines in the above have been transformed to two text lines, with the *derivation* appearing on a text line following a text line containing the first part of that logical line. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 2.5 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

  1. FH1.2 is derivable from OML.

  2. The proof in Section 3.0 is, as far as I know, novel.

  3. Companion papers show that the other three Dishkant-implication restricted FH theorems are also derivable from OML.

  4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Dishkant-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Dishkant-implication-restricted FH theorems is the subject of future work.

5.0 Acknowledgements

This work benefited from discussions with Tom Oberdan, Frank Pecchioni, Tony Pawlicki, and the late John K. Prentice, whose passion for foundations of physics inspired those of us privileged to have known him. For any infelicities that remain, I am solely responsible.

6.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4.* URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH1.2 from OML. 2013. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I.* Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics.* Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Dishkant H. Semantics of the minimal logic of quantum mechanics. *Studia Logica* 30 (1972), 17-29.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Dishkant-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 2

Jack K. Horner

P. O. Box 266

Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Dishkant" implication (one of the QL implication-connectives), from OML theory.

Keywords: Dishkant implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11], [19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Dishkant-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
    x = c(c(x))                    # label("AxL1").
    x v y = y v x                  # label("AxL2").
    (x v y) v z = x v (y v z)      # label("AxL3").
    (x ^ y) ^ z = x ^ (y ^ z)      # label("AxL4").
    x v (x ^ y) = x                # label("AxL5").
    x ^ (x v y) = x                # label("AxL6").

% Ortholattice axioms
    c(x) ^ x = 0                   # label("AxOL1").
    c(x) v x = 1                   # label("AxOL2").
    x ^ y = c(c(x) v c(y))         # label("AxOL3").

% Definition of Dishkant implication ([22])
    i2(x,y) = c(c(y)) v (c(y) ^ c(x))
                                   # label("Df: i2").

% Definition of Dishkant-implication-restricted join
    u2(x,y) =  i2(c(x),y)          # label("Df: u2").

% Definition of Dishkant-implication-restricted meet
    int2(x,y) = c(i2(x,c(y)))      # label("Df: int2").

% Definition of x commutes with y
    C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
          # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Dishkant-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. c(x) denotes the orthocomplement of x. v denotes lattice join; ^ denotes lattice meet.

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
    x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)   (x v (y ^ z) =
        (x v y) ^ (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

        C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int2 and u2, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i2 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 2 denotes Dishkant-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH2.2, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH2.2 is a consequence of OML:

```
============================= PROOF =================================

% Proof 1 at 3.63 (+ 0.11) seconds: "Foulis-Holland Theorem 2.2".
% Length of proof is 84.
% Level of proof is 14.
% Maximum clause weight is 51.
% Given clauses 295.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> int2(y,u2(x,z)) = u2(int2(y,x),int2(y,z)) # label("Foulis-Holland
Theorem 2.2") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
12 C(c1,c3) # label("Foulis-Holland Theorem 2.2").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
```

```
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
27 i2(x,y) = c(c(y)) v (c(y) ^ c(x)) # label("Df: i2").  [assumption].
28 i2(x,y) = y v c(y v x).  [copy(27),rewrite([14(3),23(4),14(3),14(3)])].
37 u2(x,y) = i2(c(x),y) # label("Df: u2").  [assumption].
38 u2(x,y) = y v c(y v c(x)).  [copy(37),rewrite([28(3)])].
47 int2(x,y) = c(i2(x,c(y))) # label("Df: int2").  [assumption].
48 int2(x,y) = c(c(y) v c(c(y) v x)).  [copy(47),rewrite([28(3)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int2(c2,u2(c1,c3)) != u2(int2(c2,c1),int2(c2,c3)) # label("Foulis-Holland Theorem
2.2") # answer("Foulis-Holland Theorem 2.2").  [deny(4)].
78 c(c(c3 v c(c3 v c(c1))) v c(c2 v c(c3 v c(c3 v c(c1))))) != c(c(c3) v c(c2 v c(c3))) v
c(c(c1) v (c(c2 v c(c1)) v c(c(c3) v c(c2 v c(c3))))) # answer("Foulis-Holland Theorem
2.2").
[copy(77),rewrite([38(4),48(9),15(18),48(24),15(27),48(33),15(36),38(40),14(49),15(48),16
(48)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(2)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
97 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
144 x v (c(x) v y) = 1.  [back_rewrite(97),rewrite([141(5)])].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
193 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
195 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([193(5)])].
196 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
207 x v (y v (c(x v y) v z)) = 1.  [para(144(a,1),16(a,1)),flip(a)].
210 x v (y v (z v c(x v y))) = 1.  [para(195(a,1),16(a,1)),flip(a)].
220 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
221 x v (c(y v c(x)) v z) = x v z.  [para(151(a,1),16(a,1,1)),flip(a)].
225 x v (y v c(z v c(x))) = y v x.  [para(151(a,1),89(a,1,2)),flip(a)].
234 c(x) v (c(x v y) v z) = c(x) v z.  [para(133(a,1),16(a,1,1)),flip(a)].
236 c(x) v (y v c(x v z)) = y v c(x).  [para(133(a,1),89(a,1,2)),flip(a)].
257 c(x) v (c(y v x) v z) = c(x) v z.  [para(220(a,1),16(a,1,1)),flip(a)].
260 c(x) v (y v c(z v x)) = y v c(x).  [para(220(a,1),89(a,1,2)),flip(a)].
313 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),210(a,1,2,2,2,1))].
665 x v (y v (c(z v c(x)) v u)) = y v (x v u).  [para(221(a,1),89(a,1,2)),flip(a)].
708 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),225(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
727 c(c(c3) v c(c2 v c(c3))) v c(c(c1) v (c(c2 v c(c1)) v c(c(c3) v c(c2 v c(c3))))) !=
c(c(c1 v c3) v c(c2 v c(c1 v c3))) # answer("Foulis-Holland Theorem 2.2").
[back_rewrite(78),rewrite([708(7),708(12)]),flip(a)].
742 x v c(x v c(y v c(x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),196(8)])].
936 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).  [para(234(a,1),89(a,1,2)),flip(a)].
1410 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),260(a,1,2)),rewrite([16(9)])].
3701 x v c(y v c(x v y)) = x.
[para(742(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
```

```
3711 x v c(y v c(y v x)) = x.  [para(15(a,1),3701(a,1,2,1,2,1))].
3720 x v c(y v x) = x v c(y).
[para(3701(a,1),151(a,1,2,1)),rewrite([15(5),236(5)]),flip(a)].
3961 c(c(c2) v c(c3)) v c(c(c1) v (c(c2 v c(c1)) v c(c(c2) v c(c3)))) != c(c(c2) v c(c1 v
c3)) # answer("Foulis-Holland Theorem 2.2").
[back_rewrite(727),rewrite([3720(8),15(5),3720(21),15(18),3720(35),15(30)])].
4094 x v c(x v y) = x v c(y).
[para(3711(a,1),151(a,1,2,1)),rewrite([15(5),260(5)]),flip(a)].
4095 x v (c(y v x) v z) = x v (c(y) v z).
[para(207(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),936(6)]),flip(a)].
4101 x v (y v c(x v z)) = x v (y v c(z)).
[para(313(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1410(6)]),flip(a)].
4104 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3711(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),260(8),15(4),133(4)]),flip(
a)].
4151 c(c(c2) v c(c1 v c3)) != c(c3 v (c(c1) v c(c2))) v c(c(c2) v c(c3)) #
answer("Foulis-Holland Theorem 2.2").
[back_rewrite(3961),rewrite([4095(21),4094(17),14(13),15(12),89(13),15(15)]),flip(a)].
4341 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3720(a,1,2,1))].
4345 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3720(a,1),257(a,1,2)),rewrite([257(6)]),flip(a)].
4871 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4095(a,1),89(a,1,2)),flip(a)].
5754 c(x v y) v c(y v c(z v x)) = c(y).
[para(4341(a,1),4345(a,1,2,1)),rewrite([15(11),220(11)])].
5779 c(x v c(y v z)) v c(z v x) = c(x).  [para(5754(a,1),15(a,1)),flip(a)].
5899 c(x v c(y v z)) v c(y v x) = c(x).  [para(15(a,1),5779(a,1,1,1,2,1))].
6092 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5899(a,1,1,1))].
6102 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4101(a,1),5899(a,1,2,1)),rewrite([16(6)])].
6196 c(x v (c(y v z) v u)) = c(x v (z v (c(y) v u))) v c(x v (c(z) v u)).
[para(665(a,1),4104(a,1,2,1)),rewrite([14(2),15(5),4871(5),14(12)]),flip(a)].
6230 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6102),rewrite([6196(7),133(4),16(16),6092(15)]),flip(a)].
6565 $F # answer("Foulis-Holland Theorem 2.2").
[back_rewrite(4151),rewrite([6230(8),15(7),16(7)]),xx(a)].

============================== end of proof ==========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH2.2 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. Note that some of "logical" proof lines in the above have been transformed to two text lines, with the *derivation* appearing on a text line following a text line containing the first part of that logical line. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 2.5 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

1. FH2.2 is derivable from OML.

2. The proof in Section 3.0 is, as far as I know, novel.

3. Companion papers show that the other three Dishkant-implication restricted FH theorems are also derivable from OML.

4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Dishkant-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Dishkant-implication-restricted FH theorems is the subject of future work.

5.0 Acknowledgements

6.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4.* URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH2.2 from OML. 2013. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I.* Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

 [12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics*. Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Dishkant H. Semantics of the minimal logic of quantum mechanics. *Studia Logica* 30 (1972), 17-29.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Dishkant-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 3

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Dishkant" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Dishkant implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11], [19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Dishkant-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
    x = c(c(x))                    # label("AxL1").
    x v y = y v x                  # label("AxL2").
    (x v y) v z = x v (y v z)      # label("AxL3").
    (x ^ y) ^ z = x ^ (y ^ z)      # label("AxL4").
    x v (x ^ y) = x                # label("AxL5").
    x ^ (x v y) = x                # label("AxL6").

% Ortholattice axioms
    c(x) ^ x = 0                   # label("AxOL1").
    c(x) v x = 1                   # label("AxOL2").
    x ^ y = c(c(x) v c(y))         # label("AxOL3").

% Definition of Dishkant implication ([22])
    i2(x,y) = c(c(y)) v (c(y) ^ c(x))
                                   # label("Df: i2").

% Definition of Dishkant-implication-restricted join
    u2(x,y) =  i2(c(x),y)          # label("Df: u2").

% Definition of Dishkant-implication-restricted meet
    int2(x,y) = c(i2(x,c(y)))      # label("Df: int2").

% Definition of x commutes with y
    C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
          # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Dishkant-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. c(x) denotes the orthocomplement of x. v denotes lattice join; ^ denotes lattice meet.

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
    y v (c(y) ^ (x v y)) =
      x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
    (D)   (x v (y ^ z) =
          (x v y) ^ (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

      C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int2 and u2, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i2 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 2 denotes Dishkant-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH3.2, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH3.2 is a consequence of OML:

```
============================== PROOF =================================

% Proof 1 at 3.62 (+ 0.09) seconds: "Foulis-Holland Theorem 3.2".
% Length of proof is 95.
% Level of proof is 14.

4 C(Cx,y) & C(X,z) -> u2(x,int2(y,z)) = int2(u2(x,y),u2(x,z)) # label("Foulis-Holland
Theorem 3.2") # label(non_clause) # label(goal).  [goal].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
```

```
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
27 i2(x,y) = c(c(y)) v (c(y) ^ c(x)) # label("Df: i2").  [assumption].
28 i2(x,y) = y v c(y v x).  [copy(27),rewrite([14(3),23(4),14(3),14(3)])].
37 u2(x,y) = i2(c(x),y) # label("Df: u2").  [assumption].
38 u2(x,y) = y v c(y v c(x)).  [copy(37),rewrite([28(3)])].
47 int2(x,y) = c(i2(x,c(y))) # label("Df: int2").  [assumption].
48 int2(x,y) = c(c(y) v c(c(y) v x)).  [copy(47),rewrite([28(3)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int2(u2(c3,c1),u2(c3,c2)) != u2(c3,int2(c1,c2)) # label("Foulis-Holland Theorem 3.2")
# answer("Foulis-Holland Theorem 3.2").  [deny(4)].
78 c(c(c2 v c(c2 v c(c3))) v c(c1 v (c(c1 v c(c3)) v c(c2 v c(c2 v c(c3)))))) != c(c(c2)
v c(c1 v c(c2))) v c(c(c3) v c(c(c2) v c(c1 v c(c2))))) # answer("Foulis-Holland Theorem
3.2").
[copy(77),rewrite([38(3),38(10),48(15),15(24),16(24),48(31),15(34),38(38),15(48)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
97 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
109 x v (c(x v c(y v x)) v z) = y v (x v z).
[para(76(a,1),16(a,1,1)),rewrite([16(2)]),flip(a)].
110 x v (y v c(x v (y v c(z v (x v y))))) = z v (x v y).
[para(76(a,1),16(a,1)),rewrite([16(7)]),flip(a)].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
126 x v (c(y v x) v c(c(y) v x)) = 1.  [back_rewrite(68),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
144 x v (c(x) v y) = 1.  [back_rewrite(97),rewrite([141(5)])].
146 x v c(c(x) v y) = x.  [para(14(a,1),85(a,1,2,1,2))].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
165 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
193 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
195 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([193(5)])].
196 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
207 x v (y v (c(x v y) v z)) = 1.  [para(144(a,1),16(a,1)),flip(a)].
210 x v (y v (z v c(x v y))) = 1.  [para(195(a,1),16(a,1)),flip(a)].
215 x v (c(c(x) v y) v z) = x v z.  [para(146(a,1),16(a,1,1)),flip(a)].
220 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
221 x v (c(y v c(x)) v z) = x v z.  [para(151(a,1),16(a,1,1)),flip(a)].
223 x v c(y v (z v c(x))) = x.  [para(16(a,1),151(a,1,2,1,1))].
230 x v (y v c(y v x)) = 1.  [para(15(a,1),98(a,1,2,2,1))].
234 c(x) v (c(x v y) v z) = c(x) v z.  [para(133(a,1),16(a,1,1)),flip(a)].
236 c(x) v (y v c(x v z)) = y v c(x).  [para(133(a,1),89(a,1,2)),flip(a)].
257 c(x) v (c(y v x) v z) = c(x) v z.  [para(220(a,1),16(a,1,1)),flip(a)].
259 c(x v y) v c(y v c(x v y)) = c(y).
[para(220(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),220(11)])].
260 c(x) v (y v c(z v x)) = y v c(x).  [para(220(a,1),89(a,1,2)),flip(a)].
262 x v (y v (c(y v x) v z)) = 1.
[para(230(a,1),16(a,1,1)),rewrite([141(2),16(5)]),flip(a)].
305 x v (y v c(z v (u v c(x)))) = y v x.  [para(223(a,1),89(a,1,2)),flip(a)].
313 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),210(a,1,2,2,2,1))].
326 c(c(x) v y) v (z v x) = z v x.
[para(146(a,1),110(a,1,2,2,1,2,2,1,2)),rewrite([215(10),165(9),146(9)])].
665 x v (y v (c(z v c(x)) v u)) = y v (x v u).  [para(221(a,1),89(a,1,2)),flip(a)].
741 x v c(x v c(y v (x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),196(8)])].
855 c(x) v (y v (c(z v x) v u)) = c(x) v (y v u).
[para(326(a,1),215(a,1,2,1,1)),rewrite([16(6),16(9)])].
```

```
935 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).  [para(234(a,1),89(a,1,2)),flip(a)].
1409 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),260(a,1,2)),rewrite([16(9)])].
3701 x v c(y v c(x v y)) = x.
[para(741(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
3711 x v c(y v c(y v x)) = x.  [para(15(a,1),3701(a,1,2,1,2,1))].
3720 x v c(y v x) = x v c(y).
[para(3701(a,1),151(a,1,2,1)),rewrite([15(5),236(5)]),flip(a)].
3723 x v c(x v c(y)) = x v y.
[para(3701(a,1),109(a,1,2)),rewrite([3720(3),3720(7),3720(10),14(8),305(10)])].
3993 c(x v y) v c(y v c(x)) = c(y).  [back_rewrite(259),rewrite([3720(5)])].
3994 x v (y v c(x v c(z))) = y v (z v x).  [back_rewrite(165),rewrite([3720(3)])].
3996 c(c(c2 v c3) v c(c1 v (c3 v c(c2)))) != c3 v c(c(c1) v c(c2)) # answer("Foulis-
Holland Theorem 3.2").
[back_rewrite(78),rewrite([3723(7),3723(17),15(15),3994(16),15(11),15(12),16(12),3720(11)
,3720(21),15(18),3720(29),15(26),3720(30),14(22),15(21)])].
4094 x v c(x v y) = x v c(y).
[para(3711(a,1),151(a,1,2,1)),rewrite([15(5),260(5)]),flip(a)].
4095 x v (c(y v x) v z) = x v (c(y) v z).
[para(207(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),935(6)]),flip(a)].
4100 x v (c(x v y) v z) = c(y) v (x v z).
[para(262(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),855(6)]),flip(a)].
4101 x v (y v c(x v z)) = x v (y v c(z)).
[para(313(a,1),3711(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1409(6)]),flip(a)].
4104 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3711(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),260(8),15(4),133(4)]),flip(
a)].
4105 c(x v y) v c(c(x) v y) = c(y).
[para(126(a,1),3711(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),260(8),15(4),220(4)]),flip(
a)].
4340 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3720(a,1,2,1))].
4344 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3720(a,1),257(a,1,2)),rewrite([257(6)]),flip(a)].
4424 c(c(x v y) v z) = c(c(x) v z) v c(c(y) v (x v z)).
[para(234(a,1),3993(a,1,1,1)),rewrite([14(8),15(7),4100(7)]),flip(a)].
4512 c(c2 v c(c3)) v c(c(c2) v c(c1 v c3)) != c3 v c(c(c1) v c(c2)) # answer("Foulis-
Holland Theorem 3.2").
[back_rewrite(3996),rewrite([4424(13),4340(10),223(19),15(12),15(14)])].
4871 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4095(a,1),89(a,1,2)),flip(a)].
5754 c(x v y) v c(y v c(z v x)) = c(y).
[para(4340(a,1),4344(a,1,2,1)),rewrite([15(11),220(11)])].
5779 c(x v c(y v z)) v c(z v x) = c(x).  [para(5754(a,1),15(a,1)),flip(a)].
5899 c(x v c(y v z)) v c(y v x) = c(x).  [para(15(a,1),5779(a,1,1,1,2,1))].
6092 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5899(a,1,1,1))].
6102 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4101(a,1),5899(a,1,2,1)),rewrite([16(6)])].
6196 c(x v (c(y v z) v u)) = c(x v (z v (c(y) v u))) v c(x v (c(z) v u)).
[para(665(a,1),4104(a,1,2,1)),rewrite([14(2),15(5),4871(5),14(12)]),flip(a)].
6230 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6102),rewrite([6196(7),133(4),16(16),6092(15)]),flip(a)].
6561 $F # answer("Foulis-Holland Theorem 3.2").
[back_rewrite(4512),rewrite([6230(13),15(12),16(12),89(21),4105(20),14(11),15(10),4094(10
)]),xx(a)].

============================== end of proof ==========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH3.2 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. Note that some of "logical" proof lines in the above have been transformed to two text lines, with the *derivation* appearing on a text line following a text line containing the first part of that logical line. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 4.5 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

1. FH3.2 is derivable from OML.

2. The proof in Section 3.0 is, as far as I know, novel.

3. Companion papers show that the other three Dishkant-implication restricted FH theorems are also derivable from OML.

4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Dishkant-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Dishkant-implication-restricted FH theorems is the subject of future work.

5.0 Acknowledgements

6.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4*. URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH3.2 from OML. 2013. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I*. Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics*. Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Dishkant H. Semantics of the minimal logic of quantum mechanics. *Studia Logica* 30 (1972), 17-29.

[23] Horner JK. See *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Dishkant-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 4

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Dishkant" implication (one of the QL implication-connectives), from OML theory.

Keywords: Dishkant implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Dishkant-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
     x = c(c(x))                    # label("AxL1").
     x v y = y v x                  # label("AxL2").
     (x v y) v z = x v (y v z)      # label("AxL3").
     (x ^ y) ^ z = x ^ (y ^ z)      # label("AxL4").
     x v (x ^ y) = x                # label("AxL5").
     x ^ (x v y) = x                # label("AxL6").

% Ortholattice axioms
     c(x) ^ x = 0                   # label("AxOL1").
     c(x) v x = 1                   # label("AxOL2").
     x ^ y = c(c(x) v c(y))         # label("AxOL3").

% Definition of Dishkant implication ([22])
     i2(x,y) = c(c(y)) v (c(y) ^ c(x))
                                    # label("Df: i2").

% Definition of Dishkant-implication-restricted join
     u2(x,y) =  i2(c(x),y)          # label("Df: u2").

% Definition of Dishkant-implication-restricted meet
     int2(x,y) = c(i2(x,c(y)))      # label("Df: int2").

% Definition of x commutes with y
     C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
           # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Dishkant-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. c(x) denotes the orthocomplement of x. v denotes lattice join; ^ denotes lattice meet.

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
   x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)   (x v (y ^ z) =
        (x v y) ^ (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

      C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int2 and u2, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i2 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 2 denotes Dishkant-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH4.2, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH4.2 is a consequence of OML:

```
=============================== PROOF =================================

% Proof 1 at 3.60 (+ 0.11) seconds: "Foulis-Holland Theorem 4.1".
% Length of proof is 123.
% Level of proof is 16.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> u1(y,int1(x,z)) = int1(u1(y,x),u1(y,z)) # label("Foulis-Holland
Theorem 4.1") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
11 C(c1,c2) # label("Foulis-Holland Theorem 4.1").  [deny(4)].
12 C(c1,c3) # label("Foulis-Holland Theorem 4.1").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
```

```
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
35 u1(x,y) = i1(c(x),y) # label("Df: u1").  [assumption].
36 u1(x,y) = x v c(x v c(y)).  [copy(35),rewrite([26(3),14(3),14(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int1(u1(c2,c1),u1(c2,c3)) != u1(c2,int1(c1,c3)) # label("Foulis-Holland Theorem 4.1")
# answer("Foulis-Holland Theorem 4.1").  [deny(4)].
78 c(c(c2 v c(c2 v c(c1))) v c(c2 v (c(c2 v c(c3)) v c(c2 v c(c2 v c(c1)))))) != c2 v
c(c2 v (c(c1) v c(c3 v c(c1)))) # answer("Foulis-Holland Theorem 4.1").
[copy(77),rewrite([36(3),36(10),46(15),15(24),16(24),46(31),15(34),36(38),14(39)])].
79 (c1 ^ c2) v (c1 ^ c(c2)) = c1.  [resolve(11,a,10,a)].
80 c(c2 v c(c1)) v c(c(c1) v c(c2)) = c1.
[copy(79),rewrite([23(3),23(10),14(11),15(10),15(12)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
97 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
101 x v (c(y v x) v (c(c(y) v x) v z)) = 1_2 v z.
[para(68(a,1),16(a,1,1)),rewrite([16(9)]),flip(a)].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
110 x v (y v c(x v (y v c(z v (x v y))))) = z v (x v y).
[para(76(a,1),16(a,1)),rewrite([16(7)]),flip(a)].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
122 x v (c(y v x) v (c(c(y) v x) v z)) = 1 v z.  [back_rewrite(101),rewrite([113(9)])].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
138 c(x v y) v c(x v c(x v y)) = c(x).
[para(84(a,1),76(a,1,2,1,2)),rewrite([15(5),133(11)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
143 x v (c(y v x) v (c(c(y) v x) v z)) = 1.  [back_rewrite(122),rewrite([141(10)])].
144 x v (c(x) v y) = 1.  [back_rewrite(97),rewrite([141(5)])].
146 x v c(c(x) v y) = x.  [para(14(a,1),85(a,1,2,1,2))].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
165 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
166 c(c(c2 v c(c2 v c(c1))) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v
c(c3 v c(c1)))) # answer("Foulis-Holland Theorem 4.1").
[para(89(a,1),78(a,1,1,2,1)),rewrite([108(23),15(17),15(18),16(18)])].
194 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
196 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([194(5)])].
197 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
208 x v (y v (c(x v y) v z)) = 1.  [para(144(a,1),16(a,1)),flip(a)].
211 x v (y v (z v c(x v y))) = 1.  [para(196(a,1),16(a,1)),flip(a)].
216 x v (c(c(x) v y) v z) = x v z.  [para(146(a,1),16(a,1,1)),flip(a)].
218 x v (y v c(c(x) v z)) = y v x.  [para(146(a,1),89(a,1,2)),flip(a)].
221 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
222 x v (c(y v c(x)) v z) = x v z.  [para(151(a,1),16(a,1,1)),flip(a)].
224 x v c(y v (z v c(x))) = x.  [para(16(a,1),151(a,1,2,1))].
226 x v (y v c(z v c(x))) = y v x.  [para(151(a,1),89(a,1,2)),flip(a)].
231 x v (y v c(y v x)) = 1.  [para(15(a,1),98(a,1,2,2,1))].
```

```
235 c(x) v (c(x v y) v z) = c(x) v z.  [para(133(a,1),16(a,1,1)),flip(a)].
237 c(x) v (y v c(x v z)) = y v c(x).  [para(133(a,1),89(a,1,2)),flip(a)].
258 c(x) v (c(y v x) v z) = c(x) v z.  [para(221(a,1),16(a,1,1)),flip(a)].
260 c(x v y) v c(y v c(x v y)) = c(y).
[para(221(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),221(11)])].
261 c(x) v (y v c(z v x)) = y v c(x).  [para(221(a,1),89(a,1,2)),flip(a)].
263 x v (y v (c(y v x) v z)) = 1.
[para(231(a,1),16(a,1,1)),rewrite([141(2),16(5)]),flip(a)].
314 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),211(a,1,2,2,2,1))].
327 c(c(x) v y) v (z v x) = z v x.
[para(146(a,1),110(a,1,2,2,1,2,2,1,2)),rewrite([216(10),165(9),146(9)])].
666 x v (y v (c(z v c(x)) v u)) = y v (x v u).  [para(222(a,1),89(a,1,2)),flip(a)].
708 c2 v c(c2 v c(c1)) = c1 v c2.
[para(80(a,1),226(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
709 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),226(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
728 c(c(c1 v c2) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v c(c3 v
c(c1)))) # answer("Foulis-Holland Theorem 4.1").  [back_rewrite(166),rewrite([708(7)])].
743 x v c(x v c(y v c(x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),197(8)])].
773 c3 v c(c1 v c3) = c3 v c(c1).  [para(709(a,1),108(a,1,2,1)),rewrite([15(10)])].
857 c(x) v (y v (c(z v x) v u)) = c(x) v (y v u).
[para(327(a,1),216(a,1,2,1,1)),rewrite([16(6),16(9)])].
937 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).  [para(235(a,1),89(a,1,2)),flip(a)].
990 c(c1 v c3) v c(c3 v c(c1)) = c(c3).
[para(709(a,1),138(a,1,1,1)),rewrite([709(12),773(10)])].
1029 c(c1) v c(c3 v c(c1)) = c(c1) v c(c3).  [para(990(a,1),235(a,1,2)),flip(a)].
1030 c(c(c1 v c2) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 4.1").  [back_rewrite(728),rewrite([1029(27)])].
1169 c(x) v (y v (z v c(x v u))) = y v (z v c(x)).
[para(16(a,1),237(a,1,2)),rewrite([16(9)])].
1412 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),261(a,1,2)),rewrite([16(9)])].
3703 x v c(y v c(x v y)) = x.
[para(743(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
3713 x v c(y v c(y v x)) = x.  [para(15(a,1),3703(a,1,2,1,2,1))].
3722 x v c(y v x) = x v c(y).
[para(3703(a,1),151(a,1,2,1)),rewrite([15(5),237(5)]),flip(a)].
3995 c(x v y) v c(y v c(x)) = c(y).  [back_rewrite(260),rewrite([3722(5)])].
3996 x v (y v c(x v c(z))) = y v (z v x).  [back_rewrite(165),rewrite([3722(3)])].
4062 c(c(c1 v c2) v c(c2 v (c3 v c(c1)))) != c2 v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 4.1").
[back_rewrite(1030),rewrite([3996(14),15(9),15(10),16(10)])].
4096 x v c(x v y) = x v c(y).
[para(3713(a,1),151(a,1,2,1)),rewrite([15(5),261(5)]),flip(a)].
4097 x v (c(y v x) v z) = x v (c(y) v z).
[para(208(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),937(6)]),flip(a)].
4098 x v (y v c(z v x)) = x v (y v c(z)).
[para(211(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1169(6)]),flip(a)].
4102 x v (c(x v y) v z) = c(y) v (x v z).
[para(263(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),857(6)]),flip(a)].
4103 x v (y v c(x v z)) = x v (y v c(z)).
[para(314(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1412(6)]),flip(a)].
4106 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3713(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),261(8),15(4),133(4)]),flip(
a)].
4109 c(x v y) v (c(c(x) v y) v z) = c(y) v z.
[para(143(a,1),3713(a,1,2,1,2,1)),rewrite([83(9),150(9),15(9),857(9),258(5)]),flip(a)].
4128 c(c(c1 v c2) v c(c2 v (c3 v c(c1)))) != c2 v c(c(c1) v c(c3)) # answer("Foulis-
Holland Theorem 4.1").  [back_rewrite(4062),rewrite([4096(23)])].
4343 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3722(a,1,2,1))].
4347 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3722(a,1),258(a,1,2)),rewrite([258(6)]),flip(a)].
4379 x v c(y v (x v z)) = x v c(y v z).  [para(89(a,1),4096(a,1,2,1))].
4424 c(x v c(c(y) v z)) = c(x v y) v c(c(y) v (x v c(z))).
[para(218(a,1),3995(a,1,1,1)),rewrite([15(8),4103(8)]),flip(a)].
4427 c(c(x v y) v z) = c(c(x) v z) v c(c(y) v (x v z)).
[para(235(a,1),3995(a,1,1,1)),rewrite([14(8),15(7),4102(7)]),flip(a)].
4514 c(c1 v c(c2)) v c(c(c1) v c(c2 v c3)) != c2 v c(c(c1) v c(c3)) # answer("Foulis-
Holland Theorem 4.1").
[back_rewrite(4128),rewrite([4427(13),4343(10),224(19),15(12),15(14)])].
```

```
4571 c(x v c(y v z)) = c(z v (x v c(y))) v c(x v c(z)).
[para(226(a,1),4106(a,1,2,1)),rewrite([14(2),15(4),4098(4),14(10)]),flip(a)].
4874 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4097(a,1),89(a,1,2)),flip(a)].
5181 c(x v y) v c(c(y) v (x v z)) = c(x v y) v c(x v z).
[para(4102(a,1),4379(a,1,2,1))].
5194 c(x v c(c(y) v z)) = c(x v y) v c(x v c(z)).
[back_rewrite(4424),rewrite([5181(13)])].
5757 c(x v y) v c(y v c(z v x)) = c(y).
[para(4343(a,1),4347(a,1,2,1)),rewrite([15(11),221(11)])].
5782 c(x v c(y v z)) v c(z v x) = c(x).   [para(5757(a,1),15(a,1)),flip(a)].
5902 c(x v c(y v z)) v c(y v x) = c(x).   [para(15(a,1),5782(a,1,1,1,2,1))].
6095 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5902(a,1,1,1))].
6105 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4103(a,1),5902(a,1,2,1)),rewrite([16(6)])].
6199 c(x v (c(y v z) v u)) = c(x v (c(y) v u))) v c(x v (c(z) v u)).
[para(666(a,1),4106(a,1,2,1)),rewrite([14(2),15(5),4874(5),14(12)]),flip(a)].
6233 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6105),rewrite([6199(7),133(4),16(16),6095(15)]),flip(a)].
6535 c(x v (y v z)) v c(x v c(y)) = c(x v z) v c(x v c(y)).
[back_rewrite(5194),rewrite([6233(5),14(2)])].
6563 c(x v (y v c(z))) v c(y v c(x)) = c(y v c(z)) v c(y v c(x)).
[back_rewrite(4571),rewrite([6233(4),6535(8)]),flip(a)].
6564 $F # answer("Foulis-Holland Theorem 4.1").
[back_rewrite(4514),rewrite([6233(13),89(12),6563(20),4109(19),14(3)]),xx(a)].

============================= end of proof =========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH4.2 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. Note that some of "logical" proof lines in the above have been transformed to two text lines, with the *derivation* appearing on a text line following a text line containing the first part of that logical line. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 3.7 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

1. FH4.2 is derivable from OML.

2. The proof in Section 3.0 is, as far as I know, novel.

3. Companion papers show that the other three Dishkant-implication restricted FH theorems are also derivable from OML.

4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Dishkant-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Dishkant-implication-restricted FH theorems is the subject of future work.

5.0 Acknowledgements

whose passion for foundations of physics inspired those of us privileged to have known him. For any infelicities that remain, I am solely responsible.

6.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4*. URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH4.2 from OML. 2013. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I.* Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics*. Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Dishkant H. Semantics of the minimal logic of quantum mechanics. *Studia Logica* 30 (1972), 17-29.

[23] Horner JK. See *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Sasaki-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 1

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Sasaki" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Sasaki implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Sasaki-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
     x = c(c(x))                   # label("AxL1").
     x v y = y v x                 # label("AxL2").
     (x v y) v z = x v (y v z)     # label("AxL3").
     (x ^ y) ^ z = x ^ (y ^ z)     # label("AxL4").
     x v (x ^ y) = x               # label("AxL5").
     x ^ (x v y) = x               # label("AxL6").

% Ortholattice axioms
     c(x) ^ x = 0                  # label("AxOL1").
     c(x) v x = 1                  # label("AxOL2").
     x ^ y = c(c(x) v c(y))        # label("AxOL3").

% Definition of Sasaki implication ([22])
     i1(x,y) = c(x) v (x ^ y)      # label("Df: i1").

% Definition of Sasaki-implication-restricted join
     u1(x,y) =  i1(c(x),y)         # label("Df: u1").

% Definition of Sasaki-implication-restricted meet
     int1(x,y) = c(i1(x,c(y)))     # label("Df: int1").

% Definition of x commutes with y
     C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
            # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Sasaki-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. c(x) denotes the orthocomplement of x. v denotes lattice join; ^ denotes lattice meet.

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
   x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)   (x v (y ^ z) =
     (x v y) ^ (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

      C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int1 and u1, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i1 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 1 denotes Sasaki-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH1.1, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH1.1 is a consequence of OML:

```
============================= PROOF =================================

% Proof 1 at 2.37 (+ 0.06) seconds: "Foulis-Holland Theorem 1.1".
% Length of proof is 84.
% Level of proof is 13.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> int1(x,u1(y,z)) = u1(int1(x,y),int1(x,z)) # label("Foulis-Holland
Theorem 1.1") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
11 C(c1,c2) # label("Foulis-Holland Theorem 1.1").  [deny(4)].
12 C(c1,c3) # label("Foulis-Holland Theorem 1.1").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
```

```
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
35 u1(x,y) = i1(c(x),y) # label("Df: u1").  [assumption].
36 u1(x,y) = x v c(x v c(y)).  [copy(35),rewrite([26(3),14(3),14(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int1(c1,u1(c2,c3)) != u1(int1(c1,c2),int1(c1,c3)) # label("Foulis-Holland Theorem
1.1") # answer("Foulis-Holland Theorem 1.1").  [deny(4)].
78 c(c(c1) v c(c(c1) v (c2 v c(c2 v c(c3))))) != c(c(c1) v c(c2 v c(c1))) v c(c(c1) v
(c(c3 v c(c1)) v c(c(c1) v c(c2 v c(c1))))) # answer("Foulis-Holland Theorem 1.1").
[copy(77),rewrite([36(4),46(9),46(18),15(21),46(27),15(30),36(34),14(43),15(42),16(42)])]
.
79 (c1 ^ c2) v (c1 ^ c(c2)) = c1.  [resolve(11,a,10,a)].
80 c(c2 v c(c1)) v c(c(c1) v c(c2)) = c1.
[copy(79),rewrite([23(3),23(10),14(11),15(10),15(12)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
90 c(c(c1) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c(c(c1) v c(c2 v c(c1))) v c(c(c1) v
(c(c3 v c(c1)) v c(c(c1) v c(c2 v c(c1))))) # answer("Foulis-Holland Theorem 1.1").
[back_rewrite(78),rewrite([89(12)])].
99 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
100 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
109 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
114 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
125 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(100),rewrite([114(8)])].
134 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
138 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
139 c(x v y) v c(x v c(x v y)) = c(x).
[para(84(a,1),76(a,1,2,1,2)),rewrite([15(5),134(11)])].
142 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([138(6)])].
151 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
152 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
164 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
166 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
169 c(c(c1) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c(c(c1) v c(c2 v c(c1))) v c(c2 v
(c(c1) v c(c3 v c(c1)))) # answer("Foulis-Holland Theorem 1.1").
[back_rewrite(90),rewrite([166(42),15(34),16(34)])].
195 x v 1 = 1.  [para(142(a,1),15(a,1)),flip(a)].
197 x v (y v c(x)) = 1.  [back_rewrite(164),rewrite([195(5)])].
198 0 v x = x.  [para(151(a,1),15(a,1)),flip(a)].
212 x v (y v (z v c(x v y))) = 1.  [para(197(a,1),16(a,1)),flip(a)].
222 c(x) v c(y v x) = c(x).  [para(14(a,1),152(a,1,2,1,2))].
227 x v (y v c(z v c(x))) = y v x.  [para(152(a,1),89(a,1,2)),flip(a)].
236 c(x) v (c(x v y) v z) = c(x) v z.  [para(134(a,1),16(a,1,1)),flip(a)].
238 x v (y v c(x v z)) = y v c(x).  [para(134(a,1),89(a,1,2)),flip(a)].
261 c(x v y) v c(y v c(x v y)) = c(y).
[para(222(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),222(11)])].
262 c(x) v (y v c(z v x)) = y v c(x).  [para(222(a,1),89(a,1,2)),flip(a)].
315 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),212(a,1,2,2,2,1))].
709 c2 v c(c2 v c(c1)) = c1 v c2.
[para(80(a,1),227(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
710 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),227(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
732 c2 v c(c1 v c2) = c2 v c(c1).  [para(709(a,1),109(a,1,2,1)),rewrite([15(10)])].
```

```
743 x v c(x v c(y v c(x v y))) = 1.
[para(99(a,1),125(a,1,2,1,1)),rewrite([83(2),15(6),198(8)])].
773 c3 v c(c1 v c3) = c3 v c(c1).  [para(710(a,1),109(a,1,2,1)),rewrite([15(10)])].
989 c(c1 v c2) v c(c2 v c(c1)) = c(c2).
[para(709(a,1),139(a,1,1,1)),rewrite([709(12),732(10)])].
990 c(c1 v c3) v c(c3 v c(c1)) = c(c3).
[para(710(a,1),139(a,1,1,1)),rewrite([710(12),773(10)])].
1002 c(c1) v c(c2 v c(c1)) = c(c1) v c(c2).  [para(989(a,1),236(a,1,2)),flip(a)].
1003 c(c(c1) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c(c(c1) v c(c2)) v c(c2 v (c(c1) v
c(c3 v c(c1)))) # answer("Foulis-Holland Theorem 1.1").
[back_rewrite(169),rewrite([1002(23)])].
1030 c(c1) v c(c3 v c(c1)) = c(c1) v c(c3).  [para(990(a,1),236(a,1,2)),flip(a)].
1031 c(c(c1) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c(c(c1) v c(c2)) v c(c2 v (c(c1) v
c(c3))) # answer("Foulis-Holland Theorem 1.1").
[back_rewrite(1003),rewrite([1030(30)])].
1413 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),262(a,1,2)),rewrite([16(9)])].
3704 x v c(y v c(x v y)) = x.
[para(743(a,1),109(a,1,2,1)),rewrite([83(2),151(2),15(5)]),flip(a)].
3714 x v c(y v c(y v x)) = x.  [para(15(a,1),3704(a,1,2,1,2,1))].
3723 x v c(y v x) = x v c(y).
[para(3704(a,1),152(a,1,2,1)),rewrite([15(5),238(5)]),flip(a)].
3996 c(x v y) v c(y v c(x)) = c(y).  [back_rewrite(261),rewrite([3723(5)])].
3997 x v (y v c(y v c(z))) = y v (z v x).  [back_rewrite(166),rewrite([3723(3)])].
4063 c(c(c1) v c(c2 v (c3 v c(c1)))) != c(c(c1) v c(c2)) v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 1.1").
[back_rewrite(1031),rewrite([3997(12),15(7),15(8),16(8)])].
4104 x v (y v c(x v z)) = x v (y v c(z)).
[para(315(a,1),3714(a,1,2,1,2,1)),rewrite([83(6),151(6),15(6),1413(6)]),flip(a)].
4343 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3723(a,1,2,1))].
4351 c(c(c1) v c(c2 v c3)) != c(c(c1) v c(c2)) v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 1.1").  [back_rewrite(4063),rewrite([4343(10)])].
4429 c(x v c(y v z)) = c(x v c(y)) v c(y v (x v c(z))).
[para(238(a,1),3996(a,1,1,1)),rewrite([14(8),15(7),4104(7)]),flip(a)].
4430 $F # answer("Foulis-Holland Theorem 1.1").  [resolve(4429,a,4351,a)].

============================== end of proof ==========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH1.1 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. Note that some of "logical" proof lines in the above have been transformed to two text lines, with the *derivation* appearing on a text line following a text line containing the first part of that logical line. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 2.5 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

1. FH1.1 is derivable from OML.

2. The proof in Section 3.0 is, as far as I know, novel.

3. Companion papers show that the other three Sasaki-implication restricted FH theorems are also derivable from OML.

4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Sasaki-implication-restricted

variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Sasaki-implication-restricted FH theorems is the subject of future work.

5.0 Acknowledgements

This work benefited from discussions with Tom Oberdan, Frank Pecchioni, Tony Pawlicki, and the late John K. Prentice, whose passion for foundations of physics inspired those of us privileged to have known him. For any infelicities that remain, I am solely responsible.

6.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4.* URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH1.1 from OML. 2011. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I.* Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr.

Bulletin of the American Mathematical Society 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics.* Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient

compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Sasaki U. Orthocomplemented lattices satisfying the exchange axiom. *Journal of Science Hiroshima University A* 17 (1964), 293-302.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Sasaki-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 2

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Sasaki" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Sasaki implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Sasaki-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
     x = c(c(x))                    # label("AxL1").
     x v y = y v x                  # label("AxL2").
     (x v y) v z = x v (y v z)      # label("AxL3").
     (x ^ y) ^ z = x ^ (y ^ z)      # label("AxL4").
     x v (x ^ y) = x                # label("AxL5").
     x ^ (x v y) = x                # label("AxL6").

% Ortholattice axioms
     c(x) ^ x = 0                   # label("AxOL1").
     c(x) v x = 1                   # label("AxOL2").
     x ^ y = c(c(x) v c(y))         # label("AxOL3").

% Definition of Sasaki implication ([22])
     i1(x,y) = c(x) v (x ^ y)       # label("Df: i1").

% Definition of Sasaki-implication-restricted join
   u1(x,y) =  i1(c(x),y)           # label("Df: u1").

% Definition of Sasaki-implication-restricted meet
   int1(x,y) = c(i1(x,c(y)))       # label("Df: int1").

% Definition of x commutes with y
     C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
           # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Sasaki-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. "c(x)" denotes the orthocomplement of x. "v" denotes lattice join; "^" denotes lattice meet. "=" denotes identity; "<->" denotes "if and only if"

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
    x v y     (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)    (x v (y ^ z) = (x v y) ^
        (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


  where C(x,y), "x commutes with y" is defined as

        C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication- restricted meet and join (e.g., int1 and u1, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i1 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 1 denotes Sasaki-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16], [21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH2.1, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista* *Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH2.1 is a consequence of OML:

```
============================= PROOF =================================

% Proof 1 at 17.32 (+ 0.27) seconds: "Foulis-Holland Theorem 2.1".
% Length of proof is 134.
% Level of proof is 17.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> int1(y,u1(x,z)) = u1(int1(y,x),int1(y,z)) # label("Foulis-Holland
Theorem 2.1") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
11 C(c1,c2) # label("Foulis-Holland Theorem 2.1").  [deny(4)].
12 C(c1,c3) # label("Foulis-Holland Theorem 2.1").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
```

```
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
35 u1(x,y) = i1(c(x),y) # label("Df: u1").  [assumption].
36 u1(x,y) = x v c(x v c(y)).  [copy(35),rewrite([26(3),14(3),14(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int1(c2,u1(c1,c3)) != u1(int1(c2,c1),int1(c2,c3)) # label("Foulis-Holland Theorem
2.1") # answer("Foulis-Holland Theorem 2.1").  [deny(4)].
78 c(c(c2) v c(c(c2) v (c1 v c(c1 v c(c3))))) != c(c(c2) v c(c1 v c(c2))) v c(c(c2) v
(c(c3 v c(c2)) v c(c(c2) v c(c1 v c(c2))))) # answer("Foulis-Holland Theorem 2.1").
[copy(77),rewrite([36(4),46(9),46(18),15(21),46(27),15(30),36(34),14(43),15(42),16(42)])]
.
79 (c1 ^ c2) v (c1 ^ c(c2)) = c1.  [resolve(11,a,10,a)].
80 c(c2 v c(c1)) v c(c(c1) v c(c2)) = c1.
[copy(79),rewrite([23(3),23(10),14(11),15(10),15(12)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
90 c(c(c2) v c(c1 v (c(c2) v c(c1 v c(c3))))) != c(c(c2) v c(c1 v c(c2))) v c(c(c2) v
(c(c3 v c(c2)) v c(c(c2) v c(c1 v c(c2))))) # answer("Foulis-Holland Theorem 2.1").
[back_rewrite(78),rewrite([89(12)])].
98 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
99 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
100 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
102 x v (c(y v x) v (c(c(y) v x) v z)) = 1_2 v z.
[para(68(a,1),16(a,1,1)),rewrite([16(9)]),flip(a)].
109 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
111 x v (y v c(x v (y v c(z v (x v y))))) = z v (x v y).
[para(76(a,1),16(a,1,1)),rewrite([16(7)]),flip(a)].
114 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
123 x v (c(y v x) v (c(c(y) v x) v z)) = 1 v z.  [back_rewrite(102),rewrite([114(9)])].
125 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(100),rewrite([114(8)])].
134 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
138 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
139 c(x v y) v c(x v c(x v y)) = c(x).
[para(84(a,1),76(a,1,2,1,2)),rewrite([15(5),134(11)])].
142 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([138(6)])].
144 x v (c(y v x) v (c(c(y) v x) v z)) = 1.  [back_rewrite(123),rewrite([142(10)])].
145 x v (c(x) v y) = 1.  [back_rewrite(98),rewrite([142(5)])].
147 x v c(c(x) v y) = x.  [para(14(a,1),85(a,1,2,1,2))].
151 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
152 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
164 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
166 x v (y v c(z v x)) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
169 c(c(c2) v c(c1 v (c(c2) v c(c1 v c(c3))))) != c(c(c2) v c(c1 v c(c2))) v c(c1 v
(c(c2) v c(c3 v c(c2)))) # answer("Foulis-Holland Theorem 2.1").
[back_rewrite(90),rewrite([166(42),15(34),16(34)])].
195 x v 1 = 1.  [para(142(a,1),15(a,1)),flip(a)].
197 x v (y v c(x)) = 1.  [back_rewrite(164),rewrite([195(5)])].
198 0 v x = x.  [para(151(a,1),15(a,1)),flip(a)].
209 x v (y v (c(x v y) v z)) = 1.  [para(145(a,1),16(a,1)),flip(a)].
211 c(x) v (y v x) = 1.  [para(14(a,1),197(a,1,2,2))].
212 x v (y v (z v c(x v y))) = 1.  [para(197(a,1),16(a,1)),flip(a)].
216 x v (c(x v y) v (z v y)) = 1.  [para(89(a,1),211(a,1,2)),rewrite([89(5)])].
217 x v (c(c(x) v y) v z) = x v z.  [para(147(a,1),16(a,1,1)),flip(a)].
```

```
222 c(x) v c(y v x) = c(x).  [para(14(a,1),152(a,1,2,1,2))].
223 x v (c(y v c(x)) v z) = x v z.  [para(152(a,1),16(a,1,1)),flip(a)].
227 x v (y v c(z v c(x))) = y v x.  [para(152(a,1),89(a,1,2)),flip(a)].
236 c(x) v (c(x v y) v z) = c(x) v z.  [para(134(a,1),16(a,1,1)),flip(a)].
237 c(x v y) v c(x v (y v z)) = c(x v y).  [para(16(a,1),134(a,1,2,1))].
238 c(x) v (y v c(x v z)) = y v c(x).  [para(134(a,1),89(a,1,2)),flip(a)].
259 c(x) v (c(y v x) v z) = c(x) v z.  [para(222(a,1),16(a,1,1)),flip(a)].
261 c(x v y) v c(y v c(x v y)) = c(y).
[para(222(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),222(11)])].
262 c(x) v (y v c(z v x)) = y v c(x).  [para(222(a,1),89(a,1,2)),flip(a)].
315 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),212(a,1,2,2,2,1))].
328 c(c(x) v y) v (z v x) = z v x.
[para(147(a,1),111(a,1,2,2,1,2,2,1,2)),rewrite([217(10),166(9),147(9)])].
338 x v (c(y v x) v (z v y)) = 1.  [para(15(a,1),216(a,1,2,1,1))].
667 x v (y v (c(z v c(x)) v u)) = y v (x v u).  [para(223(a,1),89(a,1,2)),flip(a)].
709 c2 v c(c2 v c(c1)) = c1 v c2.
[para(80(a,1),227(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
710 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),227(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
732 c2 v c(c1 v c2) = c2 v c(c1).  [para(709(a,1),109(a,1,2,1)),rewrite([15(10)])].
743 x v c(x v c(y v c(x v y))) = 1.
[para(99(a,1),125(a,1,2,1,1)),rewrite([83(2),15(6),198(8)])].
773 c3 v c(c1 v c3) = c3 v c(c1).  [para(710(a,1),109(a,1,2,1)),rewrite([15(10)])].
857 c(x) v (y v (c(z v x) v u)) = c(x) v (y v u).
[para(328(a,1),217(a,1,2,1,1)),rewrite([16(6),16(9)])].
933 c(x v y) v (z v c(c(x v y) v (z v c(c(x) v z)))) = c(x) v z.
[para(236(a,1),76(a,1,2,1,2,1)),rewrite([16(10),16(12),236(17)])].
937 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).  [para(236(a,1),89(a,1,2)),flip(a)].
989 c(c1 v c2) v c(c2 v c(c1)) = c(c2).
[para(709(a,1),139(a,1,1,1)),rewrite([709(12),732(10)])].
990 c(c1 v c3) v c(c3 v c(c1)) = c(c3).
[para(710(a,1),139(a,1,1,1)),rewrite([710(12),773(10)])].
1000 c1 v c(c1 v c2) = c1 v c(c2).
[para(989(a,1),227(a,1,2)),rewrite([15(10)]),flip(a)].
1005 c1 v c(c1 v c(c2)) = c1 v c2.  [para(1000(a,1),109(a,1,2,1)),rewrite([15(10)])].
1014 c(c1 v c2) v c(c1 v c(c2)) = c(c1).
[para(1000(a,1),139(a,1,1,1)),rewrite([1000(12),1005(12),15(10)])].
1027 c1 v c(c1 v c3) = c1 v c(c3).
[para(990(a,1),227(a,1,2)),rewrite([15(10)]),flip(a)].
1089 c1 v c(c1 v c(c3)) = c1 v c3.  [para(1027(a,1),109(a,1,2,1)),rewrite([15(10)])].
1100 c1 v (x v c(c1 v c(c3))) = x v (c1 v c3).  [para(1089(a,1),89(a,1,2)),flip(a)].
1107 c(c(c2) v c(c1 v (c3 v c(c2)))) != c(c(c2) v c(c1 v c(c2))) v c(c1 v (c(c2) v c(c3 v
c(c2)))) # answer("Foulis-Holland Theorem 2.1").
[back_rewrite(169),rewrite([1100(12),15(8),16(8)])].
1401 c(c2) v c(c1 v c(c2)) = c(c1) v c(c2).
[para(1014(a,1),259(a,1,2)),rewrite([15(5)]),flip(a)].
1412 c(c(c2) v c(c1 v (c3 v c(c2)))) != c(c(c1) v c(c2)) v c(c1 v (c(c2) v c(c3 v
c(c2)))) # answer("Foulis-Holland Theorem 2.1").
[back_rewrite(1107),rewrite([1401(19)])].
1413 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),262(a,1,2)),rewrite([16(9)])].
3704 x v c(y v c(x v y)) = x.
[para(743(a,1),109(a,1,2,1)),rewrite([83(2),151(2),15(5)]),flip(a)].
3714 x v c(y v c(y v x)) = x.  [para(15(a,1),3704(a,1,2,1,2,1))].
3723 x v c(y v x) = x v c(y).
[para(3704(a,1),152(a,1,2,1)),rewrite([15(5),238(5)]),flip(a)].
3935 c(c(c2) v c(c1 v (c3 v c(c2)))) != c(c(c1) v c(c2)) v c(c1 v (c(c2) v c(c3))) #
answer("Foulis-Holland Theorem 2.1").  [back_rewrite(1412),rewrite([3723(26)])].
3956 c(x v y) v (z v c(c(x v y) v (z v x))) = c(x) v z.
[back_rewrite(933),rewrite([3723(8),14(6)])].
3997 c(x v y) v c(y v c(x)) = c(y).  [back_rewrite(261),rewrite([3723(5)])].
4097 x v c(x v y) = x v c(y).
[para(3714(a,1),152(a,1,2,1)),rewrite([15(5),262(5)]),flip(a)].
4098 x v (c(y v x) v z) = x v (c(y) v z).
[para(209(a,1),3714(a,1,2,1,2,1)),rewrite([83(6),151(6),15(6),937(6)]),flip(a)].
4104 x v (y v c(x v z)) = x v (y v c(z)).
[para(315(a,1),3714(a,1,2,1,2,1)),rewrite([83(6),151(6),15(6),1413(6)]),flip(a)].
4105 c(x v y) v (z v x) = c(y) v (z v x).
[para(338(a,1),3714(a,1,2,1,2,1)),rewrite([83(6),151(6),15(6),259(6)]),flip(a)].
```

```
4107 c(x v y) v c(c(y) v x) = c(x).
[para(125(a,1),3714(a,1,2,1,2,1)),rewrite([83(8),151(8),15(8),262(8),15(4),134(4)]),flip(
a)].
4110 c(x v y) v (c(c(x) v y) v z) = c(y) v z.
[para(144(a,1),3714(a,1,2,1,2,1)),rewrite([83(9),151(9),15(9),857(9),259(5)]),flip(a)].
4227 c(x v y) v (z v c(c(y) v (z v x))) = c(x) v z.
[back_rewrite(3956),rewrite([4105(6)])].
4343 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3723(a,1,2,1))].
4347 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3723(a,1),259(a,1,2)),rewrite([259(6)]),flip(a)].
4351 c(c(c2) v c(c1 v c3)) != c(c(c1) v c(c2)) v c(c1 v (c(c2) v c(c3))) #
answer("Foulis-Holland Theorem 2.1").  [back_rewrite(3935),rewrite([4343(10)])].
4366 c(x v y) v c(y v (z v x)) = c(x v y).
[para(15(a,1),237(a,1,2,1)),rewrite([16(4)])].
4380 x v c(y v (x v z)) = x v c(y v z).  [para(89(a,1),4097(a,1,2,1))].
4388 c(x v y) v (z v c(c(y) v x)) = c(x) v z.  [back_rewrite(4227),rewrite([4380(7)])].
4422 c(x v (y v z)) v c(y v (z v c(x))) = c(y v z).  [para(16(a,1),3997(a,1,2,1))].
4874 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4098(a,1),89(a,1,2)),flip(a)].
5757 c(x v y) v c(y v c(z v x)) = c(y).
[para(4343(a,1),4347(a,1,2,1)),rewrite([15(11),222(11)])].
5782 c(x v c(y v z)) v c(z v x) = c(x).  [para(5757(a,1),15(a,1)),flip(a)].
5902 c(x v c(y v z)) v c(y v x) = c(x).  [para(15(a,1),5782(a,1,1,1,2,1))].
6095 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5902(a,1,1,1))].
6105 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4104(a,1),5902(a,1,2,1)),rewrite([16(6)])].
6114 c(x v (y v z)) v c(z v x) = c(z v x).  [para(4366(a,1),15(a,1)),flip(a)].
6199 c(x v (c(y v z) v u)) = c(x v (z v (c(y) v u))) v c(x v (c(z) v u)).
[para(667(a,1),4107(a,1,2,1)),rewrite([14(2),15(5),4874(5),14(12)]),flip(a)].
6233 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6105),rewrite([6199(7),134(4),16(16),6095(15)]),flip(a)].
6568 c(c(c1) v c(c2)) v c(c1 v (c(c2) v c(c3))) != c(c3 v (c(c1) v c(c2))) v c(c(c2) v
c(c3)) # answer("Foulis-Holland Theorem 2.1").
[back_rewrite(4351),rewrite([6233(8),15(7),16(7)]),flip(a)].
7296 c(x v (y v z)) v c(z v c(x)) = c(y v z) v c(z v c(x)).
[para(6114(a,1),4110(a,1,2))].
7310 c(c(c1) v c(c2)) v c(c1 v (c(c2) v c(c3))) != c(c(c1) v c(c2)) v c(c(c2) v c(c3)) #
answer("Foulis-Holland Theorem 2.1").  [back_rewrite(6568),rewrite([7296(30)])].
7522 c(x v (y v z)) v c(y v c(z)) = c(x v y) v c(y v c(z)).
[para(4366(a,1),4388(a,1,2)),rewrite([16(2)])].
10933 c(x v (y v z)) v c(z v (c(x) v y)) = c(y v z).
[para(15(a,1),4422(a,1,2,1)),rewrite([16(6)])].
16826 c(c(x) v y) v c(x v (y v c(z))) = c(c(x) v y) v c(y v c(z)).
[para(10933(a,1),4388(a,1,2)),rewrite([16(3),7522(8)]),flip(a)].
16827 $F # answer("Foulis-Holland Theorem 2.1").  [resolve(16826,a,7310,a)].

============================== end of proof =========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH2.1 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 17.5 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

   1. FH2.1 is derivable from OML.

   2. The proof in Section 3.0 is, as far as I know, novel.

   3. Companion papers show that the other three Sasaki-implication restricted FH theorems are also derivable from OML.

   4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Sasaki-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Sasaki-implication-restricted FH theorems, is the subject of future work.

5.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4*. URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH2.1 from OML. 2011. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.
[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I*. Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

 [12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics*. Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner .pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Sasaki U. Orthocomplemented lattices satisfying the exchange axiom. *Journal of Science Hiroshima University A* 17 (1964), 293-302.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Co*

An Automated Deduction of a "Sasaki-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 3

Jack K. Horner
P. O. Box 266
Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Sasaki" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Sasaki implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Sasaki-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
     x = c(c(x))                # label("AxL1").
     x v y = y v x              # label("AxL2").
     (x v y) v z = x v (y v z)  # label("AxL3").
     (x ^ y) ^ z = x ^ (y ^ z)  # label("AxL4").
     x v (x ^ y) = x            # label("AxL5").
     x ^ (x v y) = x            # label("AxL6").

% Ortholattice axioms
     c(x) ^ x = 0               # label("AxOL1").
     c(x) v x = 1               # label("AxOL2").
     x ^ y = c(c(x) v c(y))     # label("AxOL3").

% Definition of Sasaki implication ([22])
     i1(x,y) = c(x) v (x ^ y)   # label("Df: i1").

% Definition of Sasaki-implication-restricted join
     u1(x,y) =  i1(c(x),y)      # label("Df: u1").

% Definition of Sasaki-implication-restricted meet
     int1(x,y) = c(i1(x,c(y)))  # label("Df: int1").

% Definition of x commutes with y
     C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
          # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Sasaki-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. "c(x)" denotes the orthocomplement of x. "v" denotes lattice join; "^" denotes lattice meet. "=" denotes identity; "<->" denotes "if and only if".

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
    x v y     (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)     (x v (y ^ z) = (x v y) ^
           (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) -> ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) -> ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) -> ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) -> ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


 where C(x,y), "x commutes with y" is defined as

        C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int1 and u1, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i1 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where N = 1 denotes Sasaki-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH3.1, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH3.1 is a consequence of OML:

```
============================== PROOF =================================

% Proof 1 at 3.81 (+ 0.05) seconds: "Foulis-Holland Theorem 3.1".
% Length of proof is 109.
% Level of proof is 16.

4 C(Cx,y) & C(X,z) -> u1(x,int1(y,z)) = int1(u1(x,y),u1(x,z)) # label("Foulis-Holland
Theorem 3.1") # label(non_clause) # label(goal).  [goal].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
15 x v y = y v x # label("AxL2").  [assumption].
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
```

```
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
35 u1(x,y) = i1(c(x),y) # label("Df: u1").  [assumption].
36 u1(x,y) = x v c(x v c(y)).  [copy(35),rewrite([26(3),14(3),14(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int1(u1(c3,c1),u1(c3,c2)) != u1(c3,int1(c1,c2)) # label("Foulis-Holland Theorem 3.1")
# answer("Foulis-Holland Theorem 3.1").  [deny(4)].
78 c(c(c(c3 v c(c3 v c(c1))) v c(c3 v (c(c3 v c(c2)) v c(c3 v c(c3 v c(c1)))))) != c3 v
c(c(c1) v (c(c1) v c(c2 v c(c1)))) # answer("Foulis-Holland Theorem 3.1").
[copy(77),rewrite([36(3),36(10),46(15),15(24),16(24),46(31),15(34),36(38),14(39)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
97 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
109 x v (c(x v c(y v x)) v z) = y v (x v z).
[para(76(a,1),16(a,1,1)),rewrite([16(2)]),flip(a)].
110 x v (y v c(x v (y v c(z v (x v y))))) = z v (x v y).
[para(76(a,1),16(a,1)),rewrite([16(7)]),flip(a)].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
126 x v (c(y v x) v c(c(y) v x)) = 1.  [back_rewrite(68),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
144 x v (c(x) v y) = 1.  [back_rewrite(97),rewrite([141(5)])].
146 x v c(c(x) v y) = x.  [para(14(a,1),85(a,1,2,1,2))].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
165 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
166 c(c(c3 v c(c3 v c(c1))) v c(c3 v (c(c1) v c(c3 v c(c2))))) != c3 v c(c3 v (c(c1) v
c(c2 v c(c1)))) # answer("Foulis-Holland Theorem 3.1").
[para(89(a,1),78(a,1,1,2,1)),rewrite([108(23),15(17),15(18),16(18)])].
194 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
196 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([194(5)])].
197 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
208 x v (y v (c(x v y) v z)) = 1.  [para(144(a,1),16(a,1)),flip(a)].
211 x v (y v (z v c(x v y))) = 1.  [para(196(a,1),16(a,1)),flip(a)].
216 x v (c(c(x) v y) v z) = x v z.  [para(146(a,1),16(a,1,1)),flip(a)].
218 x v (y v c(c(x) v z)) = y v x.  [para(146(a,1),89(a,1,2)),flip(a)].
221 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
222 x v (c(y v c(x)) v z) = x v z.  [para(151(a,1),16(a,1,1)),flip(a)].
224 x v c(y v (z v c(x))) = x.  [para(16(a,1),151(a,1,2,1))].
226 x v (y v c(z v c(x))) = y v x.  [para(151(a,1),89(a,1,2)),flip(a)].
231 x v (y v c(y v x)) = 1.  [para(15(a,1),98(a,1,2,2,1))].
235 c(x) v (c(x v y) v z) = c(x) v z.  [para(133(a,1),16(a,1,1)),flip(a)].
237 c(x) v (y v c(x v z)) = y v c(x).  [para(133(a,1),89(a,1,2)),flip(a)].
258 c(x) v (c(y v x) v z) = c(x) v z.  [para(221(a,1),16(a,1,1)),flip(a)].
260 c(x v y) v c(y v c(x v y)) = c(y).
[para(221(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),221(11)])].
261 c(x) v (y v c(z v x)) = y v c(x).  [para(221(a,1),89(a,1,2)),flip(a)].
263 x v (y v (c(y v x) v z)) = 1.
[para(231(a,1),16(a,1,1)),rewrite([141(2),16(5)]),flip(a)].
306 x v (y v c(z v (u v c(x)))) = y v x.  [para(224(a,1),89(a,1,2)),flip(a)].
314 x v (y v (z v c(y v x))) = 1.  [para(15(a,1),211(a,1,2,2,2,1))].
327 c(c(x) v y) v (z v x) = z v x.
[para(146(a,1),110(a,1,2,2,1,2,2,1,2)),rewrite([216(10),165(9),146(9)])].
666 x v (y v (c(z v c(x)) v u)) = y v (x v u).  [para(222(a,1),89(a,1,2)),flip(a)].
```

```
742 x v c(x v c(y v c(x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),197(8)])].
856 c(x) v (y v (c(z v x) v u)) = c(x) v (y v u).
[para(327(a,1),216(a,1,2,1,1)),rewrite([16(6),16(9)])].
936 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).  [para(235(a,1),89(a,1,2)),flip(a)].
1167 c(x) v (y v (z v c(x v u))) = y v (z v c(x)).
[para(16(a,1),237(a,1,2)),rewrite([16(9)])].
1410 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),261(a,1,2)),rewrite([16(9)])].
3702 x v c(y v c(x v y)) = x.
[para(742(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
3712 x v c(y v c(y v x)) = x.   [para(15(a,1),3702(a,1,2,1,2,1))].
3721 x v c(y v x) = x v c(y).
[para(3702(a,1),151(a,1,2,1)),rewrite([15(5),237(5)]),flip(a)].
3724 x v c(x v c(y)) = x v y.
[para(3702(a,1),109(a,1,2)),rewrite([3721(3),3721(7),3721(10),14(8),306(10)])].
3994 c(x v y) v c(y v c(x)) = c(y).  [back_rewrite(260),rewrite([3721(5)])].
3995 c(c(c1 v c3) v c(c3 v (c(c1) v (c3 v c(c2))))) != c3 v c(c3 v (c(c1) v c(c2))) #
answer("Foulis-Holland Theorem 3.1").
[back_rewrite(166),rewrite([3724(7),15(3),3721(27)])].
3996 x v (y v c(x v c(z))) = y v (z v x).  [back_rewrite(165),rewrite([3721(3)])].
4052 c(c(c1 v c3) v c(c2 v (c3 v c(c1)))) != c3 v c(c3 v (c(c1) v c(c2))) #
answer("Foulis-Holland Theorem 3.1").
[back_rewrite(3995),rewrite([3996(14),15(10),16(10)])].
4096 x v c(x v y) = x v c(y).
[para(3712(a,1),151(a,1,2,1)),rewrite([15(5),261(5)]),flip(a)].
4097 x v (c(y v x) v z) = x v (c(y) v z).
[para(208(a,1),3712(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),936(6)]),flip(a)].
4098 x v (y v c(z v x)) = x v (y v c(z)).
[para(211(a,1),3712(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1167(6)]),flip(a)].
4102 x v (c(x v y) v z) = c(y) v (x v z).
[para(263(a,1),3712(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),856(6)]),flip(a)].
4103 x v (y v c(x v z)) = x v (y v c(z)).
[para(314(a,1),3712(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1410(6)]),flip(a)].
4106 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3712(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),261(8),15(4),133(4)]),flip(
a)].
4107 c(x v y) v c(c(x) v y) = c(y).
[para(126(a,1),3712(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),261(8),15(4),221(4)]),flip(
a)].
4128 c(c(c1 v c3) v c(c2 v (c3 v c(c1)))) != c3 v c(c(c1) v c(c2)) # answer("Foulis-
Holland Theorem 3.1").  [back_rewrite(4052),rewrite([4096(23)])].
4343 x v c(y v (z v x)) = x v c(y v z).  [para(16(a,1),3721(a,1,2,1))].
4347 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3721(a,1),258(a,1,2)),rewrite([258(6)]),flip(a)].
4379 x v c(y v (x v z)) = x v c(y v z).  [para(89(a,1),4096(a,1,2,1))].
4424 c(x v c(c(y) v z)) = c(x v y) v c(c(y) v (x v c(z))).
[para(218(a,1),3994(a,1,1,1)),rewrite([15(8),4103(8)]),flip(a)].
4427 c(c(x v y) v z) = c(c(x) v z) v c(c(y) v (x v z)).
[para(235(a,1),3994(a,1,1,1)),rewrite([14(8),15(7),4102(7)]),flip(a)].
4514 c(c1 v c(c3)) v c(c(c1) v c(c2 v c3)) != c3 v c(c(c1) v c(c2)) # answer("Foulis-
Holland Theorem 3.1").
[back_rewrite(4128),rewrite([4427(13),4343(10),224(19),15(12),15(14)])].
4571 c(x v c(y v z)) = c(z v (x v c(y))) v c(x v c(z)).
[para(226(a,1),4106(a,1,2,1)),rewrite([14(2),15(4),4098(4),14(10)]),flip(a)].
4874 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4097(a,1),89(a,1,2)),flip(a)].
5181 c(x v y) v c(c(y) v (x v z)) = c(x v y) v c(x v z).
[para(4102(a,1),4379(a,1,2,1))].
5194 c(x v c(c(y) v z)) = c(x v y) v c(x v c(z)).
[back_rewrite(4424),rewrite([5181(13)])].
5757 c(x v y) v c(y v c(z v x)) = c(y).
[para(4343(a,1),4347(a,1,2,1)),rewrite([15(11),221(11)])].
5782 c(x v c(y v z)) v c(z v x) = c(x).  [para(5757(a,1),15(a,1)),flip(a)].
5902 c(x v c(y v z)) v c(y v x) = c(x).  [para(15(a,1),5782(a,1,1,1,2,1))].
6095 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5902(a,1,1,1))].
6105 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4103(a,1),5902(a,1,2,1)),rewrite([16(6)])].
6199 c(x v (c(y v z) v u)) = c(x v (z v (c(y) v u))) v c(x v (c(z) v u)).
[para(666(a,1),4106(a,1,2,1)),rewrite([14(2),15(5),4874(5),14(12)]),flip(a)].
```

```
6233 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6105),rewrite([6199(7),133(4),16(16),6095(15)]),flip(a)].
6535 c(x v (y v z)) v c(x v c(y)) = c(x v z) v c(x v c(y)).
[back_rewrite(5194),rewrite([6233(5),14(2)])].
6563 c(x v (y v c(z))) v c(y v c(x)) = c(y v c(z)) v c(y v c(x)).
[back_rewrite(4571),rewrite([6233(4),6535(8)]),flip(a)].
6564 $F # answer("Foulis-Holland Theorem 3.1").
[back_rewrite(4514),rewrite([6233(13),89(12),6563(20),89(19),4107(18),14(9),15(8)]),xx(a)
].

============================= end of proof =========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH3.1 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 3.9 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

1. FH3.1 is derivable from OML.

2. The proof in Section 3.0 is, as far as I know, novel.

3. Companion papers show that the other three Sasaki-implication restricted FH theorems are also derivable from OML.

4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Sasaki-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Sasaki-implication-restricted FH theorems, is the subject of future work.

5.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4*. URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH3.1 from OML. 2011. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics*. URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I*. Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices*.

Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.*   URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics*. Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls _pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information*. Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Sasaki U. Orthocomplemented lattices satisfying the exchange axiom. *Journal of Science Hiroshima University A* 17 (1964), 293-302.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Computer Science.*

An Automated Deduction of a "Sasaki-Implication-Restricted" Foulis-Holland Theorem from Orthomodular Quantum Logic: Part 4

Jack K. Horner

P. O. Box 266

Los Alamos, New Mexico 87544 USA

ICAI 2013

Abstract

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of quantum-mechanical behaviors and operations. The algebra, C(H), of closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space is a logic of the system of "measurement-propositions" quantum mechanical systems and is a model of an ortholattice (OL). An OL can thus be thought of as a kind of "quantum logic" (QL). C(H) is also a model of an orthomodular lattice (OML), which is an ortholattice to which the orthomodular law has been conjoined. An OML can thus be regarded as an orthomodular (quantum) logic (OMLogic). Now a QL can be thought of as a BL in which the distributive law does not hold. Under certain commutativity conditions, a QL does satisfy the distributive law; among the most well known of these relationships are the Foulis-Holland theorems (FHTs). Megill and Pavičić have defined variants of the QL "meet" and "join" connectives in terms of each of the five implication connnectives of QL; we can call these variant meet and join connectives "implication-restricted". Here I provide an automated deduction of one of the four Foulis-Holland theorems, restricted to "Sasaki" implication (one of the QL implication-connectives) , from OML theory.

Keywords: Sasaki implication, quantum computing , orthomodular lattice, Foulis-Holland theorems

1.0 Introduction

The optimization of quantum computing circuitry and compilers at some level must be expressed in terms of the description of quantum-mechanical behaviors ([1], [17], [18], [20]). In much the same way that conventional propositional (Boolean) logic (BL,[12]) is the logical structure of the system of measurement-propositions (e.g., "The position of particle P at time T is X") of classical physical systems and is isomorphic to a Boolean lattice ([10], [11],

[19]), so also the algebra, *C(H)*, of the closed linear subspaces of (equivalently, the system of linear operators on (observables in)) a Hilbert space *H* ([1], [4], [6], [9], [13]) is *a* logic of the system of measurement-propositions of quantum mechanical systems and is a model ([10]) of an ortholattice (OL; [4]). An OL can thus be thought of as a kind of "quantum logic" (QL; [19]). The lattice and ortholattice axioms, and the Sasaki-implication-restricted variants of "meet" and "join" are shown in Figure 1.

```
% Lattice axioms
    x = c(c(x))                     # label("AxL1").
    x v y = y v x                   # label("AxL2").
    (x v y) v z = x v (y v z)       # label("AxL3").
    (x ^ y) ^ z = x ^ (y ^ z)       # label("AxL4").
    x v (x ^ y) = x                 # label("AxL5").
    x ^ (x v y) = x                 # label("AxL6").

% Ortholattice axioms
    c(x) ^ x = 0                    # label("AxOL1").
    c(x) v x = 1                    # label("AxOL2").
    x ^ y = c(c(x) v c(y))          # label("AxOL3").

% Definition of Sasaki implication ([22])
    i1(x,y) = c(x) v (x ^ y)        # label("Df: i1").

% Definition of Sasaki-implication-restricted join
    u1(x,y) =  i1(c(x),y)           # label("Df: u1").

% Definition of Sasaki-implication-restricted meet
    int1(x,y) = c(i1(x,c(y)))       # label("Df: int1").

% Definition of x commutes with y
    C(x,y) <-> (x = ((x ^ y) v (x ^ c(y))))
          # label("Df:  commutes").
```

Figure 1. Lattice, ortholattice, and ortholattice axioms, and Sasaki-implication-restricted definitions of "meet" and "join". x, y, and z denote lattice nodes. "c(x)" denotes the orthocomplement of x. "v" denotes lattice join; "^" denotes lattice meet. "=" denotes identity; "<->" denotes "if and only if".

C(H) is also a model of an orthomodular lattice (OML; [4], [7]), which is an OL conjoined with the orthomodularity axiom (OMLaw):

```
y v (c(y) ^ (x v y)) =
    x v y      (OMLaw)
```

An OML can thus be thought of an "orthomodular (quantum) logic".

The rationalization of the OMA as a claim proper to physics has proven problematic ([13], Section 5-6), motivating the question of whether the OMA is required in an adequate characterization of QL. Thus formulated, the question suggests that the OMA and its equivalents are specific to an OML, and that as a consequence, banning the OMA from QL yields a "truer" quantum logic.

Now a QL can be thought of as a BL in which the distributive law

```
(D)    (x v (y ^ z) = (x v y) ^
          (x v z))
```

does not hold. Under certain commutativity conditions, a QL does satisfy (D); among the most well known of these relationships are the Foulis-Holland theorems (FHTs ([7])):

```
% Foulis-Holland theorem FH1
(C(x,y) & C(x,z)) ->  ( (x ^ (y v z)) = ((x ^ y) v (x ^ z))  )

% Foulis-Holland theorem FH2
(C(x,y) & C(x,z)) ->  ( (y ^ (x v z)) = ((y ^ x) v (y ^ z))  )


% Foulis-Holland theorem FH3
(C(x,y) & C(x,z)) ->  ( (x v (y ^ z)) = ((x v y) ^ (x v z))  )

% Foulis-Holland theorem FH4
(C(x,y) & C(x,z)) ->  ( (y v (x ^ z)) = ((y v x) ^ (y v z))  )


where C(x,y), "x commutes with y" is defined as

       C(x,y) <-> (x = ((x ^ y) v (x ^ c(y)))))
```

Figure 2. The Foulis-Holland theorems.

To form an implication-relativized version of any of the Foulis-Holland theorems, I replace meet and join in the consequents of Figure 2 with the corresponding implication-restricted meet and join (e.g., int1 and u1, as defined in Figure 1, are respectively meet and join defined in terms of implication-connective i1 in Figure 1). I denote the resulting Foulis-Holland variant by adding the suffix ".N" to the name of original theorem, where $N = 1$ denotes Sasaki-implication-restricted.

2.0 Method

The OML axiomatizations of Megill, Pavičić, and Horner ([5], [14], [15], [16],

[21]), were implemented in a *prover9* ([2]) script ([3]) configured to derive FH4.1, then executed in that framework on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 3 shows the proof, generated by [3] on the platform described in Section 2.0, that FH4.1 is a consequence of OML:

```
============================== PROOF =================================

% Proof 1 at 3.73 (+ 0.05) seconds: "Foulis-Holland Theorem 4.1".
% Length of proof is 123.
% Level of proof is 16.
% Maximum clause weight is 41.
% Given clauses 298.

3 C(x,y) <-> x = (x ^ y) v (x ^ c(y)) # label("Df: commutes") # label(non_clause).
[assumption].
4 C(x,y) & C(x,z) -> u1(y,int1(x,z)) = int1(u1(y,x),u1(y,z)) # label("Foulis-Holland
Theorem 4.1") # label(non_clause) # label(goal).  [goal].
10 -C(x,y) | (x ^ y) v (x ^ c(y)) = x # label("Df: commutes").  [clausify(3)].
11 C(c1,c2) # label("Foulis-Holland Theorem 4.1").  [deny(4)].
12 C(c1,c3) # label("Foulis-Holland Theorem 4.1").  [deny(4)].
13 x = c(c(x)) # label("AxL1").  [assumption].
14 c(c(x)) = x.  [copy(13),flip(a)].
```

```
15 x v y = y v x # label("AxL2").  [assumption].
16 (x v y) v z = x v (y v z) # label("AxL3").  [assumption].
18 x v (x ^ y) = x # label("AxL5").  [assumption].
19 x ^ (x v y) = x # label("AxL6").  [assumption].
20 c(x) ^ x = 0 # label("AxOL1").  [assumption].
21 c(x) v x = 1 # label("AxOL2").  [assumption].
22 x v c(x) = 1.  [copy(21),rewrite([15(2)])].
23 x ^ y = c(c(x) v c(y)) # label("AxOL3").  [assumption].
25 i1(x,y) = c(x) v (x ^ y) # label("Df: i1").  [assumption].
26 i1(x,y) = c(x) v c(c(x) v c(y)).  [copy(25),rewrite([23(3)])].
35 u1(x,y) = i1(c(x),y) # label("Df: u1").  [assumption].
36 u1(x,y) = x v c(x v c(y)).  [copy(35),rewrite([26(3),14(3),14(3)])].
45 int1(x,y) = c(i1(x,c(y))) # label("Df: int1").  [assumption].
46 int1(x,y) = c(c(x) v c(c(x) v y)).  [copy(45),rewrite([26(3),14(5)])].
67 1_2 = x v ((y ^ c(x)) v (c(y) ^ c(x))) # label("Df. 2.20").  [assumption].
68 x v (c(y v x) v c(c(y) v x)) = 1_2.
[copy(67),rewrite([23(3),14(4),23(7),14(6),14(6),15(7)]),flip(a)].
75 x v (c(x) ^ (y v x)) = y v x # label("OMLaw").  [assumption].
76 x v c(x v c(y v x)) = y v x.  [copy(75),rewrite([23(3),14(2)])].
77 int1(u1(c2,c1),u1(c2,c3)) != u1(c2,int1(c1,c3)) # label("Foulis-Holland Theorem 4.1")
# answer("Foulis-Holland Theorem 4.1").  [deny(4)].
78 c(c(c2 v c(c2 v c(c1))) v c(c2 v (c(c2 v c(c3)) v c(c2 v c(c2 v c(c1)))))) != c2 v
c(c2 v (c(c1) v c(c2 v c(c1)))) # answer("Foulis-Holland Theorem 4.1").
[copy(77),rewrite([36(3),36(10),46(15),15(24),16(24),46(31),15(34),36(38),14(39)])].
79 (c1 ^ c2) v (c1 ^ c(c2)) = c1.  [resolve(11,a,10,a)].
80 c(c2 v c(c1)) v c(c(c1) v c(c2)) = c1.
[copy(79),rewrite([23(3),23(10),14(11),15(10),15(12)])].
81 (c1 ^ c3) v (c1 ^ c(c3)) = c1.  [resolve(12,a,10,a)].
82 c(c3 v c(c1)) v c(c(c1) v c(c3)) = c1.
[copy(81),rewrite([23(3),23(10),14(11),15(10),15(12)])].
83 c(1) = 0.  [back_rewrite(20),rewrite([23(2),14(2),22(2)])].
84 c(c(x) v c(x v y)) = x.  [back_rewrite(19),rewrite([23(2)])].
85 x v c(c(x) v c(y)) = x.  [back_rewrite(18),rewrite([23(1)])].
89 x v (y v z) = y v (x v z).  [para(15(a,1),16(a,1,1)),rewrite([16(2)])].
97 x v (c(x) v y) = 1 v y.  [para(22(a,1),16(a,1,1)),flip(a)].
98 x v (y v c(x v y)) = 1.  [para(22(a,1),16(a,1)),flip(a)].
99 x v (c(x v y) v c(c(y) v x)) = 1_2.  [para(15(a,1),68(a,1,2,1,1))].
101 x v (c(y v x) v (c(c(y) v x) v z)) = 1_2 v z.
[para(68(a,1),16(a,1,1)),rewrite([16(9)]),flip(a)].
108 x v c(x v c(x v y)) = y v x.  [para(15(a,1),76(a,1,2,1,2,1))].
110 x v (y v c(x v (y v c(z v (x v y))))) = z v (x v y).
[para(76(a,1),16(a,1)),rewrite([16(7)]),flip(a)].
113 1_2 = 1.  [para(76(a,1),68(a,1,2,1,1)),rewrite([84(13),15(7),15(8),22(8)]),flip(a)].
122 x v (c(y v x) v (c(c(y) v x) v z)) = 1 v z.  [back_rewrite(101),rewrite([113(9)])].
124 x v (c(x v y) v c(c(y) v x)) = 1.  [back_rewrite(99),rewrite([113(8)])].
133 c(x) v c(x v y) = c(x).  [para(84(a,1),14(a,1,1)),flip(a)].
137 c(0 v c(x)) = x.  [para(22(a,1),84(a,1,1,2,1)),rewrite([83(3),15(3)])].
138 c(x v y) v c(x v c(x v y)) = c(x).
[para(84(a,1),76(a,1,2,1,2)),rewrite([15(5),133(11)])].
141 1 v x = 1.  [para(83(a,1),84(a,1,1,1)),rewrite([137(6)])].
143 x v (c(y v x) v (c(c(y) v x) v z)) = 1.  [back_rewrite(122),rewrite([141(10)])].
144 x v (c(x) v y) = 1.  [back_rewrite(97),rewrite([141(5)])].
146 x v c(c(x) v y) = x.  [para(14(a,1),85(a,1,2,1,2))].
150 x v 0 = x.  [para(22(a,1),85(a,1,2,1)),rewrite([83(2)])].
151 x v c(y v c(x)) = x.  [para(76(a,1),85(a,1,2,1))].
163 x v (y v c(x)) = y v 1.  [para(22(a,1),89(a,1,2)),flip(a)].
165 x v (y v c(x v c(z v x))) = y v (z v x).  [para(76(a,1),89(a,1,2)),flip(a)].
166 c(c(c2 v c(c2 v c(c1))) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v
c(c3 v c(c1)))) # answer("Foulis-Holland Theorem 4.1").
[para(89(a,1),78(a,1,1,2,1)),rewrite([108(23),15(17),15(18),16(18)])].
194 x v 1 = 1.  [para(141(a,1),15(a,1)),flip(a)].
196 x v (y v c(x)) = 1.  [back_rewrite(163),rewrite([194(5)])].
197 0 v x = x.  [para(150(a,1),15(a,1)),flip(a)].
208 x v (y v (c(x v y) v z)) = 1.  [para(144(a,1),16(a,1)),flip(a)].
211 x v (y v (z v c(x v y))) = 1.  [para(196(a,1),16(a,1)),flip(a)].
216 x v (c(c(x) v y) v z) = x v z.  [para(146(a,1),16(a,1,1)),flip(a)].
218 x v (y v c(c(x) v z)) = y v x.  [para(146(a,1),89(a,1,2)),flip(a)].
221 c(x) v c(y v x) = c(x).  [para(14(a,1),151(a,1,2,1,2))].
222 x v (c(y v c(x)) v z) = x v z.  [para(151(a,1),16(a,1,1)),flip(a)].
224 x v c(y v (z v c(x))) = x.  [para(16(a,1),151(a,1,2,1))].
226 x v (y v c(z v c(x))) = y v x.  [para(151(a,1),89(a,1,2)),flip(a)].
```

```
231 x v (y v c(y v x)) = 1.   [para(15(a,1),98(a,1,2,2,1))].
235 c(x) v (c(x v y) v z) = c(x) v z.   [para(133(a,1),16(a,1,1)),flip(a)].
237 c(x) v (y v c(x v z)) = y v c(x).   [para(133(a,1),89(a,1,2)),flip(a)].
258 c(x) v (c(y v x) v z) = c(x) v z.   [para(221(a,1),16(a,1,1)),flip(a)].
260 c(x v y) v c(y v c(x v y)) = c(y).
[para(221(a,1),76(a,1,2,1,2,1)),rewrite([14(6),15(5),221(11)])].
261 c(x) v (y v c(z v x)) = y v c(x).   [para(221(a,1),89(a,1,2)),flip(a)].
263 x v (y v (c(y v x) v z)) = 1.
[para(231(a,1),16(a,1,1)),rewrite([141(2),16(5)]),flip(a)].
314 x v (y v (z v c(y v x))) = 1.   [para(15(a,1),211(a,1,2,2,2,1))].
327 c(c(x) v y) v (z v x) = z v x.
[para(146(a,1),110(a,1,2,2,1,2,2,1,2)),rewrite([216(10),165(9),146(9)])].
666 x v (y v (c(z v c(x)) v u)) = y v (x v u).   [para(222(a,1),89(a,1,2)),flip(a)].
708 c2 v c(c2 v c(c1)) = c1 v c2.
[para(80(a,1),226(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
709 c3 v c(c3 v c(c1)) = c1 v c3.
[para(82(a,1),226(a,1,2)),rewrite([15(3),15(10)]),flip(a)].
728 c(c(c1 v c2) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v c(c3 v
c(c1)))) # answer("Foulis-Holland Theorem 4.1").   [back_rewrite(166),rewrite([708(7)])].
743 x v c(x v c(y v c(x v y))) = 1.
[para(98(a,1),124(a,1,2,1,1)),rewrite([83(2),15(6),197(8)])].
773 c3 v c(c1 v c3) = c3 v c(c1).   [para(709(a,1),108(a,1,2,1)),rewrite([15(10)])].
857 c(x) v (y v (c(z v x) v u)) = c(x) v (y v u).
[para(327(a,1),216(a,1,2,1,1)),rewrite([16(6),16(9)])].
937 c(x) v (y v (c(x v z) v u)) = y v (c(x) v u).   [para(235(a,1),89(a,1,2)),flip(a)].
990 c(c1 v c3) v c(c3 v c(c1)) = c(c3).
[para(709(a,1),138(a,1,1,1)),rewrite([709(12),773(10)])].
1029 c(c1) v c(c3 v c(c1)) = c(c1) v c(c3).   [para(990(a,1),235(a,1,2)),flip(a)].
1030 c(c(c1 v c2) v c(c2 v (c(c1) v c(c2 v c(c3))))) != c2 v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 4.1").   [back_rewrite(728),rewrite([1029(27)])].
1169 c(x) v (y v (z v c(x v u))) = y v (z v c(x)).
[para(16(a,1),237(a,1,2)),rewrite([16(9)])].
1412 c(x) v (y v (z v c(u v x))) = y v (z v c(x)).
[para(16(a,1),261(a,1,2)),rewrite([16(9)])].
3703 x v c(y v c(x v y)) = x.
[para(743(a,1),108(a,1,2,1)),rewrite([83(2),150(2),15(5)]),flip(a)].
3713 x v c(y v c(y v x)) = x.   [para(15(a,1),3703(a,1,2,1,2,1))].
3722 x v c(y v x) = x v c(y).
[para(3703(a,1),151(a,1,2,1)),rewrite([15(5),237(5)]),flip(a)].
3995 c(x v y) v c(y v c(x)) = c(y).   [back_rewrite(260),rewrite([3722(5)])].
3996 x v (y v c(x v c(z))) = y v (z v x).   [back_rewrite(165),rewrite([3722(3)])].
4062 c(c(c1 v c2) v c(c2 v (c3 v c(c1)))) != c2 v c(c2 v (c(c1) v c(c3))) #
answer("Foulis-Holland Theorem 4.1").
[back_rewrite(1030),rewrite([3996(14),15(9),15(10),16(10)])].
4096 x v c(x v y) = x v c(y).
[para(3713(a,1),151(a,1,2,1)),rewrite([15(5),261(5)]),flip(a)].
4097 x v (c(y v x) v z) = x v (c(y) v z).
[para(208(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),937(6)]),flip(a)].
4098 x v (y v c(z v x)) = x v (y v c(z)).
[para(211(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1169(6)]),flip(a)].
4102 x v (c(x v y) v z) = c(y) v (x v z).
[para(263(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),857(6)]),flip(a)].
4103 x v (y v c(x v z)) = x v (y v c(z)).
[para(314(a,1),3713(a,1,2,1,2,1)),rewrite([83(6),150(6),15(6),1412(6)]),flip(a)].
4106 c(x v y) v c(c(y) v x) = c(x).
[para(124(a,1),3713(a,1,2,1,2,1)),rewrite([83(8),150(8),15(8),261(8),15(4),133(4)]),flip(
a)].
4109 c(x v y) v (c(c(x) v y) v z) = c(y) v z.
[para(143(a,1),3713(a,1,2,1,2,1)),rewrite([83(9),150(9),15(9),857(9),258(5)]),flip(a)].
4128 c(c(c1 v c2) v c(c2 v (c3 v c(c1)))) != c2 v c(c(c1) v c(c3)) # answer("Foulis-
Holland Theorem 4.1").   [back_rewrite(4062),rewrite([4096(23)])].
4343 x v c(y v (z v x)) = x v c(y v z).   [para(16(a,1),3722(a,1,2,1))].
4347 c(x) v c(y v c(z v x)) = c(x) v c(y).
[para(3722(a,1),258(a,1,2)),rewrite([258(6)]),flip(a)].
4379 x v c(y v (x v z)) = x v c(y v z).   [para(89(a,1),4096(a,1,2,1))].
4424 c(x v c(c(y) v z)) = c(x v y) v c(c(y) v (x v c(z))).
[para(218(a,1),3995(a,1,1,1)),rewrite([15(8),4103(8)]),flip(a)].
4427 c(c(x v y) v z) = c(c(x) v z) v c(c(y) v (x v z)).
[para(235(a,1),3995(a,1,1,1)),rewrite([14(8),15(7),4102(7)]),flip(a)].
```

```
4514 c(c1 v c(c2)) v c(c(c1) v c(c2 v c3)) != c2 v c(c(c1) v c(c3)) # answer("Foulis-
Holland Theorem 4.1").
[back_rewrite(4128),rewrite([4427(13),4343(10),224(19),15(12),15(14)])].
4571 c(x v c(y v z)) = c(z v (x v c(y))) v c(x v c(z)).
[para(226(a,1),4106(a,1,2,1)),rewrite([14(2),15(4),4098(4),14(10)]),flip(a)].
4874 x v (y v (c(z v x) v u)) = y v (x v (c(z) v u)).
[para(4097(a,1),89(a,1,2)),flip(a)].
5181 c(x v y) v c(c(y) v (x v z)) = c(x v y) v c(x v z).
[para(4102(a,1),4379(a,1,2,1))].
5194 c(x v c(c(y) v z)) = c(x v y) v c(x v c(z)).
[back_rewrite(4424),rewrite([5181(13)])].
5757 c(x v y) v c(y v c(z v x)) = c(y).
[para(4343(a,1),4347(a,1,2,1)),rewrite([15(11),221(11)])].
5782 c(x v c(y v z)) v c(z v x) = c(x).  [para(5757(a,1),15(a,1)),flip(a)].
5902 c(x v c(y v z)) v c(y v x) = c(x).  [para(15(a,1),5782(a,1,1,1,2,1))].
6095 c(x v (y v c(z v u))) v c(z v (x v y)) = c(x v y).  [para(16(a,1),5902(a,1,1,1))].
6105 c(x v (c(y v z) v c(y v u))) v c(y v (x v c(z))) = c(x v c(y v z)).
[para(4103(a,1),5902(a,1,2,1)),rewrite([16(6)])].
6199 c(x v (c(y v z) v u)) = c(x v (z v (c(y) v u))) v c(x v (c(z) v u)).
[para(666(a,1),4106(a,1,2,1)),rewrite([14(2),15(5),4874(5),14(12)]),flip(a)].
6233 c(x v c(y v z)) = c(x v (z v c(y))) v c(x v c(z)).
[back_rewrite(6105),rewrite([6199(7),133(4),16(16),6095(15)]),flip(a)].
6535 c(x v (y v z)) v c(x v c(y)) = c(x v z) v c(x v c(y)).
[back_rewrite(5194),rewrite([6233(5),14(2)])].
6563 c(x v (y v c(z))) v c(y v c(x)) = c(y v c(z)) v c(y v c(x)).
[back_rewrite(4571),rewrite([6233(4),6535(8)]),flip(a)].
6564 $F # answer("Foulis-Holland Theorem 4.1").
[back_rewrite(4514),rewrite([6233(13),89(12),6563(20),4109(19),14(3)]),xx(a)].

=============================== end of proof ===========================
```

Figure 3. Summary of a *prover9* ([2]) proof of FH4.1 from OML. The proof assumes the default inference rules of *prover9*. The general form of a line in this proof is "*line_number conclusion [derivation]*", where *line_number* is a unique identifier of a line in the proof, and *conclusion* is the result of applying the *prover9* inference rules (such as *para*modulation, *copy*ing, and *rewrit*ing), noted in square brackets (denoting the *derivation*), to the lines cited in those brackets. The detailed syntax and semantics of these notations can be found in [2]. All *prover9* proofs are by default proofs by contradiction.

The total time to produce the proofs in Figure 3 on the platform described in Section 2.0 was approximately 3.9 seconds.

4.0 Discussion

The results of Section 3.0 motivate several observations:

  1. FH4.1 is derivable from OML.

  2. The proof in Section 3.0 is, as far as I know, novel.

  3. Companion papers show that the other three Sasaki-implication restricted FH theorems are also derivable from OML.

  4. The OMLaw is equivalent to each of the four FH theorems ([23]) in OML. Whether a Sasaki-implication-restricted variant of the OMLaw is derivable from OL (i.e., OML without the OMLaw) conjoined with the Sasaki-implication-restricted FH theorems, is the subject of future work.

5.0 References

[1] von Neumann J. *Mathematical Foundations of Quantum Mechanics*. 1936. Translated by R. T. Beyer. Princeton. 1983.

[2] McCune WW. *prover9 and mace4.* URL http://www.cs.unm.edu/~mccune/prover9/. 2009.

[3] Horner JK. *prover9* script for deriving FH4.1 from OML. 2011. Available from the author on request.

[4] Dalla Chiara ML and Giuntini R. *Quantum Logics.* URL http://xxx.lanl.gov/abs/quant-ph/0101028. 2004.

[5] Megill ND and Pavičić M. Orthomodular lattices and quantum algebra. *International Journal of Theoretical Physics* 40 (2001), pp. 1387-1410.

[6] Akhiezer NI and Glazman IM. *Theory of Linear Operators in Hilbert Space. Volume I.* Translated by M. Nestell. Frederick Ungar. 1961.

[7] Holland, Jr. SS Orthomodularity in infinite dimensions: a theorem of M. Solèr. *Bulletin of the American Mathematical Society* 32 (1995), pp. 205-234.

[8] Marsden EL and Herman LM. *A condition for distribution in orthomodular lattices.* Kansas State University Technical Report #40. 1974.

[9] Knuth DE and Bendix PB. Simple word problems in universal algebras. In J. Leech, ed. *Computational Problems in Abstract Algebra.* Pergamon Press. 1970. pp. 263-297.

[10] Chang CC and Keisler HJ. *Model Theory.* North-Holland. 1990. pp. 38-39.

[11] Birkhoff G. *Lattice Theory.* Third Edition. American Mathematical Society. 1967.

[12] Church A. *Introduction to Mathematical Logic. Volume I.* Princeton. 1956.

[13] Jauch J. *Foundations of Quantum Mechanics.* Addison-Wesley. 1968.

[14] Megill ND. *Metamath.* URL http://us.metamath.org/qlegif/mmql.html#unify. 2004.

[15] Horner JK. An automated deduction system for orthomodular lattice theory. *Proceedings of the 2005 International Conference on Artificial Intelligence.* CSREA Press. 2005. pp. 260-265.

[16] Horner JK. An automated equational logic deduction of join elimination in orthomodular lattice theory. *Proceedings of the 2007 International Conference on Artificial Intelligence.* CSREA Press. 2007. pp. 481-488.

[17] Messiah A. *Quantum Mechanics.* Dover. 1958.

[18] Horner JK. Using automated theorem-provers to aid the design of efficient compilers for quantum computing. *Los Alamos National Laboratory Quantum Institute Workshop.* December 9–10, 2002. URL http://www.lanl.gov/science/centers/quantum/qls_pdfs/horner.pdf.

[19] Birkhoff G and von Neumann J. The logic of quantum mechanics. *Annals of Mathematics* 37 (1936), 823-243.

[20] Nielsen MA and Chuang L . *Quantum Computation and Quantum Information.* Cambridge. 2000.

[21] Pavičić M and Megill N. Quantum and classical implicational algebras with primitive implication. *International Journal of Theoretical Physics* 37 (1998), 2091-2098. ftp://m3k.grad.hr/pavicic/quantum-logic/1998-int-j-theor-phys-2.ps.gz.

[22] Sasaki U. Orthocomplemented lattices satisfying the exchange axiom. *Journal of Science Hiroshima University A* 17 (1964), 293-302.

[23] Horner JK. Papers submitted to the *2013 International Conference on Foundations of Computer Science.*

SESSION

HEURISTIC SEARCH + INFORMATION RETRIEVAL + PATTERN AND SIMILARITY ANALYSIS

Chair(s)

TBA

Model Checking Safety Properties through Simulation and Heuristic Search

Nicoletta De Francesco[a], Giuseppe Lettieri[a], Antonella Santone[b], Gigliola Vaglini[a]
[a]Dip. di Ingegneria dell'Informazione, University of Pisa, Pisa, Italy
[b]Dip. di Ingegneria, University of Sannio, Benevento, Italy

Abstract—**This paper presents an efficient procedure to perform model checking of a concurrent process against a temporal logic formula, through the checking of a simulation between the description of the system and of the formula in the same formalism. The approach is applied, as an example, to processes defined through a specification language very compact, the well-know Calculus of Communicating Systems (CCS) defined by Milner, but it can be applied also to different languages as Lotos and CSP. The algorithm to explore the search space for the simulation problem is based on a greedy technique. The experiments show that a considerable reduction of both state space size and time can be achieved with respect to traditional model checking algorithms.**

Keywords-**Model checking; heuristic searches; temporal logic.**

I. INTRODUCTION

A very used method to verify concurrent and distributed systems is *model checking* [1]. It is an automated technique that, given a finite-state model of a system and a formal property, systematically checks whether this property holds for (a given state in) that model.

Equivalence checking is the process of determining whether two systems are related to each other according to some mathematically defined notion of equivalence. There are several kinds of equivalences, starting from the two equivalences defined by Milner for CCS processes: the strong equivalence that means that the processes are not distinguishable; the weak equivalence that defines a notion of observability for the processes and requires that equivalence be defined only with respect to the observable actions. Other equivalences exploit a different point of view on the observability, for example, equivalences were defined based on ignoring non-τ actions irrelevant for the expected system behavior [2], [3].

It is possible to use equivalence checking to realize model checking of a system, through comparison of the system model against the model derived from the formula; automata theoretic approach follows that way, by performing the product between the automaton representing the system and the automaton representing the formula (actually the complement of the formula): an empty result of the product signifies the satisfaction of the formula. However, this formal method cannot be scaled as easily with the increasing complexity of the systems, since at least the two automata have to be built to perform their product; moreover the construction of the automaton of the formula is not so straightforward for branching temporal logics.

This paper proposes to perform model checking of a concurrent system against a formula by applying *heuristic search* techniques on AND/OR graphs to the problem of verifying whether a concurrent system simulates another one (actually, one is the system to be verified, the other corresponds to the formula to be satisfied). Heuristic search is one of the classical techniques in Artificial Intelligence and has been applied to a wide range of problem-solving tasks including puzzles, two player games and path finding problems. In heuristic searches some *utility* or *cost* can be assigned to each state to guide the search by suggesting the next state to expand; in this way the most promising paths are considered first. There are several heuristic search algorithms for AND/OR graphs: a main difference among them is due to the fact of considering cyclic AND/OR graphs, or not cyclic AND/OR graphs. In the latter class it is included the algorithm AO* [4], while in the former it is included the algorithm S2 [5].

In any case, the AND/OR graph is expanded incrementally during the execution of the algorithm, starting from the initial node; the heuristic function assigns a cost to each node not yet considered, and the optimality of the solution can be guaranteed by the property of admissibility of the function, i.e., the heuristic never overestimate the distance to the goal. Optimality is not really an issue, from the point of view of the efficiency of the equivalence checking, for example: it is sometime better to find as soon as possible a solution than to repeatedly discard solutions until the optimum is found. Discarding optimality, a greedy approach can be followed: at any step, the next state to be expanded is chosen only on the basis of the foreseen cost of that expansion.

To model check the examples we show in this work a poorly informative heuristic function is used and, for this reason, it is not shown in detail here. The interesting thing is that, also in this case, the method avoids the construction of the entire model of the system and produces a very good result in terms of state space size and time with respect to traditional model checking algorithms, for example those defined inside the Concurrency WorkBench of the New Century (CWB-NC) [6]. The heuristic function can be refined so producing hopefully even better results. We consider concurrent systems (and formulae) specified by means of the Calculus of Communicating Systems (CCS)

[7]; for the simplicity of the presentation, we consider only temporal logic formulae expressing safety properties. The temporal logic we refer to is the selective mu-calculus [2] that adds abstraction features with the objective of saving space. In conclusion, the task of verifying that a CCS process p has the property φ is reduced to: (1) translating of the formula φ into a CCS process, say it f; and (2) checking if p and f respect a notion of ρ-simulation; ρ is the set of non-τ actions not relevant to the satisfaction of the formula and the heuristic search is based on the greedy approach.

As far as we know, it is the first attempt to exploit process algebra-based heuristics for simulation checking in verifying concurrent systems. A preliminary version of the results presented in this paper can be found in [8], where the simulation relation defined by Milner [7] has been considered. The method is completely automated, i.e., there is no need for user intervention or manual effort, moreover it can be analogously applied to other specification languages.

II. PRELIMINARIES

A. Calculus of Communicating Systems

Below we present a brief overview of the main features of Milner's Calculus of Communicating Systems (CCS)[7]. The syntax of *processes* is the following:
$$p ::= nil \mid x \mid \alpha.p \mid p+p \mid p\,|\,p \mid p\backslash L \mid p[f]$$
where α ranges over a finite set of actions $\mathcal{A} = \{\tau, a, \overline{a}, b, \overline{b}, ...\}$. The action $\tau \in \mathcal{A}$ is called the *internal action*. The set of *visible actions*, $\mathcal{V}$, ranged over by $l, l' \ldots$, is defined as $\mathcal{A} - \{\tau\}$. Each action $l \in \mathcal{V}$ (resp. $\overline{l} \in \mathcal{V}$) has a *complementary action* $\overline{l}$ (resp. l). The restriction set L, in the processes of the form $p\backslash L$, is a set of actions such that $L \subseteq \mathcal{V}$. The relabeling function f, in processes of the form $p[f]$, is a total function, $f : \mathcal{A} \to \mathcal{A}$, such that the constraint $f(\tau) = \tau$ is respected. The constant x ranges over a set of constant names: each constant x is defined by a constant definition $x \stackrel{\text{def}}{=} p$, where p is called the *body* of x. We denote the set of processes by $\mathcal{P}$. The semantics of a process p, defined in Table I, is given by a set of conditional rules describing the transition relation of the automaton corresponding to the behavior expression defining p, called the *standard transition system* for p and denoted by $\mathcal{S}(p)$.

Many equivalence relations have been defined on CCS processes; they are based on the notion of bisimulation between states of the related transition systems. In the following we consider the ρ-equivalence introduced in [2] formally characterizing the notion of "the same behavior with respect to a set ρ of actions". The precise definition of ρ-simulation uses the transition relation $\stackrel{\alpha}{\longrightarrow}_\rho$, parametric with respect to $\rho \subseteq \mathcal{A}$, which ignores all non-interesting actions (i.e. those in $\mathcal{A} - \rho$).

Definition 2.1: Let p and q be two CCS processes and $\rho \subseteq \mathcal{A}$ a set of actions, we define the relation $\longrightarrow_\rho \subseteq$

Act $\dfrac{}{\alpha.p \xrightarrow{\alpha} p}$	**Sum** $\dfrac{p \xrightarrow{\alpha} p'}{p+q \xrightarrow{\alpha} p'}$ (and symm.)
Rel $\dfrac{p \xrightarrow{\alpha} p'}{p[f] \xrightarrow{f(\alpha)} p'[f]}$	**Par** $\dfrac{p \xrightarrow{\alpha} p'}{p\,|\,q \xrightarrow{\alpha} p'\,|\,q}$ (and symm.)
Con $\dfrac{p \xrightarrow{\alpha} p'}{x \xrightarrow{\alpha} p'}\ x \stackrel{\text{def}}{=} p$	**Com** $\dfrac{p \xrightarrow{l} p',\ q \xrightarrow{\overline{l}} q'}{p\,|\,q \xrightarrow{\tau} p'\,|\,q'}$
Res $\dfrac{p \xrightarrow{\alpha} p'}{p\backslash L \xrightarrow{\alpha} p'\backslash L}\ \alpha, \overline{\alpha} \notin L$	

Table I
STANDARD OPERATIONAL SEMANTICS OF CCS

$\mathcal{P} \times \rho \times \mathcal{P}$ in the following way:
$$\forall \alpha \in \rho:\ p \stackrel{\alpha}{\longrightarrow}_\rho q \stackrel{\text{def}}{=} p \stackrel{\delta\alpha}{\longrightarrow} q, \text{ where } \delta \in (\mathcal{A} - \rho)^*$$

By $p \stackrel{\alpha}{\longrightarrow}_\rho q$ we express the fact that it is possible to pass from p to q by performing a (possibly empty) sequence of actions not belonging to ρ and then the action α in ρ. Note that $\longrightarrow_\mathcal{A} = \longrightarrow$.

Definition 2.2: (ρ-**simulation**) Let p and q be two CCS processes and let $\rho \subseteq \mathcal{A}$.

- A ρ-*simulation*, $\mathcal{R}$, is a binary relation on $\mathcal{P} \times \mathcal{P}$ such that $p\mathcal{R}q$ means that:
  $$p \stackrel{\alpha}{\longrightarrow}_\rho p' \text{ implies } q \stackrel{\alpha}{\longrightarrow}_\rho q' \text{ with } p'\mathcal{R}q';$$
- q ρ-simulates p ($p \prec_\rho q$) iff there exists a ρ-simulation $\mathcal{R}$ containing the pair (p, q).

Note that the ρ-simulation, with $\rho = \mathcal{V}$ does not coincide with the weak simulation. Suppose: $p = a.nil + \tau.nil$ and $q = a.nil$. It holds that q $\mathcal{V}$-simulates p, but q does not weak simulates p. On the other hand, if $p = a.nil + a.(c.nil + \tau.nil)$ and $q = a.(c.nil + \tau.nil)$, q weak simulates p but it is not true that q $\mathcal{V}$-simultes p. In the following, with ρ-derivatives of a CCS process p, we denote the set of all processes reachable from p by $\longrightarrow_\rho$.

B. SHML logic

To the purpose of explaining our methodology without too much technicality, system properties will be defined through the so-called Selective Hennessy-Milner Logic (SHML) instead of the full selective mu-calculus, introduced by Barbuti *et al.* [2]. However, SHML is more expressive than the Hennessy-Milner logic [7] because of the intrinsic recursion of the selective operators. Given a set $\mathcal{A}$ of actions, with $K, R \subseteq \mathcal{A}$, the SHML logic is the set of formulae so defined:
$$\varphi \quad ::= \quad \texttt{tt} \mid \texttt{ff} \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \langle K \rangle_R \varphi \mid [K]_R \varphi.$$
The satisfaction of a formula φ by a state s of a transition system, written $s \models \varphi$, is defined as follows: each state satisfies $\texttt{tt}$ and no state satisfies $\texttt{ff}$; a state satisfies $\varphi_1 \vee \varphi_2$ ($\varphi_1 \wedge \varphi_2$) if it satisfies φ_1 or (and) φ_2. $[K]_R \varphi$ and $\langle K \rangle_R \varphi$ are the selective modal operators: $[K]_R \varphi$ is satisfied by a state which, for every performance of a sequence of actions not belonging to $R \cup K$, followed by an action in K,

$p \not\models$ ff		$p \models$ tt
$p \models \varphi \wedge \psi$	iff	$p \models \varphi$ and $p \models \psi$
$p \models \varphi \vee \psi$	iff	$p \models \varphi$ or $p \models \psi$
$p \models [K]_R \varphi$	iff	$\forall p'.\forall \alpha \in K.p \xrightarrow{\alpha}_{K \cup R} p'$ implies $p' \models \varphi$
$p \models \langle K \rangle_R \varphi$	iff	$\exists p'.\exists \alpha \in K.p \xrightarrow{\alpha}_{K \cup R} p'$ and $p' \models \varphi$

Table II
SATISFACTION OF A SHML FORMULA BY A STATE

evolves to a state obeying φ. $\langle K \rangle_R \varphi$ is satisfied by a state which can evolve to a state obeying φ by performing a sequence of actions not belonging to $R \cup K$, followed by an action in K. The selective modal operators $\langle K \rangle_R \varphi$ and $[K]_R \varphi$ substitute the standard modal operators $\langle K \rangle \varphi$ and $[K] \varphi$. The basic characteristic of the SHML is that each formula allows us to immediately point out the parts of the transition system that do not alter the truth value of the formula itself. More precisely, *the only actions relevant for checking a formula are the ones explicitly mentioned by the selective modal operators used in the formula itself.* Thus, the result of checking the formula is independent from all other actions. In this paper we exploit this information to efficiently check ρ-simulation. A transition system $\mathcal{T}$ satisfies a SHML formula φ, written $\mathcal{T} \models \varphi$, if and only if $p \models \varphi$, where p is the initial state of $\mathcal{T}$. A CCS process p satisfies φ if $\mathcal{S}(p)$ satisfies φ. The precise definition of the satisfaction of a SHML formula φ by a state of a transition system is given in Table II.

C. Heuristic search: AND/OR Graphs

In this section, the general concept of AND/OR graphs to represent a search problem and the Greedy algorithm defined in [9] to visit AND/OR graphs and to find a suitable solution are presented.

An AND/OR graph G is a directed graph with a special node s, called the *start* (or *root*) *node*, and a nonempty set of *terminal leaf nodes* denoted as $t, t_1, \ldots$. The start node s represents the given problem to be solved, while the terminal leaf nodes correspond to subproblems with known solutions. The nonterminal nodes of G are of three types: OR, AND, and *nonterminal leaf*. An OR node is solved if one of its immediate subproblems is solved, while an AND node is solved only when every one of its immediate subproblems is solved. A nonterminal leaf node has no successors and is unsolvable.

Given an AND/OR graph G, a solution of G is represented by an AND/OR subgraph, called *solution (sub)graph of G* with the characteristics given below.

Definition 2.3: A finite subgraph D of an AND/OR graph G is a solution subgraph of G if:

(i) the start node of G is in D;
(ii) if n is an OR node in G and n is in D, then exactly one of the immediate successors of n in G is in D;

(iii) if n is an AND node in G and n is in D, then all the immediate successors of n in G are in D;
(iv) every non cyclic maximal path in D ends in a terminal leaf node.

In most domains the AND/OR graph G representing the problem is unknown in advance, so it cannot be supplied explicitly to the search algorithm. Instead, a distinction is drawn between the *explicit graph G'* and the *implicit graph G*. The implicit graph G is specified by a start node s and a successor function. The search algorithm works on the explicit graph G', which initially consists of the start node s only. The start node is then expanded using the successor function of G to obtain a set of nodes and arcs that are added to G'. At any instant, the explicit graph G' has a number of nodes yet to be expanded, called *tip nodes*. The search algorithm chooses one of these tip nodes for expansion. In this manner, more and more nodes and arcs get added to the explicit graph, until it finally has one or more solution graphs as subgraphs. One of these solution graphs is then output by the search algorithm. In this paper we use the Greedy algorithm for the heuristic search in AND/OR graphs that has been proposed in [9].

III. THE APPROACH

In this section the heuristic-approach to deduce a property satisfaction by checking ρ-simulation is presented. From now on, to simplify the presentation, SHML formulae containing only $[K]_R$ modal operators and $\wedge$ operators are considered: they are referred to as []-formulae, expressing safety properties. In particular, to verify if the property φ holds for the CCS process p, we (1) translate φ into a CCS process, say it f; and (2) check that p is simulated by f according to the notion of ρ-simulation, with ρ the set of actions occurring in φ. We analyse each step deeply.

A. Formula-to-CCS operator

We describe the formula-to-CCS operator: two special processes, $\bot$ and $\top$, are used with the following property: $\forall p \in \mathcal{P}$, $p \prec_\rho \top$ and $p \not\prec_\rho \bot$.

The $\mathcal{T}$ operator, given a []-formula and a set of actions, returns a special CCS process, called *sCCS*, that is a CCS process that can terminate either with a constant or with one of the special processes $\bot$ and $\top$.

Definition 3.1 (The formula-to-CCS operator $\mathcal{T}$): Let φ be a []-formula. The operator $\mathcal{T}$ is inductively defined on the structure of φ.

$\mathcal{T}(\text{ff}) = \{\bot\}$,

$\mathcal{T}(\text{tt}) = \{\top\}$,

$\mathcal{T}\big([K]_R \varphi\big) = \Big\{ x \, \Big| \, x \stackrel{\text{def}}{=} \sum_{k \in K} k.p + \sum_{\alpha \in \mathcal{A} - (K \cup R)} \alpha.x + \sum_{\beta \in R} \beta.\top, \, p \in \mathcal{T}(\varphi) \Big\}$,

$\mathcal{T}(\varphi_1 \wedge \varphi_2) = \mathcal{T}(\varphi_1) \cup \mathcal{T}(\varphi_2)$.

To check whether $p \models \varphi$ with $alph(\varphi) = \rho$[1], $\mathcal{T}$ builds the sCCS processes corresponding to φ: p ρ-simulates all of them if and only if $p \models \varphi$. If $\varphi = \mathtt{ff}$, the set built by $\mathcal{T}$ contains only the special process $\bot$, which ρ-simulates no process; while if $\varphi = \mathtt{tt}$ the set contains only the process $\top$ which ρ-simulates any process. When $\varphi = [K]_R \, \varphi'$, for each $q' \in \mathcal{T}(\varphi')$, a new constant x is defined as the summation of α such that:

- if $\alpha \in K$, the process x performs α moving then to q';
- if $\alpha \in \mathcal{A} - (K \cup R)$, the process x performs α and then returns to itself;
- if $\alpha \in R$, the process x performs α moving then to $\top$.

The idea is that x represents a state in which only actions in $\mathcal{A} - (K \cup R)$ (i.e., actions that have no effect on the satisfiability of formula φ) have already been performed. Any action in R from this state leads to vacuous satisfaction of φ, while any action in K leads to the check for the subformula φ'.

B. ρ-simulation checking

The second step of our approach consists of deducing whether the CCS process p satisfies the formula φ. This is obtained by checking whether all the sCCS processes obtained by the translation of φ ρ-simulate p. The following theorem states the correctness of this step.

Lemma 3.1: Let p be a CCS process and φ be a []-formula. For all $q \in \mathcal{T}(\varphi)$, $p \prec_{alph(\varphi)} q$ implies $p \prec_\rho q$ for all $alph(\varphi) \subseteq \rho \subseteq \mathcal{A}$.

Theorem 3.1: Let p be a CCS process and φ a []-formula. It holds that: $\left(\forall q \in \mathcal{T}(\varphi), \ p \prec_{alph(\varphi)} q \right)$ iff $p \models \varphi$.

Proof. The proof is by induction on the structure of the formula φ.

To check ρ-simulation we use an heuristic approach defined in the next section

C. AND/OR graph for ρ-simulation checking

Let p be a CCS finite process and q a finite sCCS process: to check the requirements of the ρ-simulation (Definition 2.2) an AND/OR graph is built representing the fact that p and q move in alternating turns as long as they can perform the same move. Thus, the AND/OR graph has a solution iff $p \prec_\rho q$, since the construction halts with nonterminal leafs only when there is a move of p that cannot be matched by q.

In order to apply the algorithm described in Section II-C the operators used to expand a node have been defined. In the following, we denote by $G(p, q)$ the implicit AND/OR graph built from p, q and the transition relation $\longrightarrow_\rho$. The nodes of $G(p, q)$ are 3-uples $\langle r, s, \gamma \rangle$ where r is a ρ-derivative of p; s is a ρ-derivative of q or the processes $\top$ and $\bot$; and $\gamma \in \{\lambda\} \cup \mathcal{A}$.

---

[1]The function $alph(\varphi)$ returns all actions occurring in the sets R and K of all selective operators appearing in the SHML formula φ.

$\mathbf{op}_1$	$\dfrac{\{\, (p', \alpha) \mid p \xrightarrow{\alpha}_\rho p' \,\} = \{(p_1, \alpha_1), \ldots, (p_n, \alpha_n)\} \neq \varnothing}{\langle p, q, \lambda \rangle \longrightarrow \langle p_1, q, \alpha_1 \rangle \ \text{and} \ \cdots \ \text{and} \ \langle p, q, \lambda \rangle \longrightarrow \langle p_n, q, \alpha_n \rangle}$
$\mathbf{op}_2$	$\dfrac{\{\, q' \mid q \xrightarrow{\beta}_\rho q' \ \text{and} \ \beta = \alpha \,\} = \{q_1, \ldots, q_n\} \neq \varnothing}{\langle p, q, \alpha \rangle \longrightarrow \langle p, q_1, \lambda \rangle \ \text{or} \ \cdots \ \text{or} \ \langle p, q, \alpha \rangle \longrightarrow \langle p, q_n, \lambda \rangle}$

Table III
GENERAL OPERATORS.

We assume $\lambda \notin \mathcal{A}$ and s be a sCCS process. When $\gamma = \lambda$ it is the turn of r to move; when $\gamma = \alpha$ then s has to move performing the action α, i.e. the action that the process r has performed in the previous turn. Let $n = \langle r, s, \gamma \rangle$ be a node of $G(p, q)$. Node n is an AND node, OR node, terminal or nonterminal leaf according to the value of γ and the following rules:

- when $\gamma = \lambda$, node n is an AND node if $r \neq s$ and $s \notin \{\bot, \top\}$;
- when $\gamma = \alpha$, node n is an OR node if s has a $\xrightarrow{\alpha}_\rho$ transition, otherwise it is a nonterminal leaf;
- when $\gamma = \lambda$, node n is a terminal leaf, if $r = s$ or $s = \top$;
- when $\gamma = \lambda$, node n is a nonterminal leaf if $s = \bot$.

The start node of $G(p, q)$ is $\langle p, q, \lambda \rangle$. The successor function is given by the operators in Table III. The operators generate the outgoing arcs and the successor nodes of each node. The operators with the form

$$\frac{\textit{premise}}{n \longrightarrow n_1 \ \text{and} \ \cdots \ \text{and} \ n \longrightarrow n_m}$$

where *premise* is the antecedent, possibly empty, of the rule, generate all the outgoing arcs and successor nodes of the AND node n. On the other hand, the operators with the form

$$\frac{\textit{premise}}{n \longrightarrow n_1 \ \text{or} \ \cdots \ \text{or} \ n \longrightarrow n_m}$$

generate all the outgoing arcs and successor nodes of the OR node n.

The rule $\mathbf{op}_1$ points out the possible moves (α_i) of p when $\gamma = \lambda$; in this way an AND node can be connected with its successor nodes in the graph, all such successors have $\gamma = \alpha_i$; roughly speaking if p can move performing an action α_i and reaches the process p_i then it is the turn of q to move with the same action α_i. In rule $\mathbf{op}_2$, q must simulate the action α performed by p; in this case an OR node can be connected with its successor nodes in the graph, all such successors have $\gamma = \lambda$ i.e. nodes that can be transformed only through the operator $\mathbf{op}_1$ like the initial node.

It can be proved that finding a solution of $G(p, q)$ is equivalent to checking whether $p \prec_\rho q$. In this preliminary work, since the idea is to evaluate goodness of the application of

heuristic search to check properties through simulation, we use a heuristic function always equal to zero. As reflected in the experiments we made, this non informative heuristic has the advantage of a lower computational cost, while retaining the reduction of state space due to the greedy algorithm.

IV. EXPERIMENTAL RESULTS

In this section we present and discuss our experience with using the tool, called GreASE (Greedy Algorithm for System Equivalence); this tool applies the presented approach to model checking (via ρ-simulation) to several well-known CCS processes. The aim is to evaluate the performances of the approach and compare it against the CWB-NC. Experiments were executed on a 64 bit, 2.67 GHz Intel i5 CPU equipped with 8 GiB of RAM and running Gentoo Linux. The tool is freely available for download at the url: //www2.ing.unipi.it/~a080224/grease. For the evaluation, a sample of well known systems is selected from the literature. The results are shown in Table IV, where the number of generated nodes and the time (expressed in sec) resulting from model checking the selected case studies are reported.

Railway system (crail): we applied our tool to the system specification given in [10]. This system describes the British Rail's Solid State Interlocking which is devoted "to adjust, at the request of the signal operator, the setting of signal and points in the railway to permit the safe passage of trains". On this system we checked the following formula $\varphi_1 = [det]_{\{fail\}}$ ff, which means that a failure is detected only if it has actually occurred. The system does not satisfy φ_1.

Mutual (mutual8, mutula10): it is a system handling the requests of a resource shared by 8 (resp. 10) processes. It presents two alternative choices between a server based on a round robin scheduling and a server based on mutual exclusion. On this system we checked the following two formulae: $\varphi_1 = [\overline{roundrobin}]_\emptyset [\overline{work}_{i+1}]_{\{\overline{work}_i\}}$ ff and $\varphi_2 = [\overline{work}_{i+1}]_{\{\overline{work}_i\}}$ ff. φ_1 expresses the property that when a server based on a round robin scheduling is chosen, process $i+1$ cannot use the resource before process i. This property is satisfied, instead the system does not satisfy φ_2 expressing the property that, in general, process $i+1$ cannot use the resource before process i.

Philips Bounded Retransmission Protocol (BRP): the Bounded Retransmission Protocol used by the Philips Company in one of its products [11]. On this system we checked the following two formulae:

$$\varphi_1 = [\texttt{in(d1\_d10)}]_\emptyset \Big([\overline{\texttt{out(d10,ok)}}]_{\{\overline{\texttt{out(d9,inc)}}\}}\texttt{ff}\wedge$$
$$[\overline{\texttt{out(d9,inc)}}]_{\{\overline{\texttt{out(d8,inc)}}\}}\texttt{ff} \wedge \cdots \wedge$$
$$[\overline{\texttt{out(d2,inc)}}]_{\{\overline{\texttt{out(d1,fst)}}\}}\texttt{ff}\Big)$$
$$\varphi_2 = [\texttt{in(d1\_d10)}]_\emptyset \ [\overline{\texttt{in(ok)}}]_{\{\overline{\texttt{out(d10,ok)}}\}}\texttt{ff}$$

case study	our approach		CWB-NC	
	gen	time	gen	time
Crail (φ_1)	1197	0.020	3628	0.564
Mutual8 (φ_1)	644	0.507	6912	1.616
Mutual8 (φ_2)	387	0.049	6912	1.410
Mutual10 (φ_1)	2564	7.002	17408	6.885
Mutual10 (φ_2)	1539	0.322	17408	5.448
BRP (φ_1)	40	0.025	759	0.262
BRP (φ_2)	15	0.044	759	0.189
MPMC (φ_1)	19	0.004	13803	3.598
MPMC (φ_2)	1611	1.439	13803	3.609

Table IV
RESULTS FOR CCS SYSTEMS

The formula φ_1 means that, after accepting a data packet of length 10 (action in(d1_d10)), the protocol cannot deliver the i-th segment ($2 \leq \texttt{i} \leq 10$) to the receiving client before having delivered the (i-1)-th one. φ_2 means that, after accepting a data packet of length 10, the protocol cannot issue an ok confirmation (action in(ok)) to the sending client unless the last segment of the packet (action $\overline{\texttt{out(d10,ok)}}$) has been delivered to the receiving client.

Multicast Protocol for Mobile Computing (MPMC): a protocol for reliable multicast in distributed mobile systems, presented in [12]. On this system we checked the following two formulae:

$$\varphi_1 = [\overline{deliver(m,msg)}]_{\{send(msg)\}} \texttt{ff}$$
$$\varphi_2 = [\overline{deliver(m,msg)}]_\emptyset [\overline{deliver(m,msg)}]_\emptyset \texttt{ff}$$

φ_1 expresses the property that any multicast delivered by a group member has been originated by a group member; and φ_2 expresses the property that no group member delivers duplicate multicasts, i.e. duplicates are discarded. Both properties are true.

These examples provide some experimental evidence of the reduction of the state space that may result when applying the presented methodology instead of the standard algorithm used by the CWB-NC. Note that in all cases we obtain great reductions both in time and in state space.

V. CONCLUSION AND RELATED WORK

A method has been proposed for model checking safety properties for concurrent systems described in CCS. The novel contributions of the work are the following:

- application of algorithms taken from heuristic search environment for the ρ-simulation checking;
- model checking safety properties through ρ-simulation.

The most challenging task when applying automated model checking in practise is to conquer the state explosion problem. Hence, simulation algorithms with minimal space complexity are of particular interest. Two algorithmic families can be considered: one is based on the refinement principle, that is, *given an initial partition, find the coarsest partition stable with respect to the transition relation*, see

for example the algorithm proposed by Paige and Tarjan in [13]. The other family of algorithms is based on a Cartesian product traversal from the initial state [14]. These algorithms are both applied on the whole state graph, and require an explicit enumeration of such space so producing the well-known state explosion problem.

The algorithms in [15], [16] consider a problem with N states, and S simulation equivalence classes: the space complexity of both algorithms is $\mathcal{O}(S^2 + NlogS)$.

Recently great interest has grown in combining model checking and heuristics to guide the exploration of the state graph of a system. In the domain of software validation, the work of Yang and Dill [17] is one of the original ones. They enhance the bug-finding capability of a model checker by using heuristics to search the states that are most likely to lead to an error. In [18] heuristics have been used for real-time model checking in UPPAAL. In [19] heuristic search has been combined with on-the-fly techniques, while in [20] with symbolic model checking. Previous works, as for example [21], [22], [23], [24], used heuristic search to accelerate finding errors, while in [25] heuristic search is used to accelerate verification.

The presented approach starts from representing the system and the logic formula in the same formalism and, then, tries to state if the formula is satisfied by the system by proving that the system behavior is simulated the formula behavior. In [26] Vardi and Wolper describe an automata-theoretic approach to verify finite-states systems: the idea underlying this approach is that, for any linear temporal logic (LTL) formula, one can construct an automaton that accepts precisely the computations satisfied by the formula. Thus, if p is a system and φ a LTL formula, p meets φ if every infinite word generated by p, viewed as a finite-state generator, is accepted by φ, viewed as a finite state acceptor. This reduces the model checking problem to a purely automata-theoretic problem: the problem of determining if the automaton $p \cap \overline{\varphi}$ is empty, i.e. if it accepts no word. On the other hand, for branching temporal logic, automata-theoretic techniques have long been thought to introduce an exponential penalty, making them essentially useless. But, Bernholtz and Grumberg in [27] suggested that an automata-theoretic approach to branching-time model checking can be based on the concepts of amorphous automata and simultaneous trees, and they showed that CTL model checking is linearly reducible to the acceptance of a simultaneous tree by an amorphous automaton, problem that can be solved in quadratic running time. Although this constitutes a meaningful first step towards applying automata-theoretic techniques to branching-time model checking, it is not quite satisfactory, above all for the complexity of the resulting algorithm which is quadratic in both the size of the formula and the size of the system.

Our attempt is in this direction: the heuristic search methodology avoids the construction of both automata and

the results in terms of space and running time are encouraging. The next step consists in managing full selective mu-calculus formulae and in defining a more informative heuristic function.

Other approaches have been developed to solve or reduce the state explosion problem for model checking. Among them, reduction techniques based on symbolic model checking techniques [28], partial order techniques [29], [30], compositional techniques [31], [32], and abstraction approaches [33], [34]. Moreover, all the above approaches has been successfully applied in several fields, like for example clone detection [35], [36], analysis of social networks [37], incremental construction of complex systems [38], secure information flow in concurrent programs [39], banking industry [40], [41].

REFERENCES

[1] E. M. Clarke, O. Grumberg, and D. Peled, *Model checking.* MIT Press, 2001.

[2] R. Barbuti, N. De Francesco, A. Santone, and G. Vaglini, "Selective mu-calculus and formula-based equivalence of transition systems," *J. Comput. Syst. Sci.*, vol. 59, no. 3, pp. 537–556, 1999.

[3] J.-C. Fernandez, H. Garavel, A. Kerbrat, L. Mounier, R. Mateescu, and M. Sighireanu, "Cadp - a protocol validation and verification toolbox," in *CAV*, ser. Lecture Notes in Computer Science, R. Alur and T. A. Henzinger, Eds., vol. 1102. Springer, 1996, pp. 437–440.

[4] A. Mahanti and A. Bagchi, "And/or graph heuristic search methods," *J. ACM*, vol. 32, no. 1, pp. 28–51, 1985.

[5] A. Mahanti, S. Ghose, and S. K. Sadhukhan, "A framework for searching and/or graphs with cycles," *CoRR*, vol. cs.AI/0305001, 2003.

[6] R. Cleaveland and S. Sims, "The ncsu concurrency workbench," in *CAV*, ser. Lecture Notes in Computer Science, R. Alur and T. A. Henzinger, Eds., vol. 1102. Springer, 1996, pp. 394–397.

[7] R. Milner, *Communication and concurrency*, ser. PHI Series in computer science. Prentice Hall, 1989.

[8] A. Santone, "Heuristic for simulation checking," in *Proceedings of the 2011 International Conference on Artificial Intelligence (ICAI 2011), Las Vegas Nevada, USA.* CSREA Press, 2011.

[9] N. De Francesco, G. Lettieri, G. Vaglini, and A. Santone, "Grease: a tool for efficient equivalnce checking," http://www2.ing.unipi.it/~a080224/grease/TR-03-2011.pdf, December 2011.

[10] G. Bruns, "A case study in safety-critical design," in *CAV*, ser. Lecture Notes in Computer Science, G. von Bochmann and D. K. Probst, Eds., vol. 663. Springer, 1992, pp. 220–233.

[11] J. F. Groote and J. van de Pol, "A bounded retransmission protocol for large data packets," in *AMAST*, ser. Lecture Notes in Computer Science, M. Wirsing and M. Nivat, Eds., vol. 1101. Springer, 1996, pp. 536–550.

[12] G. Anastasi, A. Bartoli, N. D. Francesco, and A. Santone, "Efficient verification of a multicast protocol for mobile computing," *Comput. J.*, vol. 44, no. 1, pp. 21–30, 2001.

[13] R. Paige and R. E. Tarjan, "Three partition refinement algorithms," *SIAM J. Comput.*, vol. 16, no. 6, pp. 973–989, 1987.

[14] J.-C. Fernandez and L. Mounier, ""on the fly" verification of behavioural equivalences and preorders," in *CAV*, ser. Lecture Notes in Computer Science, K. G. Larsen and A. Skou, Eds., vol. 575. Springer, 1991, pp. 181–191.

[15] D. Bustan and O. Grumberg, "Simulation-based minimazation," *ACM Trans. Comput. Log.*, vol. 4, no. 2, pp. 181–206, 2003.

[16] R. Gentilini, C. Piazza, and A. Policriti, "From bisimulation to simulation: Coarsest partition problems," *J. Autom. Reasoning*, vol. 31, no. 1, pp. 73–103, 2003.

[17] C. H. Yang and D. L. Dill, "Validation with guided search of the state space," in *DAC*, 1998, pp. 599–604.

[18] G. Behrmann and A. Fehnker, "Efficient guiding towards cost-optimality in uppaal," in *TACAS*, ser. Lecture Notes in Computer Science, T. Margaria and W. Yi, Eds., vol. 2031. Springer, 2001, pp. 174–188.

[19] R. Alur and B.-Y. Wang, ""next" heuristic for on-the-fly model checking," in *CONCUR*, ser. Lecture Notes in Computer Science, J. C. M. Baeten and S. Mauw, Eds., vol. 1664. Springer, 1999, pp. 98–113.

[20] R. M. Jensen, R. E. Bryant, and M. M. Veloso, "Seta*: An efficient bdd-based heuristic search algorithm," in *AAAI/IAAI*, 2002, pp. 668–673.

[21] G. Francesca, A. Santone, G. Vaglini, and M. L. Villani, "Ant colony optimization for deadlock detection in concurrent systems," in *COMPSAC*, 2011, pp. 108–117.

[22] S. Gradara, A. Santone, and M. L. Villani, "Delfin$^+$: An efficient deadlock detection tool for ccs processes," *J. Comput. Syst. Sci.*, vol. 72, no. 8, pp. 1397–1412, 2006.

[23] S. Edelkamp, A. Lluch-Lafuente, and S. Leue, "Directed explicit model checking with hsf-spin," in *SPIN*, ser. Lecture Notes in Computer Science, M. B. Dwyer, Ed., vol. 2057. Springer, 2001, pp. 57–79.

[24] S. Gradara, A. Santone, and M. L. Villani, "Using heuristic search for finding deadlocks in concurrent systems," *Inf. Comput.*, vol. 202, no. 2, pp. 191–226, 2005.

[25] A. Santone, "Heuristic search + local model checking in selective mu-calculus," *IEEE Trans. Software Eng.*, vol. 29, no. 6, pp. 510–523, 2003.

[26] M. Y. Vardi and P. Wolper, "An automata-theoretic approach to automatic program verification (preliminary report)," in *LICS*, 1986, pp. 332–344.

[27] O. Bernholtz and O. Grumberg, "Branching time temporal logic and amorphous tree automata," in *CONCUR*, 1993, pp. 262–277.

[28] K. L. McMillan, *Symbolic model checking*. Kluwer, 1993.

[29] A. Santone and G. Vaglini, "Partial order interpretation of a mu-calculus-like temporal logic," in *ICSOFT*, 2013.

[30] P. Godefroid, *Partial-Order Methods for the Verification of Concurrent Systems - An Approach to the State-Explosion Problem*, ser. Lecture Notes in Computer Science. Springer, 1996, vol. 1032.

[31] E. M. Clarke, D. E. Long, and K. L. McMillan, "Compositional model checking," in *LICS*. IEEE Computer Society, 1989, pp. 353–362.

[32] A. Santone, "Automatic verification of concurrent systems using a formula-based compositional approach," *Acta Inf.*, vol. 38, no. 8, pp. 531–564, 2002.

[33] A. Santone and G. Vaglini, "Abstract reduction in directed model checking ccs processes," *Acta Inf.*, vol. 49, no. 5, pp. 313–341, 2012.

[34] E. M. Clarke, O. Grumberg, and D. E. Long, "Model checking and abstraction," *ACM Trans. Program. Lang. Syst.*, vol. 16, no. 5, pp. 1512–1542, 1994.

[35] A. Cuomo, A. Santone, and U. Villano, "A novel approach based on formal methods for clone detection," in *IWSC*, 2012, pp. 8–14.

[36] A. Santone, "Clone detection through process algebras and java bytecode," in *IWSC*, 2011, pp. 73–74.

[37] A. Santone and G. Vaglini, "Modelling and analysing social networks through formal methods and heuristic searches," in *ICSOFT*, 2012, pp. 336–339.

[38] A. Santone, G. Vaglini, and M. Villani, "Incremental construction of systems: An efficient characterization of the lacking sub-system," *Science of Computer Programming*, 2012.

[39] N. D. Francesco, A. Santone, and L. Tesei, "Abstract interpretation and model checking for checking secure information flow in concurrent systems," *Fundam. Inform.*, vol. 54, no. 2-3, pp. 195–211, 2003.

[40] A. Santone, V. Intilangelo, and D. Raucci, "Application of equivalence checking in a loan origination process in banking industry," in *WETICE*, 2013.

[41] V. Intilangelo, D. Raucci, and A. Santone, "Efficient formal verification in banking processes," in *SERVICES*, 2013.

[42] R. Alur and T. A. Henzinger, Eds., *Computer Aided Verification, 8th International Conference, CAV '96, New Brunswick, NJ, USA, July 31 - August 3, 1996, Proceedings*, ser. Lecture Notes in Computer Science, vol. 1102. Springer, 1996.

Intelligent Genomic Information Retrieval with Distributed System Resources

Wagner Arbex*
Leonardo Carvalho Nápolis Costa
Leonardo Mariano Gravina Fonseca
*Brazilian Agricultural Research
Corporation*
Juiz de Fora, MG, Brazil

Camillo de Léllis Falcão da Silva
Federal University of Juiz de Fora
Juiz de Fora, MG, Brazil

Abstract

The National Center for Biotechnology Information (NCBI) website provides innumerable computational resources from several web services, which can be reached with some programming languages and/or using specific modules of theses programming languages. Nevertheless, the totality of these services or almost all services must be accessed from knowledge of metainformation about the data to be mined. Eventually, without knowing any information about what one wants to search, such as a single DNA sequence representing "string" by computational view, hindering access to NCBI's resources. This paper presents a set of bioinformatics resources to enable metainformation discovery of raw genomic data such as genetic sequences without any knowledge about them.

Keywords: Intelligent retrieval, genomic information, NCBI, BLAST, distributed system

1. Introduction

The methods of computerized information process are essential for conducting scientific research in all fields of knowledge, such as genomic or biomedical research, and the National Center for Biotechnology Information (NCBI) keeps the largest data repositories and bioinformatics computing resources worldwide. NCBI was established on November 4, 1988, as a division of the National

wagner.arbex@embrapa.br

Library of Medicine (NLM) at the National Institutes of Health (NIH) [1]. NCBI has experience in creating and maintaining biomedical databases, serving as a research program in computational molecular biology.

The NCBI aims at developing new information technologies to help and understand molecular and genetic processes that are connected to control health and disease. Moreover, the NCBI is coordinating efforts to gather biotechnology information, and create and maintain automated systems for storing and analyzing knowledge about molecular biology, biochemistry and genetics, thus facilitating the use of databases and software for searching the scientific community [1].

The issue featured in the Section 2 occurs when it is necessary perform searches without knowing any information about the data to be studied, because almost all NCBI's search services are based on some knowledge about the investigated data. This paper presents a solution for cases where the data investigated is raw nucleotide sequences raw and which are not known any information, even the organism of origin of sequences.

2. Problem Setting

The NCBI's responsibilities [1] are diverse, featuring several tools that allow you to use mathematical and computational methods, it helps maintain collaborations with several institutions, industry and other government agencies. Promoting scientific communication through meetings, workshops, lectures and training in support of basic and

applied research in bioinformatics. It has a variety of databases and software, in addition to developing and promoting standards for data storage and biological nomenclature.

The Entrez system [2] is an example of NCBI's resources, which support databases, named Entrez Databases. However, this resource does not only allows access to databases. Actually, the Entrez is a retrieval system designed for searching several linked databases. More specifically, Entrez is a search and retrieval system of NCBI that offers users integrated access to sequencing, mapping, taxonomy and structural data and provides graphical views of sequences, maps of chromosomes or other structures.

From of the NCBI unique identifier of data, can be reached innumerable information about the them. For instance, a NCBI unique identifier is the "accession number", which is "given to a sequence when it is submitted to one of the DNA repositories (GenBank, EMBL, DDBJ)" [3] and it is possible to access many information from Entrez resources.

Using Entrez resources can retrieve all information about a DNA sequence, or part of them. In the Figure 1, it can be seen which was recovered from the query [4]:

```
http://eutils.ncbi.nlm.nih.gov/entrez/
eutils/efetch.fcgi?db=nuccore&id=34577062,
24475906&rettype=fasta&retmode=text
```

```
>gi|34577062|ref|NM_001126.2| Homo sapiens (...), mRNA
GGAAGGGGCGTGGCCTCGGTCCGGGGTGGCGGCCGTTGCCGCCACCAGGGCCTCTTC
CTGCGGGCGGTGCTGCCGAGGCCGGCCTGCGCGGGGCAGTCATGGTACCCCCTTGAG
CGGGCTGTGGCGGAGAGCGGGGCGGGGACTGGCTGGAGGGTGGCGGCCCGGCGGGGC
GGGGGCGGGGCCGGCCTCTGGCTCCTTCTTCCTCTGCATGTGGCTGGCGGCCGCAGA
GCAGTTCAGTTCGCTCACTCCTCGCCGGCCGCCTCTCCTTCGGGCTCTCCTCGCGTC
ACTGGAGCCATGGCCGTTCGCCGAGACCTACCCGGCGGCATCCTCCCTGCCCAACGGC
GATTGCGG
(...)
>gi|24475906|ref|NM_009417.2| Mus musculus (...), mRNA
CACAGAGAACACACCCAGCGGTGCACATCCTGCTTCTTCCGCTGCTGAGAAAAGGAA
AACGTTCAGCTCTAGAATGAGAACACTTGGAGCTATGGCAATAATGCTGGTGGTTAT
GGGAACTGTAATTTTCCTCTCTTTTATCCTGAGAAGCAGAGACATCTTGTGTGGGAA
GACCATGAAGTCCCATGTTATCAGTGCTGTGGAAACGAGCCAGCTCATGGTGGACCA
TGCAGTCTACAACACCATGAAAAGAAACCTCAAGAAAAGGGAAGTCCTTTCTCCAGC
CCAGCTTCTCTCTTTCTTTAAGCTGCCCGAGTCCACCAGTGGGGCTATTTCCCGAGC
AGCAGAGA
(...)
```

Figure 1. Retrieved information from Entrez query perform. The sequence of characters (...) indicates information edition to readability of the figure.

The Figure 1 shows all retrieved information NCBI's website, using **efetch.fcgi** function, performing a service provided by the Entrez system.

It is possible see two FASTA[1] format sequences recovered by the query.

As can be seen, in the query you must inform identification parameters on what you want to look for, such as the part of the query

id=34577062,24475906

which determines two **GI** numbers.

The **GI** number is "Genbank identifier", that is another kind of NCBI unique identifier. In the specific case, only in this way it was possible to retrieve other information about the sequence, including the sequence itself.

2.1. Approaching the problem with BLAST program

If there was no knowledge of these sequences, the BLAST software could be used to find the first information about them.

BLAST is an acronym for Basic Local Alignment Search Tool and this software is available to use at the NCBI site in web version. Although BLAST originated at the NCBI, there are other bioinformatic websites, as well as the NCBI, provide bioinformatic computational resources, including BLAST software.

The BLAST software is able to sequence alignment and find similarity between them. Seting up similarity between sequences is a powerful tool for identifying the unknowns in the sequence world [5].

There are three ways to use the BLAST software: directly through NCBI's website, in the BLAST page, when it is possible submit a sequence or a list of sequence, "putting" one by one, or read all sequences from a FASTA file; the second way, it is necessary to get a version of BLAST program, e. g., available from NCBI website, and the particular BLAST file to the aligning process. To install the BLAST program on a local machine to run it using the BLAST file. In the third one, the BLAST process is running by remote call, like a remote procedure call (RPC) mode.

if the BLAST program was used directly from NCBI website, there are typical constraints of a website and web access. In the second case, if the choice is acquire the BLAST program and making

---

[1]The FASTA format is currently adopted as the standard format for genomic sequence data. The FASTA format consists of a *definition line* – which is a single line that begins with ">" – and the sequences lines, which are put immediately below to definition line. Can be used as many sequence lines as needed, but they should be limited numbers of characters [5].

download of the specific BLAST database necessary to the align, certainly a very good network, stable and with large bandwidth, it is essential, because the download of any BLAST database requires severe and large load data transmission.

2.2. Using Perl to access BLAST program from remote procedure call

Perl is the acronym to Practical Extraction Report Language and it is a programming language with many features to access databases, website developing, regular expression evaluation etc.. Originally, it was developed to text handling and today it keeps all of features, but is used for numerous kinds of different applications [6, 7].

One of the advantages of Perl is the CPAN (Comprehensive Perl Archive Network) website repository. The CPAN website stores a set of over 117 thousands Perl modules, for several distribuitions [8], organized, documented and free to be used by any developer.

In the CPAN repository it is possible to find several resources to build solutions for accessing remote database or others computational resources, including modules for distributed computing. For instance, the LWP [9], "The WWW library for Perl", enables the development of distributed computing procedures over HTTP protocol.

As it is known, to bioinformatics and computational biology the BioPerl module [10, 11] is the best resource provided from Perl. The BioPerl project "is an international open-source collaboration of biologists, bioinformaticians, and computer scientists (...)" [10] which developing a set of resources, as a "toolkit of Perl modules", to provide appropriate features to help bioinformatics application programmers to develop useful programming interface to access other resources, like database or web services.

The BioPerl module has several procedures to able access NCBI's services, and it is can run the BLAST program from remote call, among other procedures. Therefore, from Perl it is possible build a distributed system in the architecture client/server, executing remote calls, over HTTP protocol.

3. Genomic Information Mining and Retrieval

This paper shows how the Perl language can be used to get metainformation about a sequence, from

```
(...)
# blastRemoteCall.pl
#
# Runs remote BLAST from NCBI, over HTTP protocol, using
# the methods of BioPerl module. Returns full information
# about searched sequence, if it was found. Currently
# works only with nucleotide sequences.
(...)
# perl blastRemoteCall.pl input_file_name output_base_file_name
(...)

use Bio::Tools::Run::RemoteBlast;
use Bio::SearchIO;
use Data::Dumper;
(...)

$prog = "blastn";
$db = "nr";
$e_val = "1e-10";

my @params = ( '-prog' => $prog,
               '-data' => $db,
               '-expect' => $e_val,
               '-readmethod' => 'SearchIO' );

my $remoteBlast = Bio::Tools::Run::RemoteBlast->new ( @params );
(...)

$remoteBlast->submit_blast ( $inpFile );

while ( my @rids = $remoteBlast->each_rid ) {

  foreach my $rid ( @rids ) {
    my $rc = $remoteBlast->retrieve_blast ( $rid );

    if ( !ref ( $rc ) ) {

      if ( $rc < 0 ) { $remoteBlast->remove_rid ( $rid ); }

      print STDERR ".";
    }
    else {
      my $result = $rc->next_result();

      my $outFile = $outBaseName.$result->query_name()."\.txt";
      $remoteBlast->save_output( $outFile );
      $remoteBlast->remove_rid( $rid );

      print "\nQuery name: ", $result->query_name(), "\n";

      while ( my $hit = $result->next_hit ) {
        print "\thit name is ", $hit->name;

        while( my $hsp = $hit->next_hsp ) {
          print " (score ", $hsp->score, ")";
        }

        print "\n";
      }
    }
  }
}
exit;
```

Figure 2. Listing of blastRemoteCall.pl script. The sequence of characters (...) indicates information edition to readability of the source code or to anonymity of authorship of the paper.

NCBI website, without any a priori information on it. In particular, to this work, was used Perl with

BioPerl module to call BLAST remote procedure, as previously explained (Sections 2.1 and 2.2).

As can be seen in the source code of blastRemoteCall.pl script, listing at Figure 2, were used the modules Bio::Tools::Run::RemoteBlast and Bio::SearchIO, which form part of the BioPerl project, and Data::Dumper.

The adopted strategy for blastRemoteCall.pl script approach was to get unknown sequences in a the FASTA format and submitted them to NCBI's BLAST program from remote call. The Bio::Tools::Run::RemoteBlast module, in this script, is the most important for script strategic role and its need to use, because the method which performs the remote BLAST is implemented in the Bio::Tools::Run::RemoteBlast module.

The necessary parameters to run the remote BLAST are setting to **@params**, which it is used to instance a new "object" nominated **remoteBlast**. These parameters assign, among others features, the "kind" of BLAST will be used and on which database will be done the search. In the blastRemoteCall.pl script, these parameters received **blastn** and **nr** values, specifying the BLAST program and database for nucleotides, respectively.

Next, the BLAST submission, actually, remote submission is done with the sequences to be investigated sent in FASTA files (Figure 3) as a submission parameter, and each the BLAST process for each sequence received a unique identification, known as **rid**, which are stored in **@rid**.

```
>
GGAAGGGGCGTGGCCTCGGTCCGGGGTGGCGGCCGTTGCCGCCACCAGGGCCTCTTC
CTGCGGGCGGTGCTGCCGAGGCCGGCCTGCGCGGGGCAGTCATGGTACCCCCTTGAG
CGGGCTGTGGCGGAGAGCGGGGCGGGGACTGGCTGGAGGGTGGCGGCCCGGCGGGGC
GGGGGCGGGGCCGGCCTCTGGCTCCTTCTTCCTCTGCATGTGGCTGGCGGCCGCAGA
GCAGTTCAGTTCGCTCACTCCTCGCCGGCCGCCTCTCCTTCGGGCTCTCCTCGCGTC
ACTGGAGCCATGGCGTTCGCCGAGACCTACCCGGCGGCATCCTCCCTGCCCAACGGC
GATTGCGG
(...)
```

Figure 3. FASTA sequence file sample for BLAST remote submission on blastRemoteCall.pl script. The sequence of characters (...) indicates information edition to readability of the figure.

One by one, the BLAST process are performed and their outputs are tested to check if is valid the returned content.

Could be observed in example (Figure 3), the definition line – marked with '>' – is empty, because its have no information about the sequence.

4. Approach Analysis

The subject of this article proposes to make remote access on NCBI's website to use their web services and perform BLAST program to get information about unknown sequences in FASTA format. Particularly, these sequences could be nucleotide sequences, as a DNA or RNA sequences.

To implement the remote procedure call was developed a Perl script, which uses modules of BioPerl project to set up the BLAST parameters; executes and submits the BLAST program to mining the database; and retrieve the answer.

The means adopted to implements this strategy was process remote calls as a RPC. The RPC are a way to implement distributed systems and use explicit calls to execute send and receive procedures [12, 13].

The interaction between send and receive procedures – or *message passing* between them, as it is technically called this kind of interaction in RPC systems – define the type of communication between the client, which performs the request, and the server, which replies to the request.

The client and server doing *synchronous communication* if the client waits for the server response, but if the client performs a request and continues processing, without waits for the reply, then they doing *asynchronous communication* [12, 13].

Clearly, the blastRemoteCall.pl script implements asynchronous communication mode. Therefore, as the BLAST server is able to receiver and handle each request independently of the others and is not necessary for it waits for the reply, then the entire process becomes more efficient.

The method of communication implemented with distributed system concepts creates a lightweight and efficient message passing between the Perl script, client application, and the server on NCBI's website.

5. Conclusions

This paper introduces and analyzes an intelligent retrieval for metadata of unknown sequences with a Perl script using Bioperl and other modules as well as resources of the NCBI website.

Search for metadata about sequence fragments can be necessary many times on bioinformatic and computational biology tasks and the NCBI website is an important place to do prospecting like these and it offers many efficient tools to do it.

However, almost all NCBI's tools need a kind identifier for the sequence which look up its metadata, usually one of NCBI's unique identifier, like the **GI** number.

Nevertheless, it is not uncommon encounter complete sequences or fragments of sequences without identifiers or knowing any information about them. Therefore, many NCBI's tools are not as useful in these searches, but the solution proposed shows another approach.

The search for metadata is done from remote execution of the BLAST alignment program, enabling the recovery all the information found by search of similarity among sequences.

Acknowledgments

References

[1] NCBI. (2013, Jan.) Our mission. [Online]. Available: http://www.ncbi.nlm.nih.gov/About/glance/ourmission.html

[2] ——. (2011, Dec.) Entrez help: NCBI bookshelf. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK3837/

[3] ——, *The NCBI Hanbook: Glossary*, J. McEntyr and J. Ostell, Eds. Bethesda: NCBI, 2002. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK21106/

[4] E. Sayers, "E-utilities quick start," in *Entrez Programming Utilities Help*, J. McEntyr and J. Ostell, Eds. Bethesda: NCBI, 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK25500/

[5] I. Korf, M. Yandell, and J. Bedell, *BLAST*. Sebastopol: O'Reilly & Associates, Inc., 2003.

[6] T. Christiansen, B. Foy, and L. Wall, *Programming Perl*, 4th ed. Sebastopol: O'Reilly Media, Inc., Feb. 2012.

[7] Perl.org, "The perl programming language," Website, 2013. [Online]. Available: http://www.perl.org/

[8] ——, "The comprehensive Perl archive network," Website, 2013. [Online]. Available: http://www.cpan.org/

[9] G. Aas and M. Koster, "The WWW library for Perl," Website, 2009. [Online]. Available: http://search.cpan.org/perldoc?LWP

[10] J. E. Stajich, D. Block, K. Boulez, S. E. Brenner, S. A. Chervitz, C. Dagdigian, G. Fuellen, J. G. R. Gilbert, I. Korf, H. Lapp, H. Lehvaslaiho, C. Matsalla, C. J. Mungall, B. I. Osborne, M. R. Pocock, P. Schattner, M. Senger, L. D. Stein, E. Stupka, M. D. Wilkinson, and E. Birney, "The Bioperl toolkit: Perl modules for the life sciences." *Genome Res*, vol. 12, no. 10, pp. 1611–1618, Oct 2002.

[11] BioPerl Core Developer, "BioPerl," Website, Aug. 2012. [Online]. Available: http://www.bioperl.org/

[12] J. Dollimore, T. Kindberg, and G. Coulouris, *Distributed systems: concepts and design*, 4th ed. Addison Wesley, 2005.

[13] A. S. Tanenbaum and M. Van Steen, *Distributed systems: principles and paradigms*, 2nd ed. Prentice Hall, 2006.

Evaluation of musical similarity on the symbolic level of the musical text

Michele Della Ventura

Department of Technology, Music Academy "Studio Musica", Treviso, Italy

Abstract - *This article presents a musical text analysis method based on the concept of similarity. The notion of similarity, in music, is particularly complex, in that it can be noticed in its various dimensions: melodic, rhythmic and harmonic. The mere repetition of a segment does not exhaust similarity: similarity has a meaning that emerges from the structural properties of the musical composition. This means that there is similarity even when certain common features of the "model" are identified in one or several points of the musical piece. In this article, similarity was examined starting from the structural elements of music, i.e. melody and rhythm. Graphical examples of analysis are shown: the similar segments are displayed separately, facilitating the identification of melodic or rhythmic structures. When there are several segments, their classification is performed using the Hidden Markov model.*

Keywords: analysis, musical object, musical similarity, musical surface, segmentation

1   Introduction

The concept of similarity in musical analysis involves the concept of comparison: it is through collation that analysis defines the structural elements of a composition and reveals the functions thereof. To collate two musical objects [1] means to search for the common features of the two objects. The similarity of two objects can be searched for in the melodic line and/or in the rhythmic structure, that is in the constitutive elements of music [2].

The concept of similarity has been a study object for various scientists, every single one of whom tried to analyze it from a different perspective: yet, all the cases deal only with melodic similarity. The purpose of this article, instead, shall be to try to analyze similarity taking simultaneously into consideration both the concept of melody and the concept of rhythm.

Generally speaking, a melodic or a rhythmic cell may be represented by a vector of this type:

$$\boldsymbol{v} = (v_1, v_2, ..., v_n)$$

where v_1, v_2, ..., v_n are respectively the single intervals or durations (see fig. 1) of the notes that compose the melody [1].

$$\mathbf{v}_i = (2, 2, -2, -2, 2, 2, -4)$$
$$\mathbf{v}_r = (3, 1, 4, 1, 1, 1, 1)$$

Fig. 1: Representation of the interval vector (vi) and of the rhythmic vector (vr). For the Interval Vector, the semitones separating a sound from the next sound are indicated: the ascending intervals are indicated with a positive number while the descending intervals are assigned a negative number. For the rhythmic Vector, the durations separating a sound from the next sound are indicated: every duration is a (whole) number which is directly proportional to the smallest duration existing in the musical segment.

There emerges, from what has been said above, that the research of similarity between two musical segments may be performed by examining the correspondence existing at a melodic or rhythmic level, respectively, between the intervals or the durations but also considering these two components simultaneously.

Fig. 2: Example of a musical segment: compared to the one in figure 1, it displays an (apparently) different melodic line, yet the same rhythmic structure.

For this purpose, every single musical segment *M* shall be represented by a matrix [3] of this type:

$$A_{x,y}$$

where *x* (number of rows) equals the number of intervals and, therefore, durations separating the single sounds that make up the musical segment *M* and *y* (the number of columns) equals 2 because the elements taken into consideration are two: the pitch of every sound and its duration (fig. 3).

$$A_{7,2} = \begin{pmatrix} 2 & 3 \\ 2 & 1 \\ -2 & 4 \\ -2 & 1 \\ 2 & 1 \\ 2 & 1 \\ -4 & 1 \end{pmatrix}$$

Fig. 3: Example of a matrix associated to the musical segment in figure 1.

In this last type of analysis, the similarity between two segments shall be evaluated by identifying the correspondences existing at a rhythmic and melodic level, based on the very alphabet [4] of the segment taken as a model: in case of several areas of correspondence, a classification from a probabilistic point of view of such areas shall be performed, using the Hidden Markov Model. The results are represented as a diagram so as to facilitate the evaluation.

This paper is organized as follows.

Section 2 describes the Hidden Markov Model. Section 3 describes the graphic representation. Section 4 shows an experimental test that illustrates the effectiveness of the proposed method. Finally, conclusions are drawn in Section 5.

2   Hidden Markov Model (HMM)

The Markov chains are a stochastic process, characterized by Markov properties.

It is a mathematical tool according to which the probability of a certain future event to occur depends uniquely on the current state [5]. Let $X_n = i$ be the current state and $X_{n+k} = j$ the state after k steps, with i, j belonging to the set of states.

The conditional probability $P[X_{n+k} = j | X_{n = i}]$ is called a transition probability in k steps of the Markov chain [6].

The probability of transitioning from state i to state j in k steps of a homogeneous chain is indicated by:

$$p_{i,j}(k) = P[X_{n+k} = j | X_n = i]$$

By tagging the states as 1, 2, ...$n+1$ we can summarize all the transition probabilities, $p_{i,j}(k)$, in a matrix $P(k)$, of the dimension $n \times n$, where in the j-th column and the i-th row there is the transition probability from state i to state j in k steps:

$$P(k) = \begin{pmatrix} p_{1,1}(k) & \dots & p_{1,j}(k) & \dots & p_{1,n}(k) \\ p_{2,1}(k) & \dots & p_{2,j}(k) & \dots & p_{2,n}(k) \\ \dots & \dots & \dots & \dots & \dots \\ p_{i,1}(k) & \dots & p_{i,j}(k) & \dots & p_{i,n}(k) \\ \dots & \dots & \dots & \dots & \dots \\ p_{n,1}(k) & \dots & p_{n,j}(k) & \dots & p_{n,n}(k) \end{pmatrix}$$

The matrix $P(k)$, for $k = 1$, performs a fundamental role in Markov's chains theory: it (known as the transition matrix) represents the probability of transitioning to the next consecutive step [7].

It is possible to represent the transition matrix P by means of a graph called transitions diagram [8] [9]. The latter consists in a graph the nodes of which represent the single states while the arcs, oriented and labeled with the probability, indicate the possible transitions.

For instance, considering the matrix

$$P(k) = \begin{pmatrix} 0,7 & 0,1 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,3 & 0,3 & 0,4 \end{pmatrix}$$

it has the following corresponding diagram (Fig. 4):
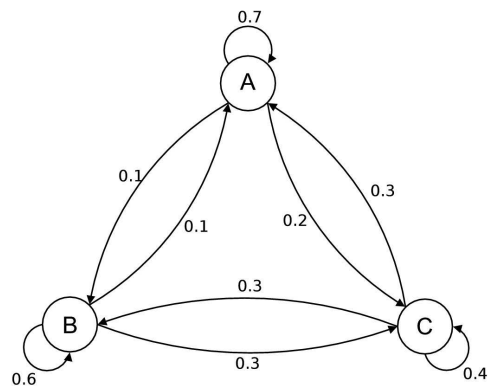
Fig. 4: State-transitions in a Hidden Markov Model.

The problem of classification of the sequences may be solved by calculating the probability of the single sequence s to be emitted by the model M: $P(s / M)$.

Formally:

$$P(s / M) = \sum \pi \, P(s, \pi / M)$$

3   Graphic representation

For the analysis of similarity the decision was made to use the Dot-Plot Matrix graphic representation [10].

Let us consider a matrix in which the first upper row coincides with *sequence1* (model) written from left to right and in which the first column on the left coincides with *sequence2* written from top to bottom; every single point in which the sequence in the row and in the column coincide is marked with an asterisk. The regions of the sequences that may be aligned emerge as a contiguous series of points on a diagonal (from upper left to lower right).

Given the sequences of notes *A* and *B*,

where sequence *A* represents the model and sequence *B* the one to collate, from the Dot-Plot Matrix associated to the

melodic analysis (see fig. 5), it emerges that the similarity between the two sequences is identified in the second beat
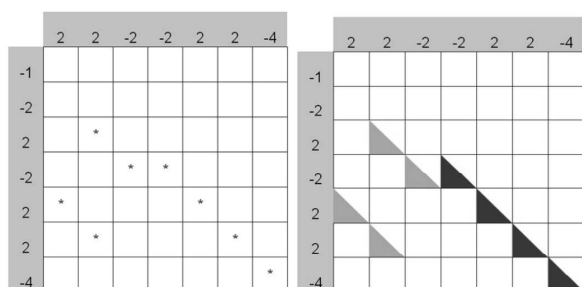
Fig. 5: Dot-Plot Matrix associated to the melodic analysis. The asterisks were replaced by colored triangles so as to have a better visual interpretation. The intervals that can be found in the same positions within both sequences are indicated in dark-gray. The intervals describing the evolution of melody in the model sequence are indicated in the first upper row; the intervals describing the evolution of melody in the second sequence are indicated in the first column on the left. The sign + or the sign – indicate, respectively, whether an interval is an ascending or a descending one. The adjacent squares (dark-gray and light-gray) indicate the melodic segments that feature common similarities, within the two sequences.

The same procedure may be applied in order to search for rhythmic similarities (fig. 6): in this case, the durations that separate a sound from the next sound in the model sequence shall be indicated in the first row, while the durations that separate a sound from the next sound in the second sequence shall be indicated in the first column on the left.
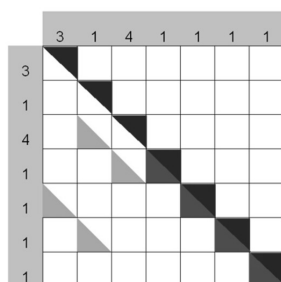
Fig. 5: Dot-Plot Matrix associated to rhythmic analysis combined with melodic analysis. The duration that separates a sound from the next is determined on the basis of the smallest duration (♪) existing in the musical segment. The adjacent squares (in black) indicate the rhythmic segments that feature common similarities, within the two sequences. There emerges, from this analysis, that the two sequences are similar at a rhythmic level.

4 The results obtained

The model of analysis set forth in this article was verified by realizing an algorithm the structure of which takes into consideration each and every single aspect described above as well as the *nature of data*: the results of the analysis are indicated by means of diagrams (Dot-Plot Matrix) that allow an immediate visualization for interpretative purposes.

An example of similarity analysis is given below, taking as model sequence a melodic sequence excerpted from Invention (two voices) in E Major, BWV 777 by J.S. Bach (fig. 6), conveniently modified as far as rhythm is concerned, so as to facilitate comprehension of the analysis.

$v_i = (-1, -1, -1, -2, -2, -1, -2, 2, 1, -1, 3, -2, 2, -3, -2, 2, -4, -1)$
$v_r = (4, 4, 4, 4, 4, 4, 4, 4, 2, 2, 2, 1, 1, 2, 1, 1, 2, 1)$

Fig. 6: Invention in two voices in E Major, BWV 777 by J.S. Bach (first three beats of the first staff) and the corresponding interval vector (v_i) and rhythmic vector (v_r).

Figure 7 shows a melodic segment compared to the model in figure 6.

$v_i = (-1, -1, -1, -2, -2, -1, -2, -2, 2, 2, -6, -1, 1, 6, -2, 2, 1, -1)$
$v_r = (4, 4, 2, 1, 1, 2, 1, 1, 4, 4, 4, 4, 4, 4, 2, 1, 1, 2)$

Fig. 7: Compared melodic segment and its related interval vector (vi) and rhythmic vector (vr).

It can immediately be noticed, from the graphic analysis (fig. 8), that the two different segments, feature, in fact, certain similarities that get them closer to each other.
At a melodic level they present two similar segments (dark-gray color), of which only one is relevant: it is segment S_0 made up of 8 sounds (therefore 7 colored triangles). In musical analysis, the melodic (or rhythmic) cell, made up of two or three notes (or durations), represents the minimum information unit insofar as it can be identified in the composition many times.
At a rhythmic level (black and light-gray colors) there emerge, between the two sequences of notes S_1 and S_2, two aligned segments (black color), that represent the minimum information unit, and several non aligned segments (light-gray color): i.e., they are not located in the same positions in each sequence.
In order to have a classification of the various existing segments, the probability of every single segment was calculated, in reference to the alphabet of the model sequence and therefore to the transition matrix according to Markov's

process. The two equally long segments S_1 and S_2, highlighted in figure 8, have the same number of sounds and, respectively, the following probabilities:

$PS_1 \cong 0.21596$

$PS_2 \cong 0.01480$

If expressed in logarithmic form:

$PS_1 \cong 2.21115$

$PS_2 \cong 6.07800$

of which it can be deduced that the segment S_2 (fig. 9) has greater significance than the segment S_1 insofar as it spotlights a well-defined rhythmic cell (see fig. 9).
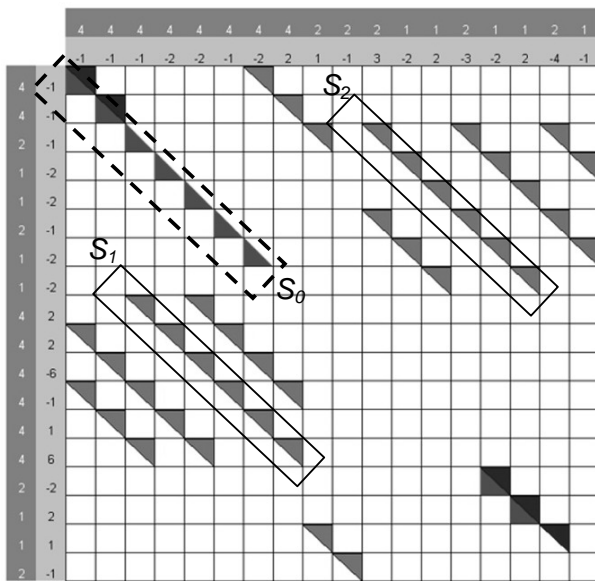
Fig. 8: Dot-Plot Matrix associated to melodic and rhythmic analysis. In the first upper row and in the first column on the left are indicated the durations that separate the various sounds, respectively of the model sequence (upper part) and of the compared sequence (on the left); in the second upper row and in the second column on the left are indicated the intervals between the various sounds, respectively of the model sequence (upper part) and of the compared sequence (on the left).
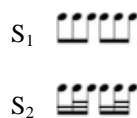
Fig. 9: Rhythmic representation of sequences S_1 and S_2. The sequence S_2 is made up of a rhythmic cell (⬛) repeated twice, consisting in conveniently distributed signs having a different duration, that bestow on it a distinctive character compared to other rhythmic sequences (see S_1).

5   Conclusions

This article examined the notion of similarity within the framework of musical segments, in relation to the notions of melody and rhythm. It highlighted the fact that they are two inseparable "dimensions".

The problem of classification related to the over-segmentation was also approached, i.e. the identification, within the compared musical sequences, of several segments having equal (that is, similar) features: a problem that was solved by means of probabilistic calculus related to the intrinsic structure (interval and rhythmic structure) of the very same musical sequences.

This method of analysis might be applied to the study of contemporary music compositions (in which it is more difficult to recognize the information transmitted by the author) and extended to studies related to Automatic Speech Recognition   (ASR), especially to the enunciation not of words, but of musical phrases.

6   References

[1] M. Della Ventura. "Analysis of algorithms' implementation for melodic operators in symbolical textual segmentation and connected evaluation of musical entropy"; in Proceedings of the International Conference on Mathematics (IAASAT '11) (pp. 66-73), Drobeta Turnu Severin, Romania, Oct 2011.

[2] M. Della Ventura. "Rhythm analysis of the"sonorous continuum" and conjoint evaluation of the musical entropy"; in Proceedings of the International Conference on Acoustic & Music: Theory & Applications (AMTA '12) (pp. 15-21). Iasi, Romania, Jun 2011.

[3] M. Della Ventura. "The concealed motif: analysis and identification"; in Proceedings of the International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED '13) (pp. 44-48). Cambridge, England (UK), Feb 2013.

[4] M. Della Ventura. "L'impronta digitale del compositore"; GDE, Italy, 2010.

[5] J. Norris. "Markov Chains"; Cambridge University Press, (1997).

[6] E. Angeleri. "Information, meaning and universality"; UTET, Turin, 2000.

[7]   Y. Bengio. "Markovian Models for Sequential Data"; Neural Computing Surveys 2 (pp. 129-162), 1999.

[8]   D.A. Bini, G. Latouche, B. Meini. "Numerical Methods for Structured Markov Chains"; Oxford University Press 2005.

[9]   L. R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition"; Proceedings of the IEEE, 77 (2), pp. 257-286, 1989.

[10] A. J. Gibbs, G. A. Mcintyre. "The Diagram. A Method for Comparing Sequences"; European Journal of Biochemistry (pp. 1-11), 1970.

SESSION

ARTIFICIAL INTELLIGENCE, ALGORITHMS, NOVEL APPLICATIONS AND CONCEPTS

Chair(s)

TBA

Crowd Evacuation Control for Public Spaces

Pejman Kamkarian[1], and Henry Hexmoor[2]

[1] *Electrical and Computer Engineering Department, Southern Illinois University, Carbondale, IL 62901, USA*

[2] Computer Science Department, Southern Illinois University, Carbondale, IL 62901, USA

pejman@siu.edu,hexmoor@cs.siu.edu

Abstract - *This paper demonstrates a simulated solution for evacuating people from a public place using the Coulomb laws. In an emergency, a plan to guide people away from danger in the shortest and safest possible ways is essential. We employed the Coulomb laws to divide the public space into safe boundaries, based on the number of people inside. This is a dynamic solution, which means, it will change the safety boundaries of a public space based on dynamic crowd movements. Security personnel will use the latest patterns of safety boundaries to assist people to evacuate through exit doors more efficiently.*

Keywords: Social network, virtual team, virtual organization, non-traditional organization, spontaneous and networked organization.

1 INTRODUCTION

It is important to understand crowd control in order to prevent unsafe behaviors, especially in emergency situations that may lead to injury or death. When an emergency occurs, occupants often panic and will attempt to exit using the nearest doors. According to Canada's NRC scientist Guylene Proulx, the activities and the interactions among the occupants in a building before the occurrence of an emergency are relevant to understanding the nature of panic, which can be used to mitigate hysterical or irrational behaviors as stated in the crowd definition and supported in literature.

In normal circumstances, people behave rationally. For example, people who arrive together will also tend to depart together. Family members will remain together before making the decision to exit [14], [8], [2]. A few people depending on their role (e.g. owners and employees) may decide to leave the building first, but at some points will return to retrieve items left behind (e.g., documents, money, and people in danger). The irrationality of their actions is evident only in a retrospective view, but at the time of the action it was determined to be perfectly rational. Evacuees are assumed to be rational, so after realizing that they need to escape, they should make decisions that would lead them to exit as quickly as possible [15]. According to the information gathered, people have to select the exit that can guarantee them the greatest opportunity to escape from the threat as fast as possible [1]. Exit selection has a great influence in the outcome of an evacuation strategy. How

people choose exits is a complex process. Generally, they make their decision based on their familiarity with the exit and its visibility [11]. The attributes affecting the exit selections include the distance to the exit, the behavior of other agents, the exit's visibility, and the familiarity of the exits. A building generally consists of enclosed areas with one or more doors. Many theories and researches have attempted to explain and model the problem of finding a way with the main goal to select a way out. Different methods are used to model it. Among those, Cellular automata models consider the variety in the environment (e.g., exit width and obstacles) to have an impact in the exit dynamics of evacuees [18], [13]. Evacuees tend to prefer familiar alternatives, because they feel that unknown alternatives increase the threat. The visibility of exits also influences their decisions. According to personality trait theory [14], [8], [3], [17], [6], these models demonstrate the influence of evacuees' personalities (e.g., shy, aggressive, collaborative) in the outcome of evacuation. A shy person would reluctantly make decisions but he would act as a "follower", an aggressive person would exhibit "selfish behavior", and a collaborative person would try to cooperate with others to come out with a suitable solution for all Agent-based models [5]. In these models, evacuees act as "rational" agents whose objectives are to find the best set of actions that will maximize their progress toward the exit doors, relying on partners that will help them reach safe places in a minimum amount of time. Finally, the choice of an exit door will depend on the interaction between evacuees and their environment. An evacuee will pre-select a route based on his knowledge of the environment, and that initial route may change depending on his estimation of queuing time, traveling time to that particular exit, and sometimes his response to group decisions [16].

An evacuation crowd consists of individuals that interact with each other [8]. Occupants have to escape from the danger as quickly as possible, and by doing so they may have to collaborate with others. Evacuees are assumed to be rational. They pursue their own interests [7]. Evacuees will tend to cooperate with another one if they estimate that they can have a good payoff (maximum chance of being safe), or in contrary will avoid cooperating or associating with another one if their safety is decreased. They might have to be collaborative with others, or try to develop skills (e.g., exploring and visual memory). The urgency theory is used to explain some behaviour of occupants, for instance blockage at exit doors, stampeding or trampling [4]. Three

attributes essential to understanding the concept of urgency are [12].

1. The nature of the emergency
2. The consequence of not exiting quickly
3. The time available to exit

By and large simulations of crowds are either macroscopic by modelling environmental parameters that affect a group of agents simultaneously, or microscopic by modelling interaction rules for each agent. A meso-level modelling is offered in [9], and [10] where movement fields are composed of external inputs that influence agents combined by their own local reasoning. In this paper, we demonstrate a strategy to repeatedly divide the environment into distinctly safe boundaries around each exit door. Boundaries are subject to change based on the location of each agent at any moment using a crowd evacuation model. We will also focus on the relationships between locations of exit doors and agent movements in the environment to develop our strategy accordingly. In the following section, we will outline pertinent environmental attributes.

2 ENVIRONMENTAL ATTRIBUTES

Generally, each environment will consist of many different groups of objects such as obstacles (e.g., a row of chairs or trash cans), the agents and the exit doors. Agents can be any type of living beings such as pets or human. In this paper, we assume that only human agents will exist in the environment. The two most important attributes in the environment for us are the agents (i.e., simulating individuals) and exit doors (i.e., evacuation points). These attributes affect each other and are used to determine safe boundaries. In this paper we considered each environment generally having one or more convex zones corresponding to the map. Each of the convex zones is composed of a collection of α zones, which are the spheres belonging to each exit door, with β zones and θ angles for each. Each convex zone must have at least a single exit door inside. If there are no exit doors available for a convex zone, we have to merge it with its adjacent convex zone, which has at least one exit door. Section 3 and section 4 offer remarks about exit door attributes. In section 5 we outline the application of Coulomb's Law.

3 EXIT DOOR ATTRIBUTES

We considered three general attributes for each exit door: zones, boundaries, and θ angles. We assumed three different zones for each exit door. The first zone is the area nearest to the exit door we call α zone. This is determined by each exit door's width. Figure 1 shows two exit doors of different widths and their proportionally sized α zones. This zone is always the same in size and never changes during the movement of each agent. The only way that an α zone could change would be if it is completely blocked or the door width is partially blocked (for instance, by some obstacle that might be a fallen agent, or debris in the

environment). In such cases, due to the change in width of the exit door, the α zone will change. Each environment may contain several α zones, one for each exit door. The second zone is called β zone (see Figure 1). The β zones are the areas that are bound by the walls, and each contains a single α zone. Areas between several α zones are divided into the same number of β zones. The third attribute is the θ angle, which is the largest angle that contains a corresponding β zone within each quadrant. Boundaries separate β zones. The θ angle varies based on the locations of agents in each moment. For example, the angle can be reduced if the number of people around a certain exit door increases. Figure 1, shows an environment consisting of three exit doors with corresponding zones and angles.
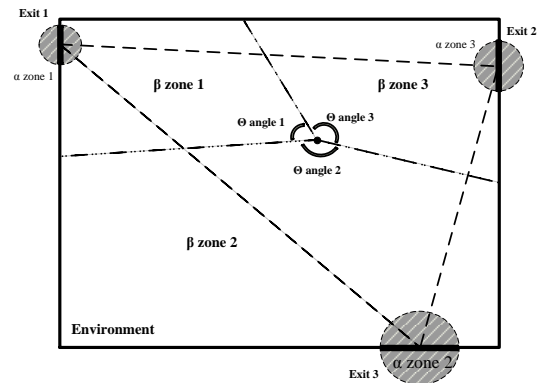
Figure 1: The α and β zones and θ angles related to exit door 1 and 2.

4 COMBINATIONS OF EXIT DOORS

As a matter of standard, each building should have at least one exit door located at a place that is visible to the public. In the case of a single exit door in the environment, the only way to evacuate people will be to guide them to the exit door and through it, and hence there is only one α zone and the rest of the environment belongs to one β zone. There will be no θ angle, which means that there are no safe boundaries available. Our solution is applicable only if we have more than two exit doors at each convex zone. If there are two or more exit doors, there will be a corresponding number of α zones, β zones, and θ angles respectively.

5 APPLICATION OF CUSTOMIZED COULOMB'S LAW

In this section, we demonstrate and apply the customized coulomb's law to exit doors based on the current locations of agents at each moment in time. We divided the process of dividing the environment and decision making of each boundary into four different phases. Initialization is the process of determining locations and status for each exit door to form the initial safety boundaries. Detection of

boundaries is another phase. There is a phase for gathering data from sensors. The sensors and detectors sense and detect the locations of agents and their physical specifications and movements by cameras and send information to the processing unit. Another phase is updating the boundaries and decision making, which entails updating and redrawing the safe boundaries based on the latest information gathered by the detector devices installed in the environment. Each phase consists of many other sub-steps. We use different techniques for each phase. The phases and relative steps are described next.

5.1   Initialization

This phase starts by detecting the location of each exit door and also measures the width of each exit door. We consider each exit door as a sphere with the diameter equal to the exit door width. We assume an abstract line drawn from each sphere's centre point to other sphere's centre points in such a manner as to form the largest enclosing polyhedral configuration. This configuration must meet each sphere only once. Because the goal of this step is to only determine the initial values and measures of locations and specifications for each exit door in the environment, to calculate and draw the initiate safe boundaries, we consider our environment to be empty; i.e., devoid of agents.

5.1.1 Initializing the exit doors

At this level, all exit doors located inside the environment will have been detected possibly using a raster map. We need to have two different measures for each. The width and the central point of each exit door are placed on the central straight line that connects two edges of exit doors. This can be done by the cameras and detectors that are installed in the environment. We may use some techniques to increase the speed of detecting and measuring locations of exit doors, such as using a wire frame viewing of the environment.

5.2   Determining the sphere of charges

We assumed that each exit door behaves as a sphere of charge with the diameter equal of the width called α zone. The α zone may be considered to have a simple 2D circular shape or a complete 3D spherical shape. In this paper we considered the α zones as a complete 3D shape. The amount of charge for each α zone is equal to the relevant exit door charge. If an exit door is blocked for any reason during monitoring the environment, we must consider its sphere, and hence its charge, to be 0.

5.3   Determining zones

At this level, we divide our environment into different convex zones. To do this, we use convexity concepts and rules that have previously been used in the neural networks. On the other hand, the environment must be divided into convex shapes such that each default line must meet the edges of the shape only two times. Each convex area must have at least one simple exit door inside; otherwise, we

have to join this zone with the adjacent convex zone which has at least a single exit door.

5.3.1 Determining the exit doors boundary

This point consists of determining the largest simple non directed cycle graph of length m where m is total number of exit doors located in each zone. This graph has m vertices and m edges. Every vertex has a degree of 2. We label this graph C_m. For example, in case of 5 exit doors among the convex zone, the largest simple, non-directed cycle graph has the length equal to 5, and also we have 5 different pairs of adjacent charges.

5.3.2 Determining the amount of charge for each exit door

We considered each exit door as a sphere with the diameter equal to the width of each exit door. Based on the metric measurement, the amount of positive charge for each exit door is given by the following equation 1:

$$Q_{ED_i} = \lceil W_{ED_i} \times 100 \rceil \qquad (1)$$

Where Q_{ED_i} is the charge of ith exit door and W_{ED_i} is the width of ith exit door. For instance, in case of an exit door with the width of 1 meter, the charge of the mentioned exit door would be 100.

5.4   Detection of Boundaries

At this phase, the process will determine the initial boundaries for each exit door using the initial values gathered in the previous phase. To determine the initial boundaries, this phase will find the point which is located on the line that connects the centres of each pair of exit doors (as charges). The location of this point obtained by considering the width of each exit door, then drawing a virtual vertical page crossing that point. This phase bisects the intersectional boundaries between different vertical pages to reach β zones.

5.4.1 Computing the charge for each adjacent pair of exit doors

At this step, we will calculate the amount of charge between each adjacent pair of exit doors. To do this, we assumed a straight line between each pair of adjacent exit doors. We also assume that the locations of all exit doors are fixed and that the charges for each exit door are positive. We will put a positive charge on the straight line between each adjacent exit door. The amount of the positive charge is given by the following equation 2:

$$Q_t = \left| \frac{Q_{ED_i} + Q_{ED_j}}{2} \right| \qquad (2)$$

Where Q_t is the amount of charge between two charges Q_{ED_i} and Q_{ED_j} is located at the adjacent pair on the mentioned straight line. We assumed that the positive

charge used between each pair is the average value for them; using this strategy guarantees that the positive charge will stay somewhere between the two fixed charges and not go beyond them.

5.5   Determining equilibrium point

At this step, the process will find the equilibrium location for each positive charge that is placed on the straight line between each pair of positive charges. All exit doors as positive charges are fixed. The positive charge will be located on the straight line and closer to the smaller positive charge. Since the two charges around the positive charge are positive, they will push the positive charge away from themselves. In such a case, the positive charge will stay closer to the smaller positive charge because the grater positive charge has a larger exert force than the smaller one. The equations demonstrated in the three electric charges in equilibrium are used to determine the location of the mentioned positive charges.

5.6   Finding the centroid point for each shape

At this step, the process finds that the geometric centre (i.e., centroid point) for each shape that contains related exit doors. The centre of mass, or centroid of a 2D shape, is the intersection point of all straight lines that divide the shape into two areas with equal movement about the line. The centroid point is the arithmetic mean of all intersection points. The next step is to draw a line from that location to all positive charges and continue that line until it reaches an environment. This line separates the environment into two different areas such that each of them belongs to a different exit door.

5.7   Determining and binding β zones

Each exit door has a θ angle that belongs to it, which is drawn from the centroid point to the equilibrium points of the straight lines crossing from the relative adjacent charges. At the next point, we bind the largest θ angle which includes only a single exit door as the β zone of the relative exit door. The process continues until all β zones are specified for each exit door.

5.8   Data Gathering and Analysis

There should be several sensors, cameras, and detectors installed at different locations in the environment in order to be able to determine the location of each exit door, especially the location of the agents at each moment. These facilities estimate the status of each exit door in each instant in terms of evacuation ability rate. These devices will send pertinent data for agents in terms of their size and an estimate of their movement speeds to the central unit in order to classify analyses and makes decisions. This phase generally gathers and analyses the data obtain by different sensors and detectors installed inside the environment.

5.8.1 Determining the agent locations

The crowd is dynamic and changes locations consistently. The rate of movement is more unpredictable when agents are faced with emergency situations. The process at this step will be to detect each individual's position in the environment. One way to determine the positions is by using a grid. In order to make decisions about the boundaries in real time, the sensors should be fast enough to determine individual locations and this information to the central processing unit for analysis.

5.8.2 Determining each agent's charge

The sensors and detectors should be able to determine the specifications of each individual, such as body sizes and individual movement rates. Having these measures we are able to assign an accurate value as a charge to each agent. At this step, the process will determine the amount of charge for each individual located in the environment based on physical specifications. Each general convex zone consists of a collection of distinct α zones, β zones, and θ angles. Each β zone has its distinct α zone and θ angle belonging to it. Furthermore, α and β zones and θ angles are non-overlapping among zones and angles.

The environment may consist of a number of general convex zones inside. Each general convex zone must consist of at least a single exit door. In case of having a convex area without an exit door, we will join it to its neighbouring zone that contains at least a single exit door.

Each zone has its own set of agents. Each agent may have a different situation in terms of the physical status.

Of all the agents available in each general convex zone, each β zone and exit door, depending upon their situations and locations, support a number of them.

We assumed that each agent has a negative charge based on the specifications that he/she possesses. The considered key features in this paper were age, sex, and health status. The amount of charge for each agent is given by the following equation 3:

$$Q_{Ag_i} = A_i \times G_i \times HS_i < 0 \qquad (3)$$

Where Q_{Ag_i} is the amount of negative charge for the ith agent, A_i is the age of the ith agent and the HS_i is the health status for the ith agent. Because all given values of the equation are negative, the final result of Q_{Ag_i} is always smaller than zero. In the case that there are no agents in any β zones, the total amount of charge for the mentioned zone will be considered to be 0. In order to have agents, the amount of charge in the β zone is relative to the amount of charge for the number of agents, and it is always smaller than 0.

The ranges of ages vary from place to place and depend on the usage of the environment and can be determined based on the average age of the majority of people inside. We consider the normal value to be (-1) and in order to have a reasonable result, we have to bind this value to the majority of people with the same range of age. For example, the usage of values in a kindergarten is different from a conference room because in a kindergarten the majority of people are children, so we may bind the normal value to the group of ages below 10 years old whereas in a

conference room, because the average range of age is between 20 and 40, we need to bind the normal value to this group of age.

The other key feature in terms of calculation of agent charges is the gender of individuals. The consideration values for each gender are different from situation to situation. In this paper, we divided agent charges into two categories of males and females. Males have a default value of (-1) and females have a default value of (-1.5).

The third physical key feature that we considered in this paper is health status. The health status may vary from place to place, depend on the environment usage, and is determined based on the health status of majority of the people inside. In this paper we considered having only two options: normal and disabled. In some places, like hospitals or elder houses, there should be other options available in order to have a better estimate of the charges for each agent. We considered a default value of (-1) for healthy status and (-2) for disable agents.

Apart from the physical specifications of each agent, focusing on the status of each exit doors is essential. We determine the amount of charge for each single exit door based on its situation at each time instant.

In different situations, considerations about the specifications for each exit door may vary. We always used the default rate for the best situation of exit doors when it is usable, reliable and can evacuate people to its full capacity. In this paper we assumed three status for each exit, including 'Still open and ready to use' with default value of (-1), 'Not stable' with value of (-1.5), and 'Partly blocked' with the value of (-2).

To obtain the new values for charges of exit doors, we have to consider the previous amount of charges and the latest safety status for each exit door. The total amount of positive charge is shown by the following equation 4:

$$Q_{ED_i} = \left| \frac{Q'_{ED_i}}{SR_{ED_i}} \right| \qquad (4)$$

Where Q_{ED_i} the total positive is charge of the ith exit door, Q_{ED_i} is the initial charge of the same exit door and SR_{ED_i} is the safety rate for the ith exit door. If the exit door is completely blocked or not usable, we have to consider its charge as 0.

The cameras and detectors will determine the safety rates of each exit door and send their status to the processing unit. In such situations we have to remove the exit door from our environment and reassign its zone to other ones that are still usable. The exit door will not be considered in forming the largest non-directed simple graph.

5.8.3 Determining the new charge for each exit door

After all agent's amount of charge is determined, the process will calculate the new amount of positive charges for each exit door based on the results obtained in the previous step. In order to determine the new value of each exit door charges, we need to consider all agents that are belonging to that exit door at the moment. The new amount of charge for each exit door is shown by the following equation 5:

$$Q'_{ED_i} = \left| Q_{Z_i} + Q_{ED_i} \right| \qquad (5)$$

Where Q'_{ED_i} is the new positive charge for i^{th} exit door, Q_{Z_i} is the initial positive amount of charge for the ith zone which belongs to the ith exit door and Q_{ED_i} is the previous positive amount of charge for the ith exit door. Determining charges of each exit door is based on the number of people in each zone as well as the status of each exit door. For example, assume having an exit door with 10 people in its zone and the adjacent exit door with the smaller width with only 3 people in its zone, leads us to expand the area of the exit door with the smaller number of people. For the next round of processing we might consider many of the people that belong to the bigger exit door for the smaller one.

5.8.4 Determining the new status for each exit door

Regardless of the already mentioned features, there are many other features that may exist in the environment that should be mentioned while determining each zone and boundary. Determining exit door status is necessary, especially in emergency situations. In the case of blocked exit doors for such reasons as smashed walls or people who block the exit door by pushing or shoving each other, the reliability of the exit door can be significantly decreased. In such cases the amount of positive charge of the exit door will reduce if its reliability decreases. We called the reliability factor for each exit door the safety rate of that exit door. At the initialization phase, the safety rate for each exit door that is ready to use is set to (+1). This rate will change based on the new environmental information gathered by sensors based on each exit door's status. The safety rate is shown by the following equation 6:

$$SR_{Ed_i} = \left| SR_{(ED_i)_{t-1}} \times SR_{(ED_i)_t} \right| \qquad (6)$$

Where $SR_{(ED_i)_t}$ is the safety rate for the ith exit door at the t moment.

5.9 Decision making and updating the boundaries

To make the decision and update the safe boundaries for each of the general convex zones, having the values described in the previous third phases is essential. Based on the new values for each β zone in each moment, the value of charges for each exit door and hence the safe boundaries of the general convex zones will change. The process of determining the boundaries for each exit door should continue and be updated by gathering new data from different sensors at each moment. Having reliable and real time hardware in order to detect and determine the different physical status of the exit doors, people, and locations is essential for forming the safe boundaries in a reasonable

time. The process refreshes the results all the time to redirect to the second phase after reaching and completing the third phase.

Having the safe boundaries, which is the result of the 4[th]phase, helps people to make better decisions. This produces lower risk and hence better results in terms of evacuating people out of danger in emergency situations.

6 IMPLEMENTATION AND EVALUATION

In this section we apply the optimized Coulomb's Law to a sample environment and compare the results as a step towards validating our model. We selected the Station nightclub environment. On Thursday, February 20, 2003, at The Station nightclub located at 211 Cowesett Avenue in West Warwick, Rhode Island, a fire accident occurred, which was the fourth deadliest nightclub fire in American history. More than 100 people lost their lives because of it. The tour manager of the evening's headlining band used pyrotechnics during the show that were the main cause of the fire. In the beginning, the fire ignited flammable sound insulation foam in the walls and then spread to ceilings surrounding the stage. Initially, there were about 132 people inside before the fire incident. Some of them were injured and about 32 escaped uninjured. The cameras and sensors that were installed inside the environment recorded growing billowing smoke and that one exit door was blocked. This made escape impossible because of limiting the vision people at the site. In our approach, we first divide the area into convex zones. We then form the bidirectional cycle crossing all exit doors, and, based on the centroid location of the formed shape, we form the α zones, β zones and θ angles. Based on some assumptions about the percentage of people who were spread in the environment and their physical specifications, we form the new zones. To apply our strategy, we consider only the map of empty buildings as a first step to form the zones. Then, regarding the crowd distribution, we form the new safe boundaries. A general view of the building shown in Figure 2.
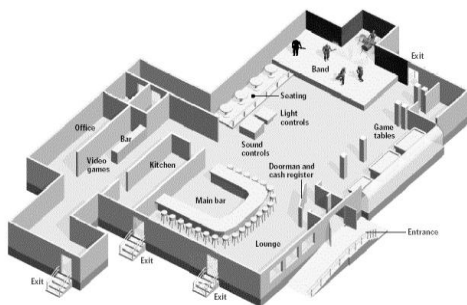
Figure 2.General view of the environment.

At this level, in order to determine the initial safe boundaries, we only focus on exit doors. In order to apply our strategy, at the first step we have to determine the exact locations of each exit door as well as the width of each.

This task will be done by using a raster technology that will send raster information to the processing unit by detectors and cameras that are installed in the environment. We also need to determine the general convex zones. Based on the environment map, we have generally two convex zones as shown by equation 7.

$$Z = \{Z_1, Z_2\} \qquad (7)$$

The first general convex zone consists of four exit doors and the second zone consists of a single exit door as shown by equations 8 and 9.

$$Z_1 = \{ED_1, ED_2, ED_3, ED_4\} \qquad (8)$$

$$Z_2 = \{ED_5\} \qquad (9)$$

Since we have more than one exit door in the first zone, there are α zones, β zones, and θ angles for each exit door, whereas the second zone only has a single α zone related to its exit door. There are no β zones or θ angles for the second zone because it consists of only a single exit door, and hence all area of the second zone belongs to its only exit door (ED_5).

We used the metric measurement herein and therefore of five exit doors available in the environment, the width of exit doors 1, 3, 4 and 5 are 1 meter and exit door 2 is 2 meters. Based on the width of each exit door, we are able to compute α zones and the charge of each one as they shown in Table 1.

Table 1. The α zones and the amount of charges for each exit doors based on their width.

Exit doors		
No	Width (m)	Charge
1	1	100
2	2	200
3	1	100
4	1	100

We consider each α zone related to each exit door as a sphere of charge which has a center equal the central width location of each exit door. Figure 3 shows the result of dividing the area into convex zones and α zone related to each exit door.
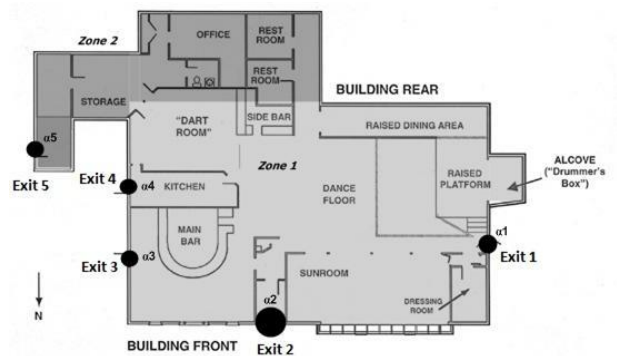
Figure 3: Convex zones of the area and α zones for each exit door.

Because the second zone doesn't have any β zones or θ angles, we only focus on the first zone. We have four exit doors in this zone; hence the largest simple non directed cycle graph has the length of 4. To form the mentioned graph, we need to connect the central points of each exit doors together through straight lines. This diagram must meet each exit door only once. Figure 4 shows the largest simple non-directed cycle graph of length 4 crossing all exit doors in the first convex zone.
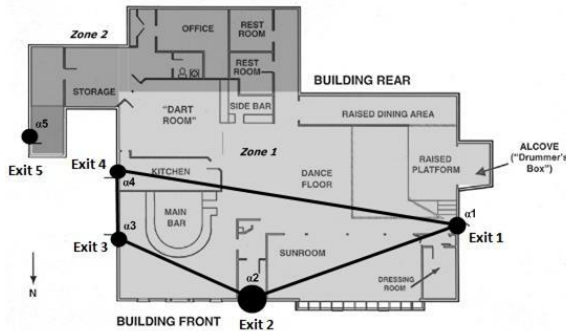
Figure 4. The largest simple non directed cycle graph of length 4 crossing exit doors in the first convex zone.

For the next step, we need to find the centroid point of the 2D shape formed by the mentioned graph. We also need to find the equilibrium points between each adjacent pairs of charges. To do this, we need to have the values of the adjacent pairs of charges. Based on our strategy, we assumed all exit doors have positive charges and are fixed in their places. To find the equilibrium position, we use a positive charge that is equal to the average of the adjacent pairs of charges. The mentioned positive charge is placed on the straight line between the pairs of charges and is closer to the smaller charge. In case of having a same amount of charges, the positive charge will locate in the middle of pairs of charges. To form β zones and θ angles, we have to connect from the centroid point to each equilibrium point and continue the line to the environment. The following Figure 5 shows the centroid location of the 2D shape for the mentioned graph, the equilibrium locations, the β zones and θ angles related to the exit doors of the first zone:
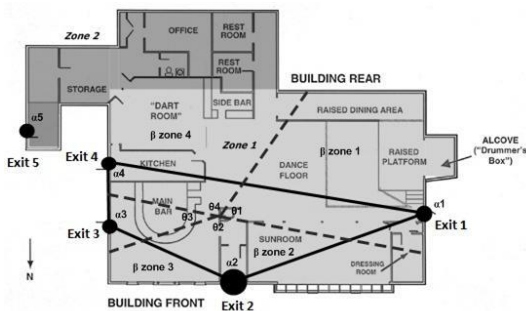
Figure 5. The centroid, Equilibrium points, β zones and θ angles for the first zone.

Figure 5 shows all areas needed for the first zone when there is not any individual available in the environment. At this step we will calculate the charges of the agents available in the environment and will update the safe boundaries based on their distribution in the environment. The process of gathering information about the physical specifications of the agents and their locations is done by sensors and detectors that are installed in the environment. This data will then be processed by our method. Of 230 people that we assumed are available in the environment, we consider 200 people are located in the first zone and 30 people are placed in the second zone. We also assumed that in each area, half of the people are male and the other half are female. We considered all people in our environment to have normal health statuses. We consider in each zone, the ages range is between 20 and 40 years old. We assumed that all exit doors are open all the time and safe to use with their full evacuation capacity, which means blocking will not happen in the environment during the experiment. We apply our strategy in two different modes. When the distribution is the same and when it's not. The following equations 10 and 11, show the collections of the agents in each zones:

$$Z_1 = \{Ag_1, Ag_2, Ag_3, \dots, Ag_{200}\} \qquad (10)$$

$$Z_2 = \{Ag_1, Ag_2, Ag_3, \dots, Ag_{30}\} \qquad (11)$$

We assumed to have a same crowd distribution in our experiment. We also assumed there are 100 males and 100 females available in the first zone. All of them have normal health status and between 20 and 40 years old. The gender percentage for each β zone is the same as 50 percent. Hence for each β zone in the general convex zone, we have 25 people consists of 50 percent male (13 people) and 50 percent female (12 people). The table 2 shows the β zones, initial α zone charges, and new α zone charges.

Table 2. β zones, initial α zones charges and new α zones charges.

β Zone	Init α zone	New αzone
1	100	75
2	200	175
3	100	75
4	100	75

Based on our assumption, we observed to have slight changes for the safe boundaries after applying the charges of agents in each β zone. There are four β zones, and thus each zone is assumed to support a quarter of the people in it. We also considered having 13 males and 12 females in each β zone. Figure 6 shows the new safe boundaries based on the crowd distribution we assumed in our experiment.
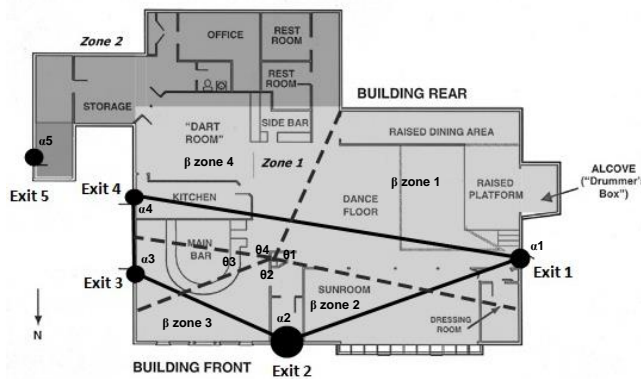
Figure 6. New safe boundaries based on the normal crowd distribution.

The illustrated crowd distribution can be used in many places such as theater saloons or conferences rooms. In such places, because of the kind of usage of environment, the distribution of the crowd is equal for all areas inside.

7   CONCLUSIONS

This paper explored a powerful mechanism for guiding the crowd out of danger using Coulomb's Law, as well as graph theory, and convex and centroid concepts in order to form safe and reliable boundaries around each exit door in the environment in order to provide a decision aid for supervisory control personnel. This yields strategies for people who are trapped in an indoor public space at a dangerous location to be most rapidly evacuated. Using this mechanism can decrease errors committed by exiting individuals in the shortest time period, especially when the circumstances in the environment are obscured due to calamities that change layouts, such as fallen walls. We have fully implemented our algorithm and demonstrated it on a real world scenario.

8   REFERENCES

[1] Aik, L.E., Exit selection behaviors during a classroom evacuation. In International Journal of the Physical Sciences Vol. 6(13). pp. 3218–3231, 2011.

[2] Binti Sharbini, H., Bade, A, Analysis of Crowd Behavior Theories in Panic Situation.In International Conference on Information and Multimedia Technology, IEEE, 2009.

[3] Braun, A.,Musse, S., Oliveira and Bardo, E., Modeling Individual behaviors in Crowd simulation. In Proceedings of the 16th International Conference on Computer Animation and Social Agents. IEE Computer Society, 2003.

[4] Cole, J.,Zhuang, J., Decisions in Disaster recovery operations: A game theoretic perspective on organization perspective. Berkeley Electronic Press, 2011.

[5] Dogbe, C., On the modeling of crowd dynamics by generalized kinetic models.In journal of Mathematical Analysis and Applications. Elsevier Inc, 2011.

[6] Guy, S.J., Kim, S., Lin, M.C., Manocha, D., Simulating Heterogeneous Crowd Behaviors using Personality trait Theory.Symposium on Computer animation.Eurographics Association, 2011.

[7] Johansson, A.,Helbing, D.,Habib Z.,Abideen, A., AlBosta, S., From Crowd Dynamics to Crowd Safety: A Video - Based Analysis Advances in Complex systems.World scientific publishing company, 2008.

[8] Korhonen, T.,Hostika, S., Modeling Social Interactions in Fire Evacuation. In proceedings 7th International Conference on Performance-Based codes and fire safety design methods, 2008.

[9] Pan J., Zhang, L., Lin, M.,Manocha, D., A Hybrid Approach for Simulating Human Motion in Constrained Environments.In Computer Animation and Virtual Worlds.vol. 21. pp. 137-149, 2010.

[10] Patil, S., Van den Berg, J., Curtis, S., Lin, M.C., Manocha, D., Directing Crowd Simulations Using Navigation Fields.In IEEE Transactions on Visualization and Computer Graphics (TVCG).vol. 17. pp. 244-254, 2010.

[11] Peacock, R.D., Averill, J.D.,Kuligowski, E.D., Stairwell Evacuation From Buildings: What We Know We Don't Know.NIST TN 1624; NIST Technical Note 1624, 2009.

[12] Peacock, R.D., Averill, J.D.,Kuligowski, E.D., Egress from the World Trade Center Towers on September 11, 2001. In Fire Technology, Springer, 2011.

[13] Perez-Delgado, C.A., Cheung, D., Models of Quantum Cellular Automata.Quantum Physics, 2009.

[14] Proulx, G., Building Egress Using Photoluminescent Markings.Construction Technology Update No. 78.National Research Council of Canada, 2011.

[15] Singh, V.K., Gupta, A.K., Agent Based Models of Social Systems and Collective Intelligence, International Conference of Intelligent Agent & Multi-Agent Systems, IAMA, 2009.

[16] Xiaoping, Z.,Tingkuan, Z.,Mengting, L., Modeling crowd evacuation of a building based on seven methodological approaches.Building and Environment n044.pp 437-445, Elsevier Ltd, 2009.

[17] Yeh, H., Curtis, S., Patil, S., Van den Berg, J.P., Manocha, D., Lin, M.C., Composite Agents.Symposium on Computer Animation.Eurographics Association, 2008.

[18] Zhao, Y., Billings, S.A., Neighborhood detection using mutual information for the identification of cellular automata.In Transactions of Systems, Man, and Cybernetics.Part B, IEEE, 2006.

Head movement artifact removal in EEG signals using Empirical Mode Decomposition and Pearson Correlation

Gerardo Rosas-Cholula[1], Juan Manuel Ramirez-Cortes[1], Jose Rangel-Magdaleno[1], Pilar Gomez-Gil[2], Vicente Alarcon-Aquino[3]

[1]National Institute of Astrophysics, Optics and Electronics. Department of Electronics
Calle Luis Enrique Erro No. 1, Tonantzintla, Puebla, 72760, Mexico
[2]National Institute of Astrophysics, Optics and Electronics. Computer Science Department
Calle Luis Enrique Erro No. 1, Tonantzintla, Puebla, 72760, Mexico
[3]University of the Americas, Puebla. Department of Electronics and Computer Science
Exhda. Sta. Catarina Martir, Cholula, Puebla, 72720, Mexico

Abstract

This paper presents an approach on head movement artifact removal from EEG signals, in the context of an ongoing brain computer interface project. The proposed artifact removal scheme is based on Empirical Mode Decomposition (EMD) applied to the signals obtained from each electrode. Correlation analysis using Pearson coefficient allows identification of those intrinsic mode functions related to common artifacts, which are associated to head movement. The goal of this experiment is separation of signals corresponding to single and double blinking from head movement artifacts. Once the preprocessing is applied, blinking detection is reduced to threshold operations. Final selection step based on Mahalanobis distance provides a detection rate of 95% in average.

Keywords: EEG, artifact, empirical mode decomposition, correlation.

1. Introduction

Artifacts caused by movement of the head during the acquisition of EEG signals constitute a significant limitation in a number of applications, such as data clinical interpretations, automated systems for analysis or detection of pathologies, human-computer interfaces (HCI), and brain computer interface (BCI) applications. A BCI system aims to translate the electrical brain signals generated by a human being as a result of some thoughts, in commands able to perform some control actions in computerized mechanisms [1]. Affordable EEG devices recently released makes attractive the development of such systems, however, technical characteristics of those devices, such as poor signal to noise ratio, and some artifacts such as

electrode displacements, or subject movements require incorporation of additional signal processing techniques [2]. An important motivation to develop BCI systems is to allow an individual with severe motor disabilities to have control over specialized devices such as assistive appliances, neural prostheses, speech synthesizers, or a personal computer directly. Among a number of currently available technologies, EEG systems constitute a good alternative to develop BCI applications given some characteristics such as non-invasiveness, affordability, transportability, and size [3, 4]. However, EEG signals detected through surface electrodes present some problems, such as weak EEG electrical signals, low spatial resolution, artifact contamination, poor electrical contacts, and contact noise arising from head movements, manifested in EEG signals as additional artifacts. In practical situations of BCI systems, the user is expected to interact with automated equipment in a natural way, and head movements are part of that interaction. Additionally, because of the fluid, bone, and skin that separate the electrodes from the actual brain activity, the already small signals are scattered and attenuated before reaching the electrodes, with the consequence that head movement artifacts can reach amplitudes in the same order of magnitude of EEG signals related to brain activity, or even more. There have been several approaches oriented to do artifact removal in EEG signals with techniques such as independent component analysis (ICA) [6, 7, 8], high-order statistics and ICA [9], wavelet transform [10], adaptive least mean square [11], canonical correlation analysis [12], and others. Most reported papers consider muscle activity and blinking as EEG artifacts, although some works concentrate in removing only artifacts associated to EMG signals. The work presented in this paper is part of a project about the use of the EEG Emotiv® headset [13] as cursor control emulating the typical operations of a computer-mouse.

Computer mouse emulation using hands-free alternatives is a project which has been broadly pursued in the last decades. The following is a partial list of some successful modalities which have been reported: visual tracking [14], voice control [15], electromyographic signals [16], electro-oculographic potentials [17], and electroencephalographic signals [18, 19]. In this application, the Emotiv headset is used to indirectly detect EMG blinking signals reflected in the EEG activity captured through the headset. Blinking is used as a mean to perform the clicks required in a computer mouse. Although Emotiv EPOC headset represents an efficient alternative that is practical and economical, the EEG detected signals are noisy and contaminated with artifacts arising from different sources. Head movements associated to the expected use of the device as a mouse pointer will produce noise on the signal acquired by the electrode due to slight movements of the electrodes over the scalp. Cursor position is controlled using information from a gyroscope included in the headset, and clicks are generated through the user's blinking with a detection procedure based on Empirical Mode Decomposition (EMD), which is used to separate desired blinking signals from head movement artifacts. The rest of the paper is organized as follows: Section 2 describes principles associated to EMD. Section 3 presents the proposed scheme and a description of the included modules. Section 4 describes the experimental setup and the experiments carried out. Section 5 presents and discusses the obtained results, and section 6 presents some concluding remarks and future work of this research.

2. Empirical Mode Decomposition (EMD)

EMD was first introduced by Huang [20] for spectral analysis of non-linear and non-stationary time series, as the first step of a two stage process, currently known as the Hilbert Huang Transform (HHT). EMD is used in this work with two objectives: signal preprocessing to reduce noise arising from head movement, and double blinking detection to simulate the "click" operation of a traditional mouse device. Essentially, EMD aims to empirically identify the intrinsic oscillatory modes or intrinsic mode functions (IMF) of a signal by its characteristic time scales, in adaptive way. These modes represent the data by means of local zero mean oscillating waves obtained by a sifting process. Thus, an IMF satisfies two main conditions: taking account the complete data set, the number of extrema points (min and max) must be equal or differ at most by one to the number of zero crossing points; the mean value of the envelopes is always zero which are defined by the local maxima and local minima. EMD can be summarized as follows [21]: Given a signal $x(t)$ identify its extrema (both minima and maxima). Generate the envelope by connecting maxima and minima points with a curve, for instance, cubic spline interpolation, although other interpolation techniques are allowed. Determine the mean by averaging and extract the detail, as expressed in eq. 1 and 2. Finally iterate on the residual $m(t)$.

$$m(t) = \frac{e_{min}(t) + e_{max}(t)}{2} \qquad (1)$$
$$d(t) = x(t) - m(t) \qquad (2)$$

There are some iteration stopping criterions such as establishing a certain number of siftings, a predefined threshold, or specifying a minimum amplitude of residual. EMD satisfies completeness and orthogonality properties in the same way as spectral decompositions, such as Fourier or wavelet transform. The completeness property is satisfied by EMD, meaning that it is possible to reconstruct the original signal based on their decompositions. These decomposition functions should all be locally orthogonal to each other, as expressed in equation 3, although some leakage may arise.

$$\overline{(x(t) - \overline{x(t)}) \cdot \overline{x(t)}} = 0 \qquad (3)$$

An orthogonality index expressed in equation 4 is used to keep track of leakage magnitude within some limits. *X* is the original signal with $i \neq j$.

$$I_O = \sum_{t=0}^{T} \left(\frac{\sum_{j=1}^{n+1} \sum_{k=1}^{n+1} IMF_j(t) IMF_k(t)}{X^2(t)} \right) \qquad (4)$$

Occasionally it is necessary to consider a local EMD. In this case, sifting operations are not applied to the full length signal. Sometimes there exist zones resulting in over-iteration to achieve a better local approximation contaminating other parts of signal and in consequence over-decomposing. Thus, the local zones where the error remains large have to be isolated, and the algorithm should maintain iterating only over these zones. Local EMD is implemented introducing a weighting function; this function must describe a soft decay outside the problem zone. In consequence equation 2 can be written as:

$$d(t) = x(t) - w(t)m(t) \qquad (5)$$

Figure 1 shows typical results obtained from an EEG signal using EMD with five decomposition iterations.

3. Proposed scheme and module description

A BCI based on EEG analyzes ongoing electric brain activity for brain patterns that originate from specific brain areas. To get consistent recordings from specific regions of the head, scientists rely on a standard system for accurately

placing electrodes, which is called the International 10–20 System , generally used in clinical EEG recording and EEG research.

Fig. 1. EEG signal decomposition with the first five IMFs.

Figure 2 shows the electrode positions and denominations used in the international 10-20 system. The squares indicate the available electrodes on Emotiv system. The EEG signals required to perform the detection are obtained from electrodes AF3/AF4, which are labeled according to the mentioned 10-20 international system.

Fig. 2. International system 10-20.

The modules proposed to simultaneously separate head movement artifacts and blinking from the EEG signals are shown in the block diagram of figure 3. The artifact could be detected considering that noise present in all electrodes over the scalp will show high correlation. Thus, the proposed system is based on finding common signals in the electrodes, empirical mode decomposition, Pearson correlation, and an integration module which includes computation of Mahalanobis distance and further thresholding. Figure 4 shows an example of double blinking events immerse in noise produced by head movement.

Fig. 3. Proposed scheme; head movement artifact and blinking detection

As previously stated, noise produced by head or body movement will appear in all electrodes of the system with small variations, therefore, correlation analysis using Pearson coefficient is used for noise detection purposes.

Fig. 4. Head movement noise during double blinking events.

Pearson correlation coefficient provides a measure of dependence between two random variables [22]. Equation 6 defines the Pearson correlation with expected values μ_X and μ_Y and standard deviations σ_X and σ_Y.

$$\rho_{X,Y} = \frac{E[(X - \mu_X)(Y - \mu_Y]}{\sigma_X \sigma_Y} \tag{6}$$

Correlation function applied directly to the signals obtained from each electrode will state dependence between channels. Common signals detected would have to be removed; however, applying directly an operation to separate those signals could cause removing also important information. Therefore, decomposing the signal from each electrode will reduce the loss of information, allowing the system to distinguish between artifacts from head movements and double blinking signals. That decomposition has been carried out using EMD technique.

Figure 5 shows an example of EMD decomposition, with a plot of IMF 1 to IMF 5 obtained from four different electrodes near AF6. Visual inspection indicates similarities in IMFs 1, 3, 4 and 5. In this part of the experiment, EMD decomposition typically yielded between 14 and 16 MFIs.

Fig. 5. EMD decomposition applied on signals obtained from four electrodes: a) FC5, b) FC6, c) P8, and d) P7.

In order to find the amount of similarity or dependence, the Pearson correlation is calculated. Additionally, a p-value is computed by transforming the correlation to create a t statistic with n-2 degrees of freedom, where n is the number of rows in the correlation matrix. The confidence bounds are based on an asymptotic normal distribution of $0.5 \log((1+R)/(1-R))$, where R is the correlation coefficient with an approximate variance of $1/(n-3)$.
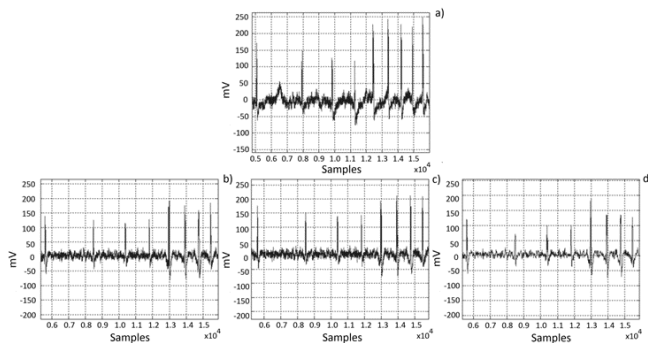
Fig. 6. Noise reduction based on correlation function removing a) 1 IMF, b) 2 IMFs, c) 3 IMFs and d) 4 IMFs.

This algorithm is repeated for all IMFs, taking as reference the electrode AF6. A slide window of 10 seconds is applied during correlation calculation. Figure 7 shows the noise reduction using the correlation coefficients associated to the Thus, p values less than 0.05 were considered to imply high correlation. Figure 17 shows an example in which IMF3 from electrode AF6 is compared to the rest, from a total

number of 12 electrodes, resulting in p-values close to 0, except for one electrode. corresponding IMF. If there is a correlation in most of the electrodes, the corresponding IMF is prevented from passing to the integration module. Once the noise is reduced, a second derivative is obtained in order to determine whether a critical point is a local maximum or a local minimum. A typical double blinking event will have two local max points inside a 0.5 seconds window. Figure 8 shows the signal after this processing, thus the classifier is reduced to a simple threshold function.
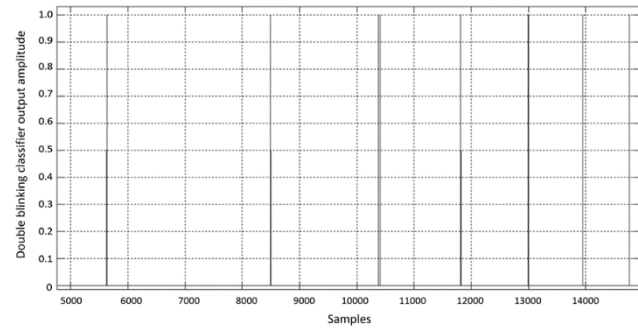
Fig. 7. Double blinking detection with noise reduction.

4.   Experimental setup

Subjects under testing were seated in a comfortable position using the Emotiv headset with a laser pointer attached at the top, as shown in figure 8. A simple application developed in visual basic, showed a red circle moving through the screen following horizontal and vertical displacements, with a linearly increasing speed. The subject was instructed to follow as closely as possible the red circle with the pointer. EEG and gyroscope data are recorded simultaneously. Additionally, the subject was told to do a double blinking when a black circle appears in the screen. In that instant, the application sends a marker to the recording system. Thus, the test considers the worst case scenario in which the user is moving the head and doing a double blinking simultaneously. This case would rather occur in a practical situation, because the user usually stops the movement before doing a click with the mouse. Testing setup system is depicted in figure 15. Head movements during this action generates artifacts which have to be removed in order to distinguish user´s blinking.
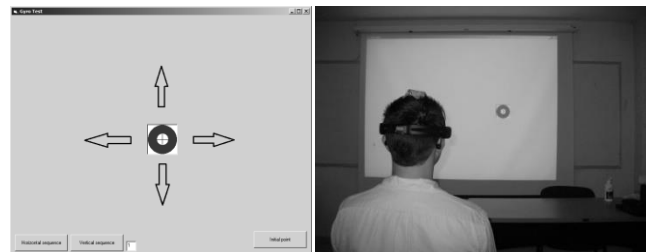
Fig. 8. Experimental setup of EEG-based mouse emulation.
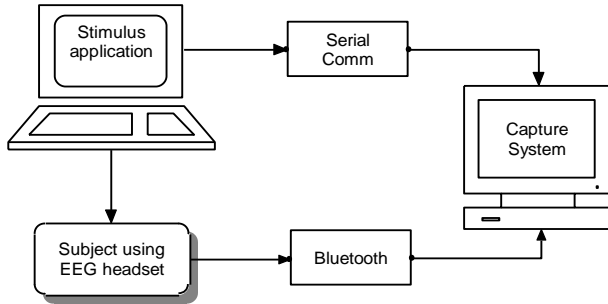
Fig. 9. Testing setup system.

5. Results

Typical blinking was found to be formed from 1 to 5 IMF and residual. Features are obtained from the energy of each IMF and residual. The obtained feature vectors generated through the IMFs are fed into a Mahalanobis-distance based classifier [23]. System test was performed using fold validation, dividing the data set in training and testing groups of 10 vectors each group, and finally the system is tested using 5 complete sequence of 91 seconds in which 10 double blinking events occur randomly. A double blinking event is experimentally found to fall inside a Mahalanobis distance value between -1.34 and -3.25.

Table I. Average detection rate and performance metrics.

TP	FN	FP	TN	FP Rate	TP Rate	Precision	Accuracy
5	5	1	75	0.0132	0.5	0.8333	0.9302
8	2	0	76	0	0.8	1	0.9767
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
7	3	1	75	0.0132	0.7	0.8751	0.9534
9	1	1	75	0.0132	0.9	0.9	0.9767
8	2	3	73	0.0395	0.8	0.7272	0.9418
8	2	3	73	0.0395	0.8	0.7272	0.9418
5	5	0	76	0	0.5	1	0.9418
8	2	1	75	0.0132	0.8	0.8888	0.9651
9	1	3	73	0.0395	0.9	0.75	0.9534
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
10	0	4	72	0.0526	1	0.7142	0.9534
6	4	0	76	0	0.6	1	0.9534
9	1	2	74	0.0263	0.9	0.8181	0.9651
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
3	7	0	76	0	0.3	1	0.9186
5	5	0	76	0	0.5	1	0.9418
5	5	0	76	0	0.5	1	0.9418
10	0	2	74	0.0263	1	0.8333	0.9767
9	1	1	75	0.0132	0.9	0.9	0.9767

The system performance is analyzed through a Receiver Operation Characteristic plot (ROC) [24], which indicates the False Positive Rate (FPR) versus the True Positive Rate (TPR). Figure 10 shows the ROC curve obtained in average. The curve indicates a common tendency of an increasing rate of true positives events with simultaneous increasing rate of false positives. A small increase of the FP rate compared to the variation of TP rate can be noticed. Table I summarizes the detection rate obtained using the proposed scheme, indicating an accuracy of 95% in average.
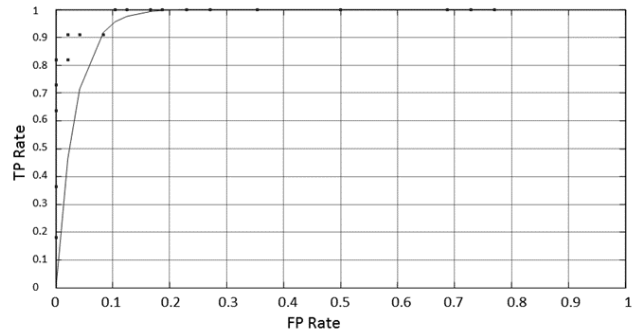
Fig. 10. Average ROC curve.

6. Conclusions

We have described an original approach on head movement artifact removal from EEG signals, in the context of an ongoing brain computer interface project. The proposed artifact removal scheme is based on Empirical Mode Decomposition (EMD) applied to the signals obtained from each electrode. Correlation analysis using Pearson coefficient allowed identification of those intrinsic mode functions related to common artifacts, which are associated to head movement. System performance showed very good results on separation of blinking from head movement artifacts. The proposed signal processing system was applied to emulation of clicks in a computer mouse. Final selection step based on Mahalanobis distance provided a detection rate of 95% in average. Analysis on detection rate indicated that EMD provided an efficient, effective and quick computational tool, adequate to non-stationary signals. The proposed noise reduction method based on information available from multiple electrodes is a preprocessing technique which can be adapted to different EEG systems, when noise caused by head or body movement is required to be removed. Additional experiments exploring the incorporation of classifiers such as Support Vector Machine and Neural Networks are currently in progress.

7. Acknowledgments

References

[1]   P. Brunner, L. Bianchi, C. Guger, F. Cincotti, and G. Schalk, "Current trends in hardware and software for brain–computer interfaces", Journal of Neural Engineering, Vol. 8, No. 2, 2011.

[2]   K. Stamps, Y. Hamam, "Towards inexpensive bci control for wheelchair navigation in the enabled environment – a hardware survey", Brain Informatics Lecture Notes in Computer Science , pp. 336-345, Vol. 6334, 2010.

[3]   C. G. Pinheiro,   E. L.M. Naves,   P. Pino,   E. Losson, A. O. Andrade, and G. Bourhis, "Alternative communication systems for people with severe motor disabilities: a survey", BioMedical Engineering OnLine, Vol. 10, No. 31, pp. 1-28, 2011.

[4]   C. T. Lin, L.W. Ko, M.H. Chang, J. R. Duann, J. Y. Chen, T. P. Su, T.P. Jung, "Review of wireless and wearable electroencephalogram systems and brain computer interfaces, a mini-review, Gerontology, Vol. 56, No. 1, pp. 112-119, 2010.

[5]   M. Fatourechi, A. Bashashati, K.W. Rabab, and G. E. Birch. "EMG and EOG artifacts in brain computer interface systems: A survey." Clinical neurophysiology, Vol. 118, No. 3, pp. 480-494, 2007.

[6]   M. Crespo-Garcia, M. Atienza, and J. L. Cantero. "Muscle artifact removal from human sleep EEG by using independent component analysis." Annals of biomedical engineering, Vol. 36, No. 3, pp. 467-475, 2008.

[7]   A. Palumbo, B. Calabrese, G. Cocorullo, M. Lanuzza, P. Veltri, P.Vizza, A. Gambardella, M. Sturniolo, "A Novel ICA-Based Hardware System for Reconfigurable and Portable BCI", In Proc. of International Workshop on Medical Measurements and Applications, Cetraro, Italy, May 29-30, 2009.

[8]   A. Kachenoura, L. Albera, L. Senhadji, and P. Comon, "ICA: A Potential Tool for BCI Systems", IEEE Signal Processing Magazine, pp. 57-68, January, 2008.

[9]   A. Delorme, S. Makeig, and T. Sejnowski. "Automatic artifact rejection for EEG data using high-order statistics and independent component analysis." In Proc. of the 3rd International Workshop on ICA, pp. 458-462, 2001.

[10]   V.   Krishnaveni,   S.   Jayaraman,   S.   Aravind,   V. Hariharasudhan,   and   K.   Ramadoss,   "Automatic identification and Removal of ocular artifacts from EEG using Wavelet transform", Measurement Science, Review 6, No. 4, pp. 45-57, 2006.

[11]   S. Mehrkanoon, M. Moghavvemi, and H. Fariborzi. "Real time ocular and facial muscle artifacts removal from EEG signals using LMS adaptive algorithm", In Proc. of IEEE International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia, pp. 1245-1250, 2007.

[12]   W. D. Clercq, A. Vergult, B. Vanrumste, W. V. Paesschen, and S. V. Huffel, "Canonical correlation analysis applied to remove muscle artifacts from the electroencephalogram", IEEE Transactions on Biomedical Engineering, Vol. 53, No. 12, pp. 2583-2587, 2006.

[13]   Emotiv - Brain Computer Interface Technology. Available online: http://www.emotiv.com

[14]   M. Betke, J. Gips, P. Fleming, "The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 10, No.1, pp. 1-10, March 2002.

[15]   S. Harada, J. Wobbrock, J. Malkin, J. Bilmes, J.A. Landay, "Longitudinal study of people learning to use continuous voice-based cursor control", In Proc. of 27th International Conference on Human factors in Computing Systems, Boston, Massachusetts, April 4-9, 2009.

[16]   T. Felzer and R. Nordmann, "Evaluating the Hands-Free Mouse Control System: An Initial Case Study", In Proc. of International Conference on Computers Helping People with Special Needs, LNCS 5105, pp. 1188–1195, Linz, Austria, 2008.

[17]   J. R. de la O. Chavez, C. Aviles-Cruz, "Pattern Recognition Applied to Mouse Pointer Controlled by Ocular Movements", WSEAS Transactions on Systems, Vol. 4, No. 8, pp. 511-520, April 2009.

[18]   J. Knezik and M. Drahansky, "Simple EEG Driven Mouse Cursor Movement", Computer Recognition Systems 2, Vol. 45, pp. 526–531, Springer-Verlag, 2007.

[19]   D. J. McFarland, D. J. Krusienski,W. A. Sarnacki, J. R. Wolpaw, "Emulation of computer mouse control with a noninvasive brain–computer interface", Journal of Neural Engineering, Vol. 5, No. 2, pp. 101–110, 2008.

[20]   N. E. Huang, Z. Shen, S. R. Long, M. L. Wu, H.H. Shih, Q. Zheng, N. C. Yen, C. C. Tung and H. H. Liu, "The empirical mode decomposition and Hilbert spectrum for nonlinear and nonstationary time series analysis," Proc. Roy. Soc. London Series A, Mathematical, Physical, and Engineering Sciences, Vol. 454, pp. 903–995, 1998.

[21]   G. Rilling, P. Flandrin, P. Goncalves, "On empirical mode decomposition and its algorithms", In Proc. of IEEE EURASIP Workshop on Nonlinear Signal and Image Processing, Vol. 3, pp. 8-11, 2003.

[22]   J. Benesty, J. Chen, and Y. Huang, "On the importance of the Pearson correlation coefficient in noise reduction", IEEE Transactions on Audio, Speech, and Language Processing, Vol. 16, No. 4, pp. 757-765, 2008.

[23]   R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, Second Ed., John Wiley and Sons, Inc., 2001.

[24]   T. Fawcett, "ROC graphs: notes and practical considerations for researchers", Machine Learning, Vol. 31, pp. 1-38, 2004.

A Personal Smart Assistant for Open Environments

Ali Hussain[1], Afshan Samani[1] and Hamada H. Ghenniwa[1]

[1] Electrical and Computer Engineering Department, Western University, London, Ontario, Canada

Abstract - *Personal Smart Assistants (PSAs) provide assistances to users to enable them achieve their goals efficiently. In order to recommend a relevant assistance, the personal smart assistant builds a user model which represents the user's information such as interests, behaviours, goals, etc. Modeling the user's behaviours is essential to provide a relevant assistance at the right time. In open environments, dynamic changes occur that alters the user's behaviours as he/she is trying to achieve the goal. Furthermore, when the personal smart assistant interacts, on behalf of the user, with other entities in open environments, then the user's privacy become a concern. In this paper, we are proposing a personal smart assistant's architecture that is able to model the user's behaviours in open environments through observing his/her deliberative and reactive actions and dynamically alter the user's plan to achieve the goal based on the changes that occur in the environment. Also, we are proposing a privacy protection approach, within the PSA architecture, to preserve the user's privacy as the personal assistant interact with other entities in open environments.*

Keywords: Personal Smart Assistant Agent; Open Environment; User Modeling; Privacy Protection.

1 Introduction

The explosion of the online, mobile, and social networks transformed computation to become a platform that redefines many aspects of our lives and businesses. As a result, many of our goals are now technology driven and for us to be able to achieve these goals we might be required to deal with several applications and services. Usually people lack the complete knowledge of working with these different applications and hence they require a form of assistance. However, for this assistance to be really beneficial, it should be personalized towards the user's needs and goals.

Smart software agents have been recognized as a promising solution for the development of user-centric applications. Personal assistants are smart software agents that provide users with services that are adaptive to the user and the environment [1]. A cognitive user model is an essential component in personal assistants. Our focus in this paper is on proposing a user behaviour model that captures the user's behaviours in an open environment. Two major approaches are followed to model the user's behaviours. The first approach to model the user's behaviours is through capturing the user's responsive actions to dynamic changes that occur in the environment. The second approach of modeling the user's behaviours is through modeling the user's deliberative actions as he/she is following a certain

plan to achieve the goal. The challenge of modeling the user's behaviours in open environments is to dynamically alter the user's plan to achieve his/her goal, based on the dynamic changes that occur in the environment. In this paper we propose a combination of these two approaches to overcome the challenge of modeling the user's behaviours in open environments.

Another challenge that is addressed in this work is the user's privacy when interacting with other entities in these open environments through their Personal Smart Assistants (PSAs). Users' interests are sensitive information that disclosing it may result in violating their privacy [2],[3],[4]. For instance, promoting movies, books, software, web sites and other types of products that are about particular heritage, religion and group of people with specific political or social opinion demonstrates users' interests in these topics which may violate their privacy. Because privacy is a subjective concept and varies among users, attending privacy in open environments is challenging. Identity and interest information of users have the potential to be exploited. There are approaches to decouple the interest and identity of the user for targeted assistance. However, because it is possible to identify users by processing their interests, solutions that address the potential of privacy violation in disclosing information are essential. One of the approaches is to take the risk of privacy violation by exposing information to assistant providers [4]. In this work, we discuss how this concept can be integrated in PSA to provide a relevant assistance that does not violate users' privacy.

This paper presents a Personal Smart Assistant (PSA) agent that builds a cognitive user model and matches it with available assistances in open environments. Our proposed PSA models the user's deliberate, goal-driven, behaviours as well as his/her reactive, event-driven, behaviours. This enables us to capture the user's actions that he/she performs to achieve their goals as well as the actions that they perform to respond to dynamic changes in the environment. Furthermore, the proposed PSA protect the user's privacy as it is acting on his/her behalf in open environments.

The paper is organized as follows; a brief review of concepts and models related to personal smart assistants, user modeling, and privacy in open environments as well as the challenges in existing approaches is given in Section 2. Section 3 describes the architecture of the proposed personal smart assistant (PSA) that assists users in open environments while protecting their privacy. Section 4 outlines the major components and algorithms of the proposed PSA. Finally, conclusions and future work is given in section 5.

2   Background and related work

This section reviews concepts, models and approaches related to personal assistants, user modeling and privacy.

2.1   Personal assistant

The core issue behind the need of Smart Assistance (SA), regardless of the domain in which it is applied for, is the user's lack of knowledge. The user does not have the complete knowledge that would help him/her to achieve his/her goals, and therefore, assistance is needed to update the user's knowledge to achieve these goals.

The Personal Smart Assistant (PSA) is a software system that helps users to achieve their goals efficiently. The PSA can work on behalf of the user to achieve a task delegated by the user directly. Furthermore, it can provide a proactive assistance based on the current context of the user and the environment. The personal smart assistant models the environment in which the user is working, and it provides the functionalities of the different applications in this environment to the user based on the user's preferences and needs that are modeled in the user model.

2.2   User modeling in open environments

A user model is a representation of information about a specific user that is vital for personalized, user-centric, applications [5].   In order to provide the needed assistance for the user, and help him/her to achieve his/her goals; the personal smart assistant should establish a user model that captures the user's goals, interests and behaviours [6]. The main focus of this work is on modeling and capturing the user's behaviours in open environments. An open environment is an environment at which entities can join and leave at any time [7]; here entities are assumed to refer to humans, software applications or features, resources, etc. These entities might cause dynamic changes to the environment that influence the user's behaviours within this environment.

Since the behaviour of the user is dynamic, uncertainty is the nature of modeling the user's behaviours. Therefore, for a personal assistant to provide its proactive assistance it should be able to reason under uncertainty. The probability theory is one of the tools that are used to reason under uncertainty [8]. Once the user is modeled and his/her current actions are observed, the personal assistant could predict the next actions and therefore, provide a proactive help in performing these predicted actions.

2.3   State-based and event-based behaviour modeling

State-based modeling and event-based modeling are both used for behaviour modeling [9]. In state-based modeling, the system's states are emphasized and explicitly enumerated, and the behaviour is modeled in terms of the changes of states. The user's behaviour under state-based model is deliberate, which means that the user is behaving under a specific plan to perform actions that can move the system from one state to another state till the goal is reached.

The state-based approach has been adapted by many in the research field of user modeling. Stanford Research Institute Problem Solver (STRIPS) is a state-based problem solver proposed to specify the different sequences of actions by generating a number of states and their dependencies [10]. Ref. [11] proposes an activity-model for users during a meeting. This activity model utilizes STRIPS to describe all combinations of the users' activities and the different sequences of actions that could achieve the goal [9]. This state-based partial order planning is then transferred into probabilistic graphical models to infer how probable a specific execution sequence is [11]. In [12], another state-based user behaviour modeling is proposed using partial order planning. The user model proposed in [12] is used to build a personal intelligent assistance that is specifically designed to help users in Collaborative Design Environment (CDE). Beside the user behaviours, the proposed user model also represents the user's goals and his/her interests [12].

Although these state-based approaches, discussed in the previous paragraph, model and capture the user's uncertain behaviours through partial planning and probabilistic modeling, however, they are not adequate for open environments. First, the state-based model is a goal-driven approach, and hence, it will model only the actions that the user perform to achieve the goal. In open environments however, users might perform actions as responses to dynamic changes regardless if these actions achieve the goal or not. The previous approaches are not able to model these responsive actions and include them in the plan of possible actions, and hence, cannot provide the required assistance to perform such actions. Second, the partial order plan that was generated through the state-based approaches are applicable to achieve the goal if the environment does not change, and if the user is the only player in the environment. However, in open environments entities can cause dynamic changes in the environment which might cause the original plan, generated by the state-based models, to be unattainable anymore.

In contrast with the state-based modeling, the focus of the event-based model is on a set of possible events that occur in the system, and the behaviour is modeled through the responses to these events [13], [14]. Event-based behaviour models are reactive, which indicates that the user's behaviour is a response to the events that occur in the environment. In other words, the events that occur in the environment derive the user's actions. In [15] an elderly health monitoring system is proposed based on the event-based modeling. The elder's actions trigger events in the environment which are realized by the caregiver and utilized to reason about the elderly's activities and hence can make better assessment about the elderly's health [15]. Event-based modeling approaches work in open environments because they model the dynamic

changes in these environments as well as the responses to these dynamic changes. However, event-based modeling is not used to achieve a user's goal, and it does not provide a plan of possible paths of actions to achieve the user's goal, as the state-based approach does. Hence, event-based modeling cannot be used alone to model the user in open environment as he/she is trying to achieve the goal.

The challenge of modeling the user's behaviours in open environments is providing a dynamic plan that achieves the user's goals, and yet can be altered during execution based on the user's required responses to the dynamic changes in the environment.

2.4 Privacy in open environments

One of the major concerns when dealing with personal assistant in open environment is the privacy of users. The information that indicates the user's interests can be collected, process or disseminated by other entities of the environment when the personal assistant interact with other entities. Because this information is private and sensitive, exploiting it may violate users' privacy [4]. One of the challenges in privacy is identification; which is relating identity to sensitive information such as interests' information. Information can be used to directly identify users (identified information) or it might be used in processing information techniques and result in identifying users (Personally Identifiable Information (PII)). Because of the possibility of conversion of non PII to PII, differentiating the setting of these concepts becomes problematic.

There are several proposed approaches in various domains that utilize personal assistant [2], [3] to resolve identification complications while providing a relevant assistance. The basic approach of [2], [3] is on separating the interests information from the identity of the user. The interests' information may convey PII information which may initiate a process that violates users' privacy. Therefore, there should be additional solutions that address privacy challenges of exposing information while personal assistant interacts with the environment.

Typically, user's information is exchanged with other entities in return of receiving a service. These entities are called providers. In this context, the service they provide is referred as assistance. One of the approaches to resolve PII issues is to take the risk of disclosing information in interactions with providers into account. If the risk is acceptable in user's assessment, he/she can proceed with the interaction; otherwise information should not be disclosed to providers [4], [16].

As privacy violation in personal assistant in open environment is the direct or indirect consequence of exposing users' information such as his/her interests information to providers, including privacy protection measures in PSA is essential. However, because of the subjective nature of privacy and PII challenges, privacy solutions in PSA have to be flexible to be applicable with dynamism of open environments [4].

Privacy in the context of "information management" is categorized as "information collection", "information processing" and "information dissemination" [17]. These categories can be addressed in the pattern of interaction among providers and users. When users choose a provider, they need to interact with it to receive the requested service in terms of information (assistance). For instance, when the user attempts to see a video about a product, the user's interests will be collected by the provider of the video. The provider processes the collected information and utilizes it for profiling the user and utilizing it for promotion objectives. In some cases, providers disseminate users' information to others for further processes or with the objective of selling information which is the evidence of privacy violation [18]. Therefore, privacy solutions that target interaction of providers and users are essential in PSA.

3 The proposed architecture of PSA in open environments

Considering open environments is essential in building a personal smart assistant (PSA) that can be used to assist users in different domains. This paper proposes architecture of a personal smart assistant that can assist users in open environments; this architecture is shown in Figure 1.
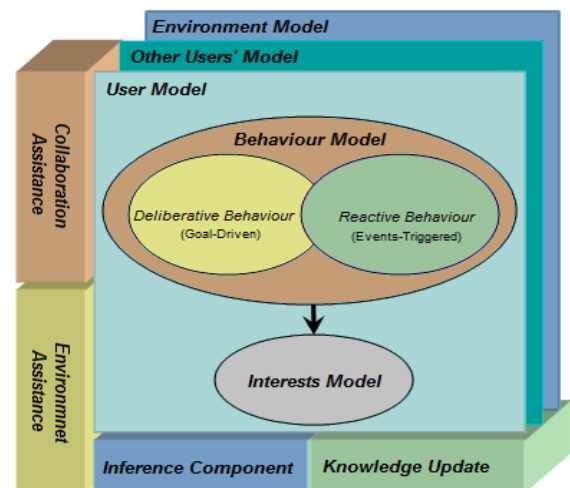
Figure 1: The proposed personal smart assistant architecture

The proposed PSA is able to model user's behaviours in open environments through extending the state-based user behaviour modeling with events that represent the changes that occur in the environment. Furthermore, the proposed PSA is able to protect the user's privacy as it interacts with other entities in the environment on behalf of the user. These two major contributions of the proposed architecture are discussed in the following subsections.

3.1 Extending state-based user behaviour modeling with events

The major component of the proposed personal smart assistant (PSA) is the user model. In this section we are proposing a user behaviour model that observers the user's deliberative, goal-driven actions, and reactive, event-driven actions. A dynamic plan of the goal-driven possible sequences of actions is generated and altered dynamically to contain reactive actions that are required to be performed by the user to response to changes in the open environment.

Before modeling the user, the domain at which the user is modeled should be represented. The entire collection of everything that exists is the world W, it is a set of the possible worlds, $W = \{w_1, w_2,...,w_k\}$. A state, s, can be defined as an instantaneous observation (snapshot) of the world. Several states can exist in one possible world w_i, $w_i = \{s_1,...,s_n\}$. An event e is defined as the occurrence of a change that triggers a response in the world. A state consists of concepts and relations. The existents in the world are called concepts C, C= $\{c_1,...,c_n\}$, where c_i is a single concept. The relations between these concepts are called conceptual relations R, R= $\{r_1,...,r_n\}$, where r_i is a single conceptual relation.

In order for users to achieve their goals, they need to transfer the world into the desired state. An assumption is made that the world stays at one state until an action occurs, therefore, if the current state is not the desired state, then the user needs to perform some actions to move the world to the desired state and achieve his/her goal. State-based modeling is used to model the user's behaviour as shown in Figure 2. Each node in Figure 2 represents a state of the world, the first node is the current state of the world S_c, and the last node represents the desired state S_d at which the user's goal is achieved. The arrow represents a transition between the states of the world; a_i is an action associated with this transition.
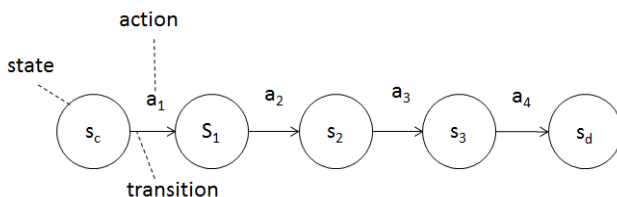
Figure 2: State-based user behaviour model

In [12] it is assumed that modeling the transition of the states is sufficient to model the user behaviour. However, we argue that in order to model the user behaviour, we also need to explicitly model the changes in the environment, and how the user is reacting to these changes, rather than just the transitions of the states. This is essential in an open environment where entities can enter and exit and any of them can change the states of the world at any time. State-based modeling is a goal-driven deliberative approach which needs to be extended to include event-driven reactive

approach. In this work, because we consider that the personal assistant is supposed to assist its user in open environment, we propose to extend the state-based modeling through modeling the events that happen in the environment explicitly as the states of the world are changing. An event is defined as a change in the environment that triggers a response, or "an external observable phenomenon, such as an environmental or a user stimulus, or a system response punctuating different stages of the system activity [14]". Hence, modeling the events explicitly will enable us to understand the behaviour of the user, as he/she acts not only to achieve the goal, but also, to react to these changes that might require the user's responses. Our work combines the deliberative approach and the reactive approach to model the user's behaviour. The next following subsections illustrate our proposed user behaviour model.

3.1.1 Modeling the actions

The user's behaviour is the sequence of actions that the user performs. Hence, in order to model the behaviour of the user with states and events, we need to model the actions. An action can be represented by i) an identification, ii) preconditions that should be satisfied for the action to take place, and iii) post conditions that are the impact of the action.

Action A = <ID, Pre-Conditions, Post-Conditions>

In this proposal, the user behaviour is modeled by events as well as states. Hence, for an action to occur, its preconditions are not only states, rather, events could also be required before an action can take place. Therefore, in this work, preconditions are modeled by states and events;

Pre-Condition = <State, Event>

Similarly, Post conditions, which are the impact of an action, are not only modeled as states, rather, events could also be an impact of an action. Therefore, in this work, post conditions are modeled by states and events.

Post-Condition = <State, Event>

Figure 3 shows how the state-based model is extended with events to model the user's actions. Each of the user's actions is either derived by the user's goal, and therefore the action is to transform the states of the world to the desired state s_d, or triggered by events, and therefore, they are responses to the events that are occurring in the environment. For an action to occur, specific states of the world should be reached and/or specific events should be triggered. Furthermore, if some events are triggered, which are preconditions of an action, these events cause an action to be performed as a response to this event, and therefore, the states of the world need to change to satisfy the rest of the preconditions of the action. Therefore, actions could also occur because of events, even though these actions might not help the user to achieve his/her goal. On the other hand, post-

conditions are the impact of the action on the environment. In this model, the action's impact is not strict to the states of the world; rather, actions could also trigger events in the environment, in other words, the environment responses to actions by producing events. The significance of extending the state-based model with events is to further analyze the user's behaviour in environments which he/she might not be the only actor. We do not model the actions occurring by other entities but we model the impact of their actions through the events that occur in the environment. So events can be produced by all entities in the environment and this does derive the user's behaviours.
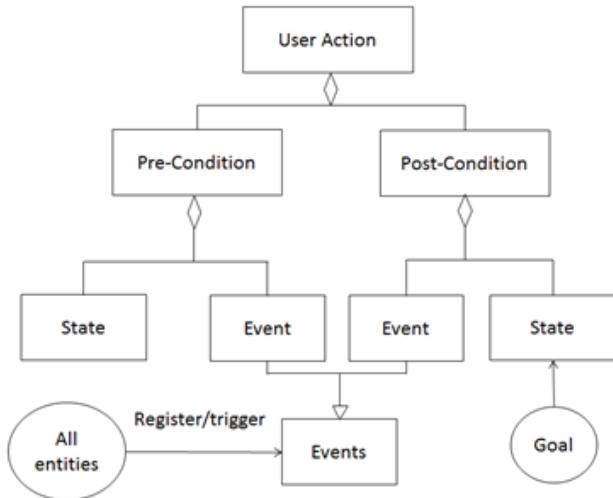
Figure 3: Proposed model of actions based on states and events

Modeling events alone does not represent the details of the system and how the possible states of the world are changing. Rather, both states and events need to be combined to model the user's behaviour. Figure 4 shows the extension of the modeled events to the state-based modeling for the user behaviour. The triggered events in the environment derive the actions of the user as he/she is trying to move the states of the world from the current state of the environment till the desired state. The dashed arrows in Figure 4 represent the actions' preconditions and post conditions in terms of events. For example, the dashed arrow from a_1 to e_1 suggests that; when the user performs a_1 to change the state of the world, e_1 is going to be triggered in the environment as a response to a_1, because e_1 is a post condition of a_1. When e_1 is triggered, the user needs to perform a_4, because e_1 is triggering a_4 and its part of its preconditions, hence a dashed arrow is directed from e_1 to a_4. The desired state could have been reached without performing a_4, however, because e_1 was triggered, the user has to response with a_4. Furthermore, some events could be triggered because of others' actions, but they still influence the user actions as he/she is trying to reach to the desired state.
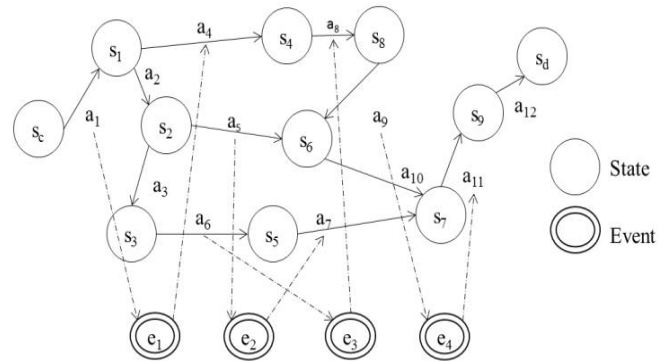
Figure 4: States and events to represent sequence of the user's actions

In this model, we are able to capture more realistic user behaviour by including actions that the user could perform in response to the environment rather than just the actions that he/she is performing to achieve the goal. All of these user actions are defined as the behaviour:

$$UB = <Actions>$$

3.1.2 Capturing the user's behaviours

It is necessary to capture the possible paths of actions that the user can follow and be able to predict the next likely action, and therefore, provide a proactive assistance that would help in achieving this action. We propose an algorithm that determines the different possible paths of actions based on the goal or any triggered event, and we refer to it as the path determination algorithm. The algorithm starts with the action a_g that can achieve the goal g (where g $\subseteq$ of post-conditions (a_g)), if the preconditions of a_g are part of the current state of the world, then a_g is the only required action to achieve the goal, however, if the preconditions of a_g are not part of the current states of the world, then this algorithm determine all the possible actions that can satisfy these preconditions. In other words, this algorithm determines all the paths of actions to reach from the current state till the desired state in which the goal is achieved. In the same way, this algorithm also identifies the possible paths of actions to response to events that are triggered in the environment. Uncertainty is considered when capturing the user's behaviour because of the dynamic nature of this behaviour. Therefore, the algorithm measures the prior probability of each action in the possible paths of actions. This prior probability is measured using binomial distribution [19], in which we measure the degree of belief that a certain action occurs in the environment given its preconditions and the overall number of actions in the environment. The pseudo code of the algorithm is given in the next section of this paper.

A probabilistic graphical modeling is used in the inference component of the PSA to predict the next action that the user is going to perform. In particular, a Bayesian network[1] is constructed using the different paths of actions that are generated from the path determination algorithm, as shown in Figure 5. Each form of arrows (straight, dotted and dashed) in Figure 5 represents a unique sequence of actions, and each action is associated with a prior probability that is calculated using binomial distribution. When the next action needs to be predicted, The Bayesian network graph, shown in Figure 5, is used to calculate the probability of each child based on the observed value of its parents. From each possible sequence, the child is the final action in that path, and the last parent is the current action that the user performed. The path with the highest probability will be considered as the predicted path and the first action of that path, after the current action that is already performed, is considered as the next predicted action to be performed by the user. After modeling the user's behaviours and forming the Bayesian network, the inference component is responsible to infer this network using inference algorithms to identify the user's next predict action based on his/her current performed actions.
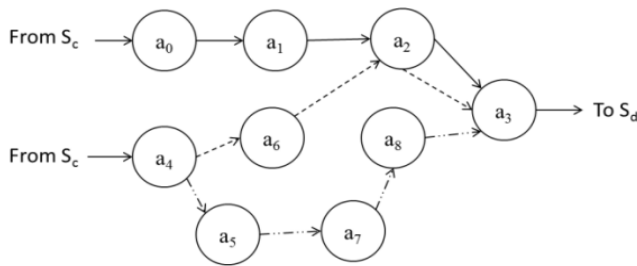
Figure 5: Bayesian network for the user's actions

In addition to modeling the user's behaviours, our user model also models the user's interests. In the proposed architecture, Figure 1, the user's interests are derived by the user's behaviours, in other words, the user's behaviour is an indication of the parts of the world that the user is interested in. The user behaviour is the sequence of actions that the user performs in the environment. These actions include preconditions, which are the requirements for the action to take place. Therefore, to move towards the desired state, and to achieve the goal, the requirements for every action are part of the user's interests. Furthermore, to realize what these actions are achieving, the impacts from these actions are also interesting to the user, these impacts are represented as post-conditions, the post-conditions of the last action performed by the user include the goal, and hence the interests of a user is the combination of preconditions and post-conditions of the user's performed and predicted actions.

3.2 Enabling the PSA to protect the user's privacy in open environment

The environment model of the PSA's architecture, Figure 1, represents the context of the environment and captures the actions that are actually performed by the user, as well as the events that are triggered in the environment. Additionally, the environment model identifies any potential assistance that exists in the environment, and forwards its information to the environment assistance component to be matched with the contents of the user model to identify the available relevant assistances. Accordingly, PSA may need to interact with entities of the environment to collect necessary information. The interaction asserts the exchange of information that raises privacy concerns. One of the major contributions of the proposed PSA is to ensure that the user's privacy is protected in open environments. Therefore, among relevant assistances that are found based on the matching results, only the ones that will not violate user's privacy will be recommended to the user.

One of the approaches to protect privacy is investigating how entities interact and what information is exchanged among them [16]. In the context of personal assistant and open environment, interaction happens between PSA and providers. The exchanged information is users' interest. In this work, we have assumed that PSA is part of user's device, software, etc. and will not exploit the information.

Fundamentally, privacy violation refers to an interaction that by exposing information to an entity to achieve a goal, it causes the other goal of the user becomes not achievable [17]. The goals do not have conflicts. In the context of PSA, the goal of the user in the world is given. However, the user's goals are not limited to the one that he/she is trying to achieve at a specific point of time. For instance, the user might be in a grocery store and her/his goal is to purchase milk and bread. But at the same time he/she may aim to get a job, open a bank account and receive a health service. Although receiving an ad and showing interest for viewing an assistance don't have conflict with other goals of the user, exposing interest information in one domain may prohibit the user to achieve his/her goals in other domains. Therefore, for resolving privacy challenges, we extend the world to the user and a set of domains. PSA can be specialized for each of these domains.

$$GlobalWorld(GW) = < User, \{X_1, \ldots \ldots, X_k\} >$$

Each domain includes providers, the world of the domain and the goal of the user in that domain.

$$X_i = < W, \text{Providers}, \text{Gl}>$$

Each user has interest information that providers require to collect for processing and tracking user's behaviour [2]. Assistance (As) are presented (*Promote*) to the user based on users' interest that is exposed to the provider (*Exposure*).

[1] The Bayesian Network is a probabilistic graph that calculates the conditional probability of a node based on the probabilities of its parents [8].

Providers convey operations (Op) to process users' interests such as tracking and profiling. The purpose (Pu) of applying operations on users' interest is to send relevant assistance.

$$User =< Interest >$$

$$< Interest > = \{interest_1, \ldots \ldots, interest_l\}$$

$$Providers = \{Provider_1, \ldots \ldots, Provider_l\}$$

$$Provider = < As, Op, Pu >$$

Accordingly, Privacy Violation (PV) can be formulated as the following:

$$PV(Interest, GW) =$$
$$\begin{cases} TRUE & if \ \exists \ x_i, x_j \in GW \ \exists \ gl \in x_j. Gl \ \wedge \\ & \exists \ provider \in X_i. Providers \ \wedge \\ \\ & Exposure \ (Interest) \rightarrow Promote(Provider. As) \ \wedge \\ & Achievable(gl) = FALSE \\ FALSE & if \ otherwise \end{cases}$$

Each provider needs to process the user's interest information after it is exposed. They apply Op to infer information they need for promoting an As. Op can be integrated with countermeasure techniques for privacy violation such as anonymization and legal agreements. That indicates a level of Privacy Protection (PP) which is applying operations that prevent privacy violation.

$$PP \ (Interest, Op) = \neg \ PV(Op(Interest), GW)$$

Countermeasure techniques for protecting privacy may not be able to fully prohibit privacy violation. Therefore, for evaluating privacy in an interaction among entities, the probability of privacy protection in operations of a provider has significant impact. providers may contain various operations. Each of them has specific probability for protecting privacy. The minimum probability of privacy protection is the value that the user requires for evaluating the privacy of the interaction. This concept is called Privacy Protection Level (PPL).

$$PPL(Provider, Interest >= Min$$
$$\begin{pmatrix} Prob((PP(Interest, Provider.Op)==TRUE) \\ \left[\forall \ op \in Provider.OP \ | Interest \in op_{\underline{Domain}} \right] \end{pmatrix}$$

Evaluating PPL is out of the scope of this paper. However, as PII 2.0 [4] proposes, this value is calculated based on the risk assessment of exposing interest to a provider. In an interaction, there are significant elements that influence privacy evaluation such as trust level, information gain of the interest, the agreement between the user and providers, the sensitivity, criticality and cost of the interest, purpose of collecting the interest, the severity operation that is going to be applied on interest [20]. In the context of personal

assistant, the purpose of collecting processing and disseminating information is providing relevant assistance. Typically, the operations that they apply on interest information are tracking and profiling that has high severity. The risk assessment model proposed in [20] collects this information from an interaction and calculates the risk of privacy violation in the given interaction. Utilizing the risk value results in evaluating PPL. Providers and users are aware of the risk assessment model. Because providers require attracting engagement of users to their assistance, they provide their capabilities in protecting users' privacy by informing their PPL value in their assistance. Therefore, the user can compare their PPL with the PPL that he/she has calculated. If it is lower than the expected PPL, the promotion will not be promoted.

PPL will be calculated at interaction level and will be presented to PSA. PSA finds the relevant assistance and exclude the ones that don't have enough PPL.

4   Implementation

This section presents the major components and algorithms used to implement the proposed PSA.

4.1   The personal smart assistant model

The personal smart assistant (PSA) needs to model the user and make decisions in terms of suggested assistance based on this modeling. It should act autonomously to reason about the user's behaviours and interests and provide the relevant assistance based on its problem solving ability. Furthermore, the PSA needs to communicate with other entities in the environment to provide the assistance to his/her user. Consequently, the PSA needs to have characteristics such as autonomy, cooperation and intelligence to be able to provide its assistance to the user; these characteristics are ensured in the CIR-Agent Model [21]. The Coordinated Intelligent Rational (CIR) provides a generic model for an agent in cooperative distributed systems. The CIR-Agent contains capabilities such as autonomy, cooperation, intelligence and rationality, and hence, it's able to model the proposed personal smart assistant.

4.2   Path determination algorithm

An important component of the implemented PSA is the path determination algorithm, which determines the possible paths of actions that the user should perform when an event is triggered in the environment. The PSA implementation contains several key objects such as Action, Preconditions, and Post-conditions that are utilized in this algorithm as shown in the algorithm's pseudo code in Figure 6.

```
Input: Goal or Event Actions (Actions which contains the desired state or triggered event in their post-
conditions)

Function Determine Path (ArrayList of inputActions)

1) For each Action A in the input ArrayList
2)      Measure Prior Belief of Action A
3)      Add to actionsInPath ArrayList
4)      If the Current States of the World contains all Preconditions of action A
5)          Return actionsInPath ArrayList
6)          End Algorithm.
7)      Else
8)          For All Actions in Environment do:
9)              If Action A's preconditions contains an environment's actions post-conditions
10)                 Add this environment action to requiredActions ArrayList
11)     Recursively Call Determine Path (ArrayList of requiredActions)
```

Figure 6: Path Determination Algorithm's Pseudo Code

4.3 Belief system

The degree of belief that a certain action will be performed next by the user is calculated at two levels. First a prior belief is measured that a certain action will be performed using binomial distribution. As indicated in the Figure 6, this belief is measured during the path determination algorithm. Every action that is produced from the path determination algorithm contains a prior probability attribute that its value is measured through the binomial distribution. The next stage in measuring the belief that a certain action will be performed next is to consider conditional probabilities; the probabilities of performing the action given that other actions are performed, this can be calculated by a Bayesian Network. JavaBayes [22] is used to provide a java library to construct the Bayesian network.

5 Conclusions and future work

This paper proposed a Personal Smart Assistant (PSA) agent that is able to assist users to achieve their goals in open environments. The proposed PSA combines a deliberative, goal-driven, approach with a reactive, event-driven approach to model the user's behaviours. Such a combination guarantees a more accurate modeling for the user behaviours in an open environment that is constantly changing with unexpected events. Also, our approach protect the user's privacy through comparing the user's Privacy Protection Level (PPL), with the PPLs of the assistance provides, unless the required PPL is matched, the assistance will not be recommended to the user regardless of its relevance. Our proposed PSA architecture and its major components and algorithms were addressed in this paper to give insight about its implementation.

In the future, an enriched environment should be created and integrated with the proposed implemented PSA's. This environment should reflect real life environments that contain many different actions for the user to do, as well as enriched goals that the users are trying to achieve. Such environments can provide several scenarios and hence many different realistic trials can be recorded. Furthermore, the environment should be able to capture the user's feedback about the assistances provided by the PSA. Building such enrich environments provide richer results to validate the proposed PSA.

Additionally, we are looking at extending the user's behaviours model to include patterns of the user's actual pervious behaviours as a prior knowledge that can help to make more accurate predictions. Sometimes, users have their own preferred patterns and sequences of actions for achieving their goals, so the PSA should be able to extract the usual patterns of behaviours and then predict the next actions based on those patterns.

6 References

[1] Y. Zhang, H. Ghenniwa, and W. Shen. Enhancing intelligent user assistance in collaborative design environments. *In Proceedings of the ninth international conference on computer supported cooperative work in design (CSCWD'05)*, pp. 107–112, 2005.

[2] V. Toubiana, H. Nissenbaum, A. Narayanan, S. Barocas and D. Boneh. Adnostic: Privacy preserving targeted advertising. *In Proceedings of the Network and Distributed System Security Symposium,* 2010.

[3] A. Juels. Targeted advertising ... and privacy too. *In Proceedings of the 2001 Conference on Topics in Cryptology*, pp. 408-424, 2001.

[4] Paul M. Schwartz, Daniel J. Solove. The PII problem: Privacy and a new concept of personally identifiable information. 2011.

[5] S. Schiaffino and A. Amandi, Intelligent user profiling. *Artificial* Intelligence: an international perspective, Vol. 1, pp 193-216, 2009.

[6] S. Wu, H. Ghenniwa, W. Shen, and K. Ma, Intelligent User Assistance in Collaborative Design Environments, *In Proceedings of the 8th International Conference on Computer Supported Cooperative Work*, pp. 259-266, 2004.

[7] E. Shakshuki, H. Ghenniwa and M. Kamel, "Agent-Based System Architecture for Dynamic and Open Environments," *Journal of Information Technology and Decision Making*, vol.2 no. 1, pp. 105-133, 2003.

[8] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed., Pearson Inc., 2010.

[9] A. McNeile and N. Simons, "Methods of Behaviour Modelling: A Commentary on Behaviour Modelling Techniques for MDA". Metamaxim Ltd, 2004, DRAFT Version 3.

[10] R. Fikes and N. Nilsson, STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving, *Artificial Intelligence* vol. 2(3/4), pp. 189-208, 1971.

[11] C. Burghardt, M. Giersich and T. Kirste, Synthesizing probabilistic models for team activities using partial order planning, *In Proceedings of Ambient Intelligence Workshop,* Germany, 2007.

[12] S. Wu, H. Ghenniwa, W. Shen and Y. Zhang, Personal assistant agents for collaborative design environments, *Computers in Industry*, vol. 57 (8/9), pp. 732–739, 2006.

[13] G. Allen and S. March, The Effects of State-Based and Event-Based Data Representation on User Performance in Query Formulation Tasks, *MIS Quarterly,* vol. 30, pp. 269-290, June 2006.

[14] F. Belli, C. Budnik, and W. Wong, Basic Operations for Generating Behavioral Mutants, *In Proceedings of th 2nd Workshop Mutation Analysis,* 2006, pp. 9-18.

[15] Y. Cao, L. Tao, and G. Xu, An event-driven context model in elderly health monitoring, *in Ubiquitous, Autonomic and Trusted Computing, Symposia and Workshops* (2009), pp. 120–124.

[16] S. Lederer, A. K. Dey and J. Mankoff. A conceptual model and a metaphor of everyday privacy in ubiquitous. University of California at Berkeley. Berkeley, CA, USA. 2002.

[17] D. J. Solove. Understanding Privacy 2008 (v.10). Available: http://books.google.ca/books?id=XU5-AAAAMAAJ.

[18] S. Spiekermann and L. F. Cranor. Engineering privacy. IEEE Trans. Software Eng. 35(1), pp. 67-82. 2009.

[19] W. Bolstad, *Introduction to Bayesian Statistics,* 2nd ed.*,* John Wiley & Sons Inc., 2007.

[20] A. Samani, H. H. Ghenniwa and J. Samarabandu. Risk-based modelling for managing privacy protection. Presented at Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference, 2012.

[21] H. Ghenniwa and M. Kamel, "Interaction devices for coordinating cooperative distributed systems," Automation and Soft Computing, vol. 6, no. 2, pp.173-184, 2000.

[22] F. Gagliardi, JavaBayes - Bayesian Networks in Java, [online], Available: http://www.cs.cmu.edu/~javabayes/Home/

Simultaneous Events Handling

N. Taylor[1], B. Yang[2]
[1]Computer Science Department, Bowie State University, Bowie, MD, USA
[2]Computer Science Department, Bowie State University, Bowie, MD, USA

Abstract— *During interactions of embedded systems in robots, thousands of transistors may be active at once, even though only one computational process is underway to achieve a target simultaneous goal. The challenge this presents to the task of intelligent agent systems has spawned increasing interest in agent complex concurrent event processing techniques and optimizing synchronous processes. This paper will provide an examination of simultaneous agent event handling literature, which assesses the agent's ability to keep two events going on at once. Literature in this area has largely focused on issues common to specific subareas such as Markov decision processes (MDPs), actions in a special kind of decision problem known as the semi-Markov decision process, or SMDP and robotics. In this review, these subareas will be explored through examining various approaches to simultaneous events handling, event-based triggering processes, concurrent action modeling, and grasp and control transport algorithm.*

Keywords: algorithm, error, resilience, communication, networks

I. INTRODUCTION

If given the opportunity to assess the ability to keep two events going on at once, people will generally report difficulty only if one of the tasks is intellectually demanding. For example, people may exhibit a harder time carrying on a serious discussion while adding up a restaurant check as opposed to performing tasks that are a part of their daily routine.[9] Despite the difficulty that may be exhibited in simultaneously performing multiple intellectually demanding tasks, people are quite sure that they can handle at least two tasks simultaneously. Recent research suggests that these assessments of the human capacity for simultaneous event handling may be over-optimistic and it appears that certain mental operations are "bottlenecks" that require the exclusive use of some cognitive resources and therefore cannot be done concurrently [8]. Much of what people perceive as simultaneous event handling in mental life is very similar to computer time sharing, in which some mental operations are carried out one at a time, much like time division multiplexing.

In recent years, there has been increased interest in decentralized approaches to solving complex real-world problems. These approaches fall into the area of distributed systems, where a number of entities work together to cooperatively solve problems. The combination of distributed systems and artificial intelligence (AI) is collectively known as distributed artificial intelligence (DAI). Traditionally, DAI is divided into two areas [23]. The first area, distributed problem solving, is usually concerned with the decomposition and distribution of a problem solving process among multiple slave nodes, and the collective construction of a solution to the problem. The second area, multi-agent systems (MAS), emphasizes the joint behaviors of agents with some degree of autonomy and the complexities arising from their interactions. Many kinds of artificial intelligent agents have some degree of autonomy [7].

In this review of literature, a focus will be given on autonomy artificial intelligent agents (robots) performing two or more tasks at the same time. Autonomous robots are robots that can perform desired tasks in unstructured environments without continuous human guidance and also have the ability to discover on their own, often through repeated trials, how to solve a given task or to minimize error in all environments whether on land, underwater, in the air, underground, or in space. Some modern factory robots are "autonomous" within the strict confines of their direct environment, yet the introduction of chaotic, unpredicted variables within the factory robot's workplace can pose a challenge to the efficiency of the factory robot's autonomy. The exact orientation and position of the next object of work (in the more advanced factories) and even the type of object and the required task must be determined, which from the robot's perspective can vary unpredictably. A fully autonomous robot has the ability to gain information about the environment (Rule #1), work for an extended period without human intervention (Rule #2), move either all or part of itself throughout its operating environment without human assistance (Rule #3), and avoid situations that are harmful to people, property, or itself unless those are part of its design specifications (Rule #4) [22]. An autonomous robot may also learn or gain new capabilities like adjusting strategies for accomplishing its task(s) or adapting to changing surroundings. The degree of autonomy of an agent's ability to simultaneously handle events shows how advance the agent is or its level of sophistication.

II. SURVEY STRUCTURE AND TAXONOMY

There are two major categories of simultaneous event handling either by autonomous or non-autonomous agents. The first category is single agents in which agent's behavior can be controlled by outside forces. Single agent

simultaneous event handling approaches typically follow traditional artificial intelligent agent action methods, but they may have scalability problems as the number of actions increases with complexities [8]. A second category of simultaneous event handling agents is cooperative multi-agents which uses multiple concurrent learning processes and interacting intelligent agents within an environment to achieve a goal.[19] These agents typically employ dependency mechanisms for each team member, seeking to reduce the joint space by projecting it into N separate spaces. The presence of multiple agents' concurrent learning and interacting makes the environment non-stationary. However, it also makes event handling and learning techniques a very complex action (like cognitive multi-agent, which has a lot of complex calculations). To illustrate the difference between the two, consider the lifting and dropping two boxes (box A and box B) at the same time scenario, where the task is to drop off the two boxes at the same time. [18]The single agent can make a copy of lift and drop of action policy and execute two policies at the same time while the multi agents will have to communicate among each other before deciding on which policy an individual team member will execute in order to achieve the task there by affecting their performance. For some multi-agent goals such as simultaneous event handling, communication is a necessity and for others, communication may nonetheless increase agent performance. Communication is defined as altering the state of the environment such that other agents can perceive the modification and decode information from it. Among other reasons, agents communicate in order to coordinate more effectively, to distribute more accurate models of the environment, and to learn subtask solutions from one another. But are communicating agents really multi-agent? Stone and Veloso [22] argue that unrestricted communication reduces a multi-agent system to something isomorphic to a single-agent system. They do this by noting that without restrictions, the agents can send complete external state information to a "central agent", and execute its commands in lock-step, acting as effectors for the central agent. A central agent may not even be necessary, as long as agents can receive all the information they need to know about the current states of all the other agents, allowing them to make independent decisions knowing exactly what the other agents will do. This allows a "central controller" on-board within each individual agent, picking the proper sub-action for the full joint action. This may lead to the belief that a true multi-agent can lead to restrictions on communication. While full, unrestricted communication can orthogonalize the learning problem into a basic single-agent problem, such an approach requires very fast communication of large amounts of information.

Real-time applications instead place considerable restrictions on communication, in terms of both throughput and latency. Unfortunately, learning is a challenging issue in itself, and difficulties associated with it often result in a simplified approach to communication, usually neglecting costs of communication with other agents. There also is a need for further research to address the issue of using selective communication only when necessary. Explicit communication can also significantly increase the learning method's search space, both by increasing the size of the external state available to the agent (it now knows state information communicated from other agents), and by increasing the agent's available choices (perhaps by adding a "communicate with agent i" action). As should be noted [21], this increase in search space can hamper learning an optimal behavior by more than communication itself may help. Even when communication is required for optimal performance, for many applications, the learning method must disregard communication or hard-code it, in order to simplify the learning process. Section 2-4 of this review explores various simultaneous event handling approaches. The review concludes with a thorough comparison of the various approaches and how the research conducted in this area can aid in the future direction of simultaneous event handling research.

III. THE EVENT-BASED TRIGGERING PROCESS

An agent's detection of run-time events triggers are a subset of its possible actions. As an agent operates in its environment, [25] it detects a stream of perceptual and cognitive events (changes), which trigger a stream of possible actions. Trigger refers to an action's enabling conditions being satisfied, binding its parameters in the current context, and producing an executable instance of the action's execution code. Thus, a "possible action" is a fully instantiated action, whose execution code the agent could execute "now" if it chose to do so. For example, a hypothetical "errand robot" as it drives east on Jackson Avenue, one of the streets in its environment, the errand robot) might know the action:

Name: Evaluate-posted-sales
    Trigger-condition: Detect-sale-sign at @site.
    Action1: Evaluate-sale at @site.
    Execution-code: ...

While driving east on Jackson Avenue, the robot might detect sale signs at three stores, potentially triggering three possible actions:
    Evaluate-sale at Truc.
    Evaluate-sale at Manny's.
    Evaluate-sale at Artifactory.

Depending on run-time conditions, the robot run-time has sequential, simultaneous and many other action conditions for multiple signs. The choice of simultaneous or sequential depends on the errand robots environment and time availability (time constrains). If it requires all three signs evaluation action at same time(),that is execute "Evaluate-sale" actions at @site with 00:00:00 seconds time difference, then the robot will do so simultaneously. Otherwise, many actions may be logically applicable--that is, their trigger

conditions may be satisfied, given the state of the environment. An agent is given goals to achieve and limited resources, therefore typically noticing only a subset of run-time conditions. Goals and limited resources also lead to only a subset of possible actions being triggered and only a subset of triggered actions being executed. For example, the errand robot is executing a single possible action, though it has triggered but not yet executed three other possible actions, and it has not triggered two other actions that are logically applicable. [24] The robot may or may not subsequently execute the three triggered actions or trigger the two logically applicable actions, depending on what other events occur, how it responds to them, etc. Naturally, an agent is biased to trigger and execute actions that are relevant to its goals. This bias may be strong or weak, depending on the specificity of the goals and the availability of computational and real-time resources. In the aforementioned example, ("Survey-sales at clothing stores") will lead the agent to more likely notice sale signs at clothing stores and to trigger and execute associated "Evaluate-sale" actions. As an additional condition, if the robot was late for an important appointment and there is a parking spot and a sales sign, it would be more likely to notice conditions and trigger actions related to parking (trying to park) and "Evaluate-sale" actions simultaneously. Though bias affects probability, it does affect the possibility of triggering and executing logically applicable actions.

IV.   CONCURRENT ACTION MODEL

Much work conducted on concurrency has focused on unit step actions and a plethora of research has been conducted on concurrent processes, dynamic logic, and temporal logic [1, 3, 4]. Rohanimanesh [5] and Mahadevan [19] provide an alternative perspective to the traditional approach and explore parallelizing temporally extended actions, considering temporal abstraction within the framework of reinforcement learning and Markov decision processes (MDPs).Options are closely related to the actions in a special kind of decision problem known as a semi-Markov decision process, or SMDP. A MDP with a fixed set of options is an SMDP, as is stated formally below. Although this fact follows more or less immediately from definitions, it is presented as a theorem to highlight and state explicitly its conditions and consequences: Theorem 1: (MDP + Options = SMDP) [6]. For any MDP and any set of options defined on that MDP, the decision process that may be selected is only among those options, executing each to termination, becoming an SMDP [7]. Building upon the SMDP framework, Sutton and colleagues introduce the Concurrent Action Model (CAM) (S;A; T ;R), where S is a set of states, A is a set of primary actions, T is a transition probability distribution s x α (A) x s x N $\rightarrow$ [0; 1], where (A) is the power-set of the primary actions, N is the set of natural numbers, and R is the reward function mapping S $\rightarrow$ R .[13] Here, a concurrent action is represented as a set of primary actions (hereafter called a multi-action), where each primary action is either a single step action, or a temporally extended action (e.g., modeled as

a closed loop policy over single step actions [7]). We denote the set of multi-actions that can be executed in a state s by A(s). In practice, this function captures resource constraints that limit how many actions an agent can execute in parallel and thus, the transition probability distribution in practice may be defined over a much smaller subset than the power-set of primary actions. Since every primary action in a multi-action may not terminate at the same time, termination of a multi-action can be defined in many ways. Three termination schemes are illustrated in Figure 1

Figure 1: Left: T_{any} termination scheme. Middle: T_{all} termination scheme. Right: $T_{continue}$ termination scheme.
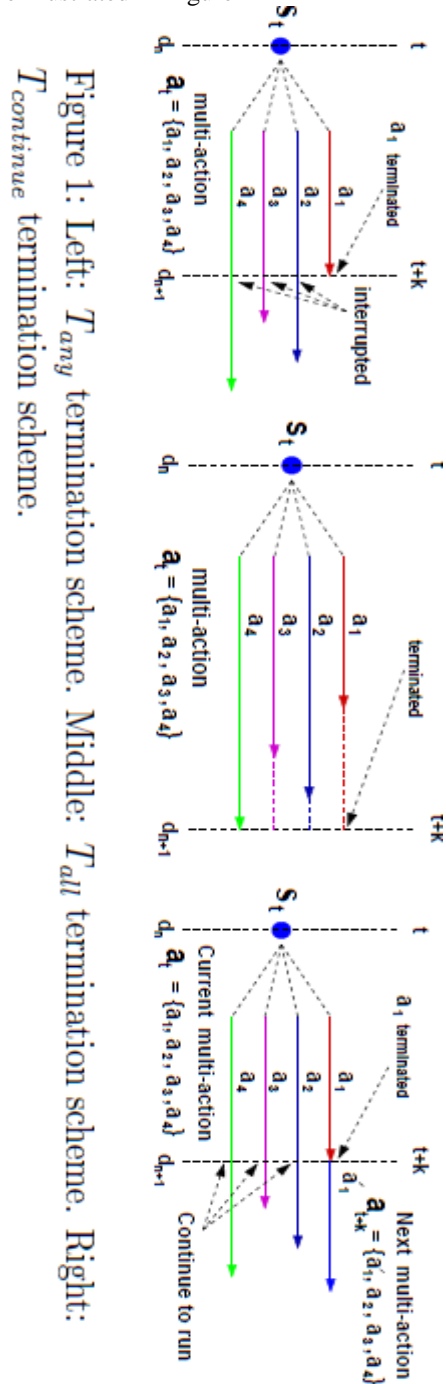
Using a grid world task comparing various termination schemes (see Figure 2), an agent with the ability to execute

two types of action concurrently is shown: (1) navigation actions, and (2) key actions. Navigation actions include a set of one-step stochastic navigation actions (Up, Left, Down and Right) that move the agent in the corresponding direction with probability 0.9 and failure at a probability of 0.1. Upon failure, the agent moves instead in one of the other three directions, each with probability 1/30. There is also a set of temporally extended actions defined over the one step navigation actions that transport the agent from within the room to one of the two hallway cells leading out of the room.

Figure 2 : A navigation problem that requires concurrent plans. There are two locks on each door, which need to be opened simultaneously. Retrieving each key takes different amounts of time.
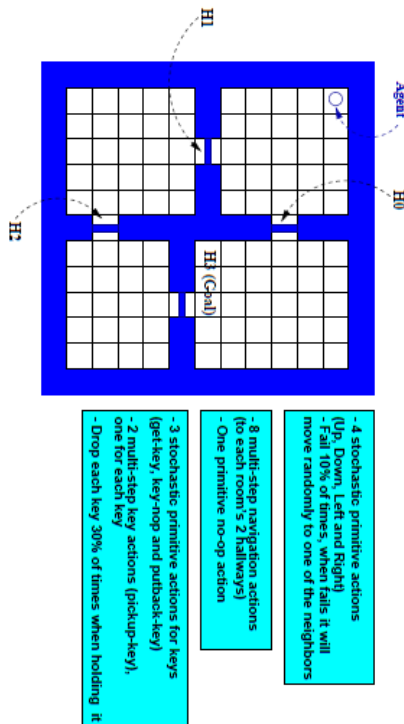
In the example presented, navigation actions can be executed concurrently with key actions. Actions that manipulate deferent keys can be also executed concurrently. Though actions may be executed concurrently, the agent is not allowed to execute more than one navigation action or more than one key action (from the same key action set) concurrently. In order to properly handle concurrent execution of actions, we have used a factored state space defined by state variables position (104 positions), key1-state (11 states) and key2-state (7states).

V. GRASP AND CONTROL TRANSPORT ALGORITHM

Yamasaki [20] and colleagues describe a method for changing the control mode based on position and force errors called the SCOME. This method is applied to the hand of a two-fingered robot called a chopsticks-type robot and is composed of elastic joints. The two-fingered robot is tasked with grasping and transporting a cylindrical object. A control strategy for the robot is first proposed by the authors, based on a desired value of position and force for grasping and transporting an unknown object. The SCOME method continues as the control mode is selected based on the characteristic curve of position vs. force where the target value of work required to reach the desired position and force is satisfied. Next, a grasp and transport control experiment is conducted using the robot to manipulate various objects made of different materials and with various outer diameters.

Figures 3(a)-3(g) show the algorithm for grasp and transport control. Subscripts 1, 2, and 3 represent Fingers 1 and 2 and the wrist, respectively, being that a cylindrical object is handled. First, the robot hand performs a 'search,' as shown in Fig. 3(a), by moving the positions of Fingers 1 and 2 to the desired position of $0 \to \pi / 2$ (rad). When either finger comes in contact with an object during the search, as shown in Fig.3(b), the desired torque of 0.1 (Nm) is satisfied, and position control is switched to torque control, shifting to the action of 'contact.' Contact is defined as the desired torque by either finger becoming satisfied during searching. In Fig. 3(b), Finger 1 comes in contact with an object before Finger 2. Finger 1 is stopped by the torque control, whereas Finger 2 increases the value of the desired position to $\pi / 2 \to \pi$ (rad) and position control is maintained until it comes in contact with the object. Upon contact with the object, Finger 2 also switches from position control to torque control, and the robot hand shifts the action to 'grasp' while both fingers are in contact with the object (Fig. 3(c)). Subsequently, the action of the robot hand shifts to 'lifting' while grasping the object (Fig. 3(d)). At this time, the torque is controlled so that it remains at the desired value of 0.5 (Nm) for each finger. Next, the robot hand shifts the action from lifting to 'transport.' The mechanism of the control of each finger during transporting was designed to have a leader-follower style, in which the front finger in the transporting direction (Finger 2) is under torque control and the rear finger (Finger 1) is under position control, as shown in Fig. 3(e). The desired position and torque during transporting for Finger 1 was set to be between the current value and $\pi / 2$ (rad) and 3 (Nm), and those for Finger 2 was set to π (rad) and 0.5 (Nm), respectively. After transporting, the robot hand puts 'down' the object (Fig. 3(f)), and 'returns' to the original state (Fig. 3(g)), completing the series of operation of the robot. The actions of lifting (Fig.3(d)) and putting down (Fig. 3(f)) are carried out by the wrist mechanism.
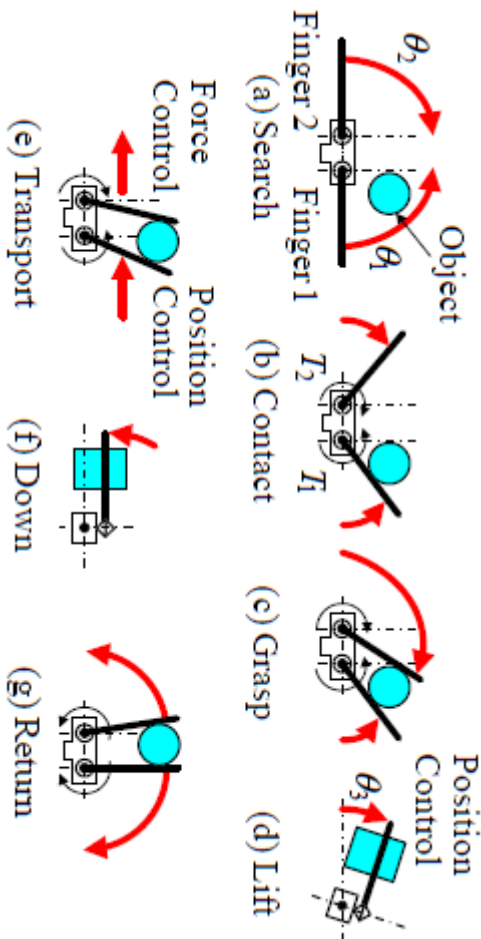
Figure 3. Operation algorithm for manipulating an object.

The desired position of the wrist was set to $0 \leftrightarrow \pi/18$ (rad). When no object is present within reach of the robot hand, both fingers have nothing to come in contact with during the search, therefore, upon the completion of 'search,' the action of the robot is shifted to the 'return' step, which has a duration of 5 (s), after which the entire series of operations is completed. Based on the SCOME method, researchers were successfully able to construct a chopsticks-type robot based on the two-fingered robot hand leading to the development of the control algorithm to grasp, transport, and release cylindrical objects, develop a new control mode selector in relation to contact with objects at a small torque, and demonstrate validity of the algorithm for grasp and control.

VI. CONCLUSIONS

Multi-action at the same time or concurrent task learning is not a relatively new area for researchers and is filled with complex and rich dynamics, holding the promise of widespread applicability. Traditionally, researchers have approached the problem from a number of different directions. The various approaches to simultaneous event handling as reviewed in this paper demonstrates that an agent can adopt different control modes, dependent on the situation presented. Depending on the predictability of its environment

and the constraint imposed by its goals, the agent in all the approaches discussed (The Event-Based Triggering Process, Concurrent Action Model and Grasp and Control Transport Algorithm) modulate its sensitivity to run-time events and its commitment to specific actions. In the Event-Based triggering simultaneous event handling approach, the effects of increasing the specificity of an agent's strategic plan leads to: (a)The agent spending more time matching possible actions to strategic plans on each reasoning cycle, requiring fewer reasoning cycles to complete its task. Depending on the magnitude of match time vs. other cycle functions, this relation entails a quantitative prediction of changes in total task time. (b) The agent ignoring a greater number of possible actions, but more reliably executing those favored by its plan. Depending on the distribution of run-time events vs. the class of possible actions favored by the plan, this relation entails a quantitative prediction of changes in the global utility of the agent's behavior. In the concurrent action model, specifying the A(s) (get-key, putback-key, pickup-key, etc) set of applicable multi-actions might significantly reduce the set of choices available, causing additional mechanisms for efficiently searching the space of multi-actions that can run simultaneously to occur. The grasp and control transport algorithm approach has the control algorithm to grasp, transport, and release objects, whose position is unknown simultaneously. The two-fingered robot hand base uses the same action policy with position modification and time variance to model with an elastic joint. Additionally, the approaches to simultaneous events handling that are examined in this review also differ in that the processes utilized by each approach varies, yet arrive at very similar conclusions in regards to simultaneous event handling and agent knowledge. The event-based triggering process utilizes a process that considers the run-time conditions of a process, dictating whether an event will occur simultaneously or sequentially. This process of an agent choosing among its possible actions is dependent on triggers that will aid in ultimate decisions being made, and thus this process is highly influenced by triggers, biased towards goals, and available options. The concurrent action model, very similarly to the event-based triggering process model, utilizes a process that is dependent on triggers, or in the concurrent action model key actions that influence the completion of an action. The final approach reviewed was the grasp and control algorithm approach which utilized a process of developing the SCOME method to devise an algorithm that was further used to investigate the process of grasping and transporting by agents. The various processes used in exploring simultaneous event handling further emphasizes the dynamic and vast arena that this topic area encompasses and provides multiple perspectives in understanding agents and their ability to simultaneously perform tasks.

VII. ACKNOWLEDGMENTS

We would like to acknowledge Yojiro Yamasaki, Toru Tsumugiwa, Ryuichi Yokogawa who first proposed a method of work control to maintain a constant amount of work done by the robot on an object. We would also like to acknowledge Khashayar Rohanimanesh and Sridhar Mahadevan who investigated a general semi-Markov Decision Process (SMDP) framework for modeling concurrent decision making, where agents learn optimal plans over concurrent temporally extended actions, thereby introducing three types of parallel termination schemes. Finally, we wish to acknowledge Hayes-Roth, B. for noting that Planning and reactive models embody two different modes of control in the Opportunistic control of action in intelligent agents paper.

VIII. REFERENCES

[1]   Craig Boutilier and Ronen Brafman. Planning with concurrent interacting actions. In Proceedings of the Fourteenth National Conference on Arti_cial Intelligence (AAAI '97), 1997.

[2]   P. Cichosz. Learning multidimensional control actions from delayed reinforcements. In Eighth In-ternational Symposium on System-Modelling-Control (SMC-8), Zakopane, Poland, 1995.

[3]   C. A. Knoblock. Generating parallel execution plans with a partial-order planner. In Proceedings of the Second International Conference on Arti_cial Intelligence Planning Systems , Chicago, IL,1994., 1994.

[4]   Ray Reiter. Natural actions, concurrency and continuous time in the situation calculus. Principles of Knowledge Representation and Reasoning: Proceedings of the Fifth International Conference (KR'96), Cambridge MA., November 5-8, 1996, 1996.

[5]   Khashayar Rohanimanesh and Sridhar Mahadevan. Decision-theoretic planning with concurrent temporally extended actions. In Proceedings of the 17th Conference on Uncertainty in Arti_cial Intelligence, 2001.

[6]   S. Singh and David Cohn. How to dynamically merge markov decision processes. Proceedings of NIPS 11, 1998.

[7]   R. Sutton, D. Precup, and S. Singh. Between MDPs and Semi-MDPs: A framework for temporal abstraction in reinforcement learning. Arti_cial Intelligence, pages 181{211, 1999.

[8]   Glynn Winskel. Topics in concurrency: Part ii comp. sci. lecture notes. Computer Science course at the University of Cambridge, 2002.

[9]   D. H. Ackley and M. Littman. Altruism in the evolution of communication. In Artificial Life IV: Proceedings of the International Workshop on the Synthesis and Simulation of Living Systems, third edition. MIT Press, 1994.

[10]   D. Andre, F. Bennett III, and J. Koza. Discovery by genetic programming of a cellular automata rule that is better than any known rule for the majority classification problem. In Genetic Programming 1996: Proceedings of the First Annual Conference. MIT Press, 1996.

[11]   D. Andre and A. Teller. Evolving team Darwin United. In M. Asada and H. Kitano, editors, RoboCup-98: Robot Soccer World Cup II. Springer Verlag, 1999.

[12]   P. Angeline and J. Pollack. Competitive environments evolve better solutions for complex tasks. In S. Forrest, editor, Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA), pages 264–270, San Mateo, CA, 1993. Morgan Kaufmann.

[13]   T. Balch. Reward and diversity in multirobot foraging. In IJCAI-99 Workshop on Agents Learning About, From and With other Agents, 1999.

[14]   B. Banerjee, R. Mukherjee, and S. Sen. Learning mutual trust. In Working Notes of AGENTS-00 Workshop on Deception, Fraud and Trust in Agent Societies, pages 9–14, 2000.

[15]   A. Barto, R. Sutton, and C. Watkins. Learning and sequential decision making. In M. Gabriel and J. Moore, editors, Learning and computational neuroscience : foundations of adaptive networks. M.I.T. Press, Cambridge, Mass, 1990.

[16]   J. Bassett and K. De Jong. Evolving behaviors for cooperating agents. In Z. Ras, editor, Proceedings from the Twelfth International Symposium on Methodologies for Intelligent Systems, pages 157–165, Charlotte, NC., 2000. Springer-Verlag.

[17]   J. K. Bassett. A study of generalization techniques in evolutionary rule learning. Master's thesis, George Mason University, Fairfax VA, USA, 2002.

[18]   R. Beckers, O. E. Holland, and J.-L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In Artificial Life IV: Proceedings of the International Workshop on the Synthesis and Simulation of Living Systems , third edition. MIT Press, 1994.

[19]   Rohanimanesh, Khashayar, and Sridhar Mahadevan. "Learning to take concurrent actions." In Conference on Neural Information Processing Systems, pp. 1619-1626. 2003.

[20]   Y. Yamasaki, H. Tsuda, and R. Yokogawa, "Grasping and Transportation Control of Two-Fingered Robot Hand with a Method for Changing Control Mode based on Potion and Force Errors," Proceedings of the 9th MOVIC, No.05-15, 2005, pp.538-543 (InJapanese).

[21]   E. Durfee, V. Lesser, and D. Corkill. Coherent cooperation among communicating problem solvers. IEEE Transactions on Computers, C-36(11):1275–1291, 1987.

[22]   P. Stone and M. M. Veloso. Multiagent systems: A survey from a machine learning perspective. Autonomous Robots, 8(3):345–383, 2000.

[23]   Liviu Panait , Sean Luke, Cooperative Multi-Agent Learning: The State of the Art, Autonomous Agents and Multi-Agent Systems, v.11 n.3, p.387-434, November 2005 [doi>10.1007/s10458- 005-2631-2]

[24]   Hayes-Roth, Barbara. "Opportunistic control of action in intelligent agents." Systems, Man and Cybernetics, IEEE Transactions on 23.6 (1993): 1575-1587.

[25]   P. E. Agre and D. Chapman. "Pengi: An implementation of a theory of activity." In Proceedings of the National Conference on Artificial Intelligence, San Matao, Ca: Morgan Kaufmann, 1987

Forgetting Classification and Measurement for Decomposition-based Reinforcement Learning

Sean Mondesire[1] **and R. Paul Wiegand**[2]

[1]College of Engineering and Computer Science, University of Central Florida,Orlando, FL, USA

[2]Institute for Simulation & Training, University of Central Florida,Orlando, FL, USA

Abstract— *Forgetting is a memory phenomenon that affects knowledge used to perform behaviors and tasks. In this work, two metrics are presented to aid the diagnosis of forgetting in decomposition-based reinforcement learning systems. With these metrics, developers now have the ability to classify types of forgetting, identify when important knowledge is lost, located wasted computation effort, and verified if a task decomposition is best suited for learning a given task. Through monolithic and decomposition-based learning experiments, the validity of these metrics is examined and recommendations are made about when these metrics are most effective.*

Keywords: Forgetting metrics, reinforcement learning.

1. Introduction

This paper presents two new metrics that classify and measure certain types of *forgetting* challenges that can occur in decomposition-based reinforcement learning. The proposed metrics contribute to this learning domain by providing a method of measuring the effects of learning a new subtask on knowledge used to perform an older, previously learned subtask when forgetting occurs. The proposed metrics supplement traditional measurements since they relying solely on task and subtask performance data. Furthermore, the metrics weigh and aggregate the performances of all subtasks to make it convenient to identify and measure the type of forgetting that has taken place.

Once it occurs, it is important to classify and measure forgetting to identify inefficiencies in the learning system and the learning process it employs. Most importantly, because forgotten knowledge can affect the ability to perform a task, the proposed metrics determine if forgotten knowledge has a positive, negative, or neutral effect on performance. Next, the metrics can be used to identify subtask performance dips or stagnation in the context of all of the subtasks. Affected subtask performance can then be targeted for subtask retraining with the goal of recovering lost knowledge and proficiency. This simple analysis is made possible through the easily produced quantifiable values the metrics generate.

The two presented metrics are purely performance-based and can be applied to a wide variety of problems and underlying metrics; however, unlike traditional performance functions, these metrics use performance histories. This use of subtask performance history provides additional details of changes over time that occur at low-decomposition levels that can easily be missed by more immediate performance

functions. Secondly, these proposed metrics aggregate all of the subtasks of a task decomposition to determine the significance of a change in knowledge prior to and post-forgetting. These aggregations may differ from a performance function used in a reinforcement learner's evaluation phase but the proposed metrics are used to primarily measure the difference in performance of altered, forgotten knowledge.

For example, in the decomposition-based paradigm of layered learning [1], the learning system learns a complex task by decomposing the task into multiple subtasks and sequentially learns to perform each subtask one-by-one. If the complex task is to play the game of soccer, a simple task decomposition is to first learn to pass the ball, receive a pass, and then to play a full game of soccer. The problem arises when the learner transitions from one subtask to another. As the soccer-playing system learns the third subtask of playing the full game of soccer, the ability to perform the subtask of passing the ball may be affected. In particular, the learner may forget some of the knowledge required to pass the ball as effectively as it did before it started to learn to play the full game.

In general, the two new metrics are intended to be deployed when it is suspected that performance of a sequential decomposition-based reinforcement learning system is affected by forgetting as the system iterates through a task decomposition. The goal is that these metrics will be able to identify periods in the learning process where forgetting causes performance to stagnate or suffer and to quantify the effect forgetting has on the system. With these tools, system developers will be able to analyze task decompositions, performance functions, and subtask transitions to identify inefficiencies and optimize their learning system designs.

To validate the metrics, we construct artificial problems that induce such forgetting challenges. These problems are not reinforcement learning problems but they abstract key aspects common to learning problems that are affected by forgetting. From our experiments, we compare the results of using the two forgetting metrics and make recommendations about when they might be most effectively used.

2. Background

In natural science, knowledge is information used to make a decision, perform a behavior, or have familiarity on a subject. Similarly, in machine learning, knowledge is the basis of all decisions. In this science, knowledge is represented by stimulus-response pairs that determine the conditions certain actions will be performed by the system.

Because acquired knowledge determines how well a behavior is performed, knowledge can be classified based on value. *Beneficial knowledge* leads a system towards optimal performance of a behavior. *Disadvantageous knowledge* is stimulus-response mapping that hinders behavior performance. *Unhelpful knowledge* is information that can be removed from the system with no positive or negative effect on performance.

According to Markovitch and Scott [2], the value of knowledge has several factors, including relevance, correctness, memory requirements, and influence on search time. Also, the importance of knowledge is dependent on other knowledge in the system. These five factors influence to which category an item of knowledge belongs.

Forgetting is the loss or modification of knowledge from a system's memory that may affect the performance of a behavior. Because memory is limited in machine learning, forgetting is an important mechanism that reconciles the storage of new information with old, acquired knowledge.

The act of forgetting can be caused by an explicitly invoked mechanism or an implicit side-effect of learning. Explicit forgetting mechanisms purposely drive out targeted acquired knowledge. A deletion strategy is an example of an explicit forgetting mechanism that removes knowledge from a system to increase behavioral performance or to free memory for new knowledge [3]. Markovitch and Scott's [2] randomly deleting knowledge items and Koychev's [4] and Nakayama and Yoshii's [5] time-based forgetting are examples of deletion strategies that have increased behavioral performance. Both strategies explicitly select and remove knowledge from a system's *policy* (the collection of a system's knowledge) with the goal of increasing behavioral performance.

Implicit forgetting, or *concept drift*, occurs when knowledge is lost as a consequence of learning [6]. Concept drift can happen when knowledge becomes outdated, inadequate, or loses performance due to a changing behavioral requirement. For example, in recommender systems, drift occurs when a user's preferences change, affecting the system's ability to identify accurate recommendations with outdated knowledge. A second implicit forgetting example is when a system's old knowledge or access to old knowledge is modified with the acquisition of new knowledge. The same as explicit forgetting, the unintended change to old knowledge can affect behavioral performance observed prior to new knowledge acquisition.

Similar to the classification of knowledge, forgetting can be classified as positive, negative, or neutral when based on performance. *Positive forgetting* is the loss or modification of knowledge that occurs when performance is increased compared to performance prior to the forgetting. *Negative forgetting* occurs when performance decreases when something is forgotten. *Neutral forgetting* is when knowledge is lost or modified and no change in performance takes place. This forgetting classification is directly related to knowledge being beneficial, unhelpful, or disadvantageous. By these definitions, if unhelpful knowledge is forgotten, then neutral forgetting occurs; if beneficial knowledge is forgotten then negative forgetting occurs; if negative knowledge is lost, positive forgetting occurs.

Performance change is not always caused by the loss of knowledge and performance is not always an accurate classifier for forgetting. Randomness, dynamic environment, and other external factors can affect the performance of a task. Additionally, depending on the system, the task, and its environment, beneficial and disadvantageous knowledge may be removed from a policy and not be reflected in a performance. With these points, classifying forgetting solely based on performance is best suited for instances of when factors, such as randomness and the environment, are controllable, task conditions are repeatable, and policy change has the greatest impact on performance.

The ability to classify and measure the magnitude of forgetting based on performance is important to a learning system because, under the right conditions, it can identify when beneficial knowledge has been lost, locate wasted computation effort, and verify if a learning approach is best suited for a task. From the developer's point-of-view, these unfruitful periods of learning can be examined to understand why learning did not improve and possibly be removed in future learning events to save on learning computation effort. Finally, with a quantifiable way to determine the magnitude of forgetting, the amount of forgetting can be compared between two approaches to determine which approach suffered or benefited the most from knowledge lost or modification. With these reasons to classify and measure forgetting, there are motivations for a process that quantifies forgetting.

2.1 Reinforcement Learning Systems

Although there are many different types of *reinforcement learning* (*RL*) approaches, this paper is concerned with *direct policy search* [8] in *evolutionary algorithms* (*EAs*), *genetic algorithms* (*GAs*), and primarily, *decomposition-based reinforcement learning*. EAs and GAs are RL algorithms that are based on ideas of evolution, genetics, and survival of the fittest. Here, the learner undergoes an evolution process, typically through the genetic operators of mutation and crossover, to modify the learner's representation. Through the cycle of selection, reproduction, modification, and evaluation, the system reinforces positive performance changes as the learner learns a task. More detailed information on EAs and GAs can be found in Holland's [9] and DeJong's [10] work on evolutionary learning systems.

Several studies show that decomposition-based learning can outperform monolithic learning approaches that learn to perform complex tasks all at once, including Jackson and Gibbon's [11] and Hsu and Gustafson's [12], [13] work. Stone and Veloso's [1] layered learning is an example of a decomposition-based reinforcement learning approach, as well as hierarchical reinforcement learning approaches *feudal reinforcement learning* [14] and *hierarchical abstract machines* (*HAMS*) [15]. In these approaches, a hierarchy of decomposition is used to guide the learning system to different abstractions of the overall task.

These decomposition-based approaches are exposed to forgetting valuable knowledge as the learner switches its subtask focus. The subtask transition can cause knowledge used for the new subtask to overwrite knowledge used for an older, previously learned subtask. A metric is needed to identify if this modification of knowledge is positive, negative, or neutral to performing the overall, complex task. In addition, the metric should be able to determine the magnitude of what is forgotten to indicate the severity of what has been lost.

2.2 Related Performance Metrics

The simplest policy comparison measures are performance ratio and difference, which directly compare a policy's (P) performance with the policy after some knowledge has been forgotten (P'). *Performance ratio* (*PR*) is the quotient of two performances. *Performance difference* (*PD*), $f(x)$, is the difference between the post-forgetting policy's performance $g(P')$ and the pre-forgetting policy performance $g(P)$. If $f(x)$ is positive, positive forgetting has taken place; if $f(x)$ is negative, negative forgetting has occurred; if $f(x)$ is zero, neutral forgetting has happened because there was no change in performance, although the policy has been modified. PD is defined in Equation 1.

$$f(x) = g(P') - g(P) \qquad (1)$$

Similar to PD, Markovitch's and Scott's [2] economics of learning measures the value of knowledge. In their approach, the payoff of learning is measured, where payoff can be positive, negative, or neutral and indicates the affect missing knowledge has on a policy. Payoff is calculated by taking the difference between two benefits, which are two separate policies, solving the same task. A benefit for one policy is the difference between the quality of the solution and the cost of solving the problem.

In Gorski and Laird's [16] work on transfer learning metrics, *transfer ratio*, *transfer regret*, *calibrated transfer ratio* (*CTR*), and *average relative reduction* (*ARR*) are examined for validity in comparing learning performances. These metrics, based on overall task performance, determine if an experimental policy, one that has learned a new behavior from an old behavior, outperforms a controlled policy.

The simplest of the four performance-change metrics in Gorski and Laird's work and the most similar to PR is transfer ratio. *Transfer ratio*, used by Morrison *et al.* [17] for performance comparison, is the ratio of the area under the experimental policy's performance from time 0 to time t over the area of the control's learning curve in the same time range. Unlike simple PR, transfer ratio considers the entire learning curve of the two compared policies.

Although the metrics above do not form a comprehensive list of all performance measures in transfer learning, they represent common methods of comparing policy performance change. An overlooked issue with such performance measurements is that they do not capture an entire policy change that occurs in decomposition-based approaches; instead, they capture the difference of only a single performance criterion: the overall task the policy tries to solve.

In decomposition-based approaches, a task is decomposed into subtasks the policy must learn to perform to solve the overall task. Although these single criterion measurements are adequate for monolithic learning, the learning of a task without decomposition, they lack the ability to analyze the changes in subtask performance caused by a policy modification in decomposition-based approaches. Tracking forgotten subtask-specific knowledge is significant because it can identify instances of when beneficial knowledge is lost and when learning at the subtask level stagnates or declines. The significance is heightened when subtask performance is a necessary component to the overall task but the evaluation of the task does not explicitly measure proficiency of subtasks. In this case, a single criterion-based forgetting metric only considers overall task performance changes and can overlook the loss of beneficial knowledge or the acquisition of negative knowledge used by important subtasks. Additionally, measuring performance changes only at the task level makes it difficult to determine the magnitude the lost knowledge has on task performance. Again, if subtask proficiency is an integral part of task performance, then measuring the impact of loss can be inaccurate if the task evaluation does not explicitly consider the importance of each subtask.

In this work, two metrics are introduced that classify which type of forgetting occurs at any point of the learning process and quantifies the magnitude of that diagnosed classification. Furthermore, the proposed forgetting measurements can be used by a range of existing monolithic- and decomposition-based approaches to identify performance changes due to forgetting. The main distinction of the metrics is their increased level of fidelity because of their consideration of subtask performance changes. Through experiments, we determine the effectiveness of each method and make recommendation under which conditions these proposed metrics are best suited for measuring forgetting.

3. Forgetting Metrics for Decomposition-based Learning

Both of the proposed forgetting metrics are based on the performance changes a policy experiences while acquiring new and forgetting old knowledge. The metrics compare performance of each subtask with its corresponding best performance to determine what effect the policy change has on each component of the task being learned. Although these metrics are not intended to be used to identify occurrences of forgetting, they are to be used when forgetting is the cause of policy and performance changes and are designed to classify and measure the type of forgetting that has occurred.

The difference between these proposed metrics and the others mentioned is that the proposed metrics explicitly factor in changes to all subtask performances instead of only the performance of the one, overall task. By considering subtask performance changes, each metric serves as an indicator of

when a policy loses or gains performance for individual sub-tasks. With that said, the proposed metrics are influenced by Markovitchs and Scott's measure of the value of knowledge and the metrics examined by Gorski and Laird.

Before describing the proposed metrics, a few definitions must be made that are common for each method: a *policy*, a collection of knowledge from the knowledge set, is modified at time-step t. At time-step t, the policy's performance of each subtask is retrieved from the function $p(s_i, t)$, where s_i is a subtask in the set of all subtasks S used to perform task T. In addition, function p returns the performance measure of task T at time-step t with the parameters of $p(T, t)$. Function p returns the real value ratio of performance to the optimum and is bounded to inclusively range from 0 to 1, where 1 represents optimal task or subtask performance and 0 represents the converse.

For each subtask s_i, there is a corresponding weight w_i. Also, there exists a weight for the task, w_t. Each weight is a real value inclusively ranging from 0 to 1, where the sum of all subtask and task weights is equal to 1. The weights determine the importance each subtask has on task performance and is defined by the developer.

3.1 Direct Forgetting Metric

The first forgetting measurement, *direct forgetting metric* (*DFM*), calculates a direct difference between two policies using the weighted sum of subtask performances and is defined in Equation 2. For simplicity, we will use policies at time-steps t and t-1 as the immediate policies that will be directly compared. By calculating subtask PD from these two time-steps, $f(t)$ makes a direct comparison between a policy and its immediate change in the next time-steps to determine the significance of the knowledge that was lost or modified.

$$f(t) = \sum_{i=1}^{|S|} w_i \left(p\left(s_i, t\right) - p\left(s_i, t-1\right) \right) \qquad (2)$$

3.2 Maximized Forgetting Metric

The second proposed forgetting measure, *maximized forgetting metric* (*MFM*), is represented as $g(t)$ and is the weighted sum of the difference of $p(s_i, t)$ and the best performance of s_i from time 0 to t-1. MFM is defined in Equation 3. This measure utilizes the *max* function that retrieves the best performance of policy P on subtask s_i up to the time t. By calculating this difference, the entire performance history of each subtask is factored into the forgetting measure.

$$g(t) = \sum_{i=1}^{|S|} w_i \left(p(s_i, t) - \max_{t' \in \{0, \ldots, t-1\}} \{p(s_i, t')\} \right) \qquad (3)$$

If $f(t)$ or $g(t)$ return a positive value, then positive forgetting has occurred in at time t. If the returned value is 0, then neutral forgetting has occurred. If $f(t)$ or $g(t)$ is less than 0, then negative forgetting has taken place. The magnitude of the occurred forgetting is represented by the returned value. For instance, if the returned value is negative, the smaller the value, the higher the negative forgetting magnitude is and denotes how much performance has suffered because of the lost or modified knowledge. Consequently, if the returned value is positive, the larger the number, the stronger performance has improved with the forgotten knowledge.

4. Experiment Setup

The proposed DFM and MFM metrics are compared to the PD metric (defined earlier) to evaluate the effectiveness of the proposed measures. Though our ultimate interest is in constructing improved methods to solve multi-agent decomposition-based reinforcement learning problems, such problems make for difficult initial study. Because of this, we construct simple Boolean-logic problems that have the properties we need to investigate our metrics. Our true problem involves a non-linear combination of two well-known Boolean problems: *LeadingOnes* (LO: the sum of the sequence of continuous ones in the prefix of the string) and *TrailingZeros* (TZ: the sum of the sequence of continuous zeros in the suffix of the string). For fixed-length binary strings $x \in \{0,1\}^n$, LOTZ(x) = LO(x)*TZ(x). The optimum of LO is the all one string; the optimum of TZ is the all zero string, and the optimum of the LOTZ function is a string in which the first 16 bits are 1 and the last 16 are 0. In these studies, we focus on bit strings of length 32. For comparison purposes, the result of any evaluation is always divided by the largest possible optimal value-for LOTZ, this is 256 (16 times 16).

In these experiments, each bit abstractly represents knowledge that is stored in a policy, which is represented as the bit string. Forgetting is simulated through the modification of any bit in the string and may affect performance of the task or subtask the policy is learning to solve. The bit string learns by toggling bits and is evaluated based on the task or subtask it is currently learning.

The Boolean-logic problem was chosen because it is a simple problem that allows for easier study of the effects of forgetting. Modifications to the policy is instantly recognizable and measurable because the policy is represented by a bit string. Secondly, LOTZ can be conveniently learned with a monolithic or decomposition-based approach. Examining the metrics on the two separate approaches directly allows a conclusion to be made if the new metrics satisfy the goal of measuring forgetting at the subtask and task levels. Lastly, it is acknowledged that these Boolean-logic problems reside in the optimization problem domain. These optimization problems have been chosen because the performance measures used to solve these problems are used in the exact same manner as how they would be employed in an RL technique. In both cases, performance information is available after evaluation with no additional effort or data collection is needed when translating the use of these metrics from optimization to RL. The only difference is that these optimization problems make it less complicated to examine the effectiveness of the proposed metrics than traditional RL problems. Though the problems are very basic, they make it convenient to induce

forgetting on the policy and provide a clear, straight-forward way of validating the metrics.

The PD metric is used as the control because it is widely used to compare performance differences of two policies, it is simple to compute and only requires performance data in its calculation, and its outputted value is easy to decipher (positive, negative, and zero values correlate to increased, decreased, and no change in performance). Although the other metrics also measure performance change, they do not easily translate to the forgetting classifications of positive, negative, and neutral and forgetting magnitude as the PD metric does.

Each metric will be evaluated on its ability to recognize the different types of forgetting as they evaluate two different learning methods employed to solve LOTZ. Method 1 learns LOTZ monolithically and method 2 learns the task with a three-subtask decomposition. Both methods use a 1+1 EA to optimize the bit string in solving the task. The *1+1 EA* is a evolutionary algorithm technique that has a single learner representation (the parent) produce a single modified version of itself (the child) through operators during the evolution process. The child replaces the parent if it outperforms or performs at least as well as the parent, depending on the system design. Borisovsky and Eremeev [18] and Wegener and Witt [19] provide performance studies on 1+1 EAs. In the context of this work, the bit string is the parent and a modified copy of the parent is the child. The mutation operator modifies the child by flipping each of the bits with independent probability $1/n$. The child replaces the parent if its performance is at least as well as the parent's. Bit flipping simulates forgetting when previously solved positions are lost or modified.

Method 1 learns the LOTZ task monolithically by repeating the child reproduction process until the optimal bit string is generated. The method is designed to evaluate the metrics on detecting non-negative forgetting. Method 1 guarantees only neutral and positive forgetting will occur because of the use of a 1+1 EA and its policy of only keeping a policy change if a mutated string produces a no-worse solution than the existing policy. The strict practice of only accepting equal or better performing policies also assures non-negative forgetting because only one performance criterion is used for policy evaluation and no decomposition is used.

Method 2 will learn LOTZ through a sequential decomposition-based approach, similar to Stone and Veloso's *layered learning* [1]. In method 2, the bit string will first learn the subtask of *leading ones* (*LO*) across the entire bit string, then *learn trailing zeros* (*TZ*) across the entire string, and finally learn the overall task of LOTZ. For the LO subtask, performance is calculated by counting the length of the all ones prefix and dividing it by the length of the string, 32. Similarly, the TZ subtask will calculate performance by counting the length of the all zero suffix and dividing it by the length of the string, 32. Both quotients indicate how close the bit string is at solving the subtask.

All forms of forgetting occur in method 2. Method 2's policy will experience positive forgetting as it learns LO from
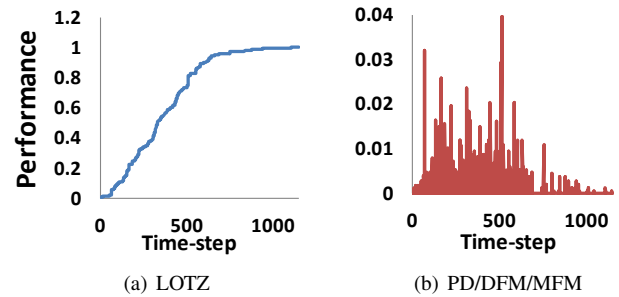
Fig. 1

<small>METHOD 1: AVERAGED LOTZ AND FORGETTING VALUES OVER TIME.</small>

the initial randomly generated string. Method 2 will then see the policy suffer from forced negative forgetting as the subtask transitions from LO to TZ because the TZ subtask is purposely designed to pressure the EA to unlearn everything from the LO subtask to maximize TZ performance. Consequently, as the performance of TZ increases, the performance of LO will decrease, causing negative forgetting. In a similar manner, the bit string can again negatively forget knowledge learned from the TZ subtask when it starts to learn LOTZ, but the negative forgetting will be at a lesser magnitude than the transition from LO to TZ because not all of the zero bits acquired for TZ will be converted into ones. Finally, whenever the policy changes but performance remains the same, neutral forgetting occurs.

The two experiment methods differ in weight assignment. Because method 1 is monolithic without any subtasks for LOTZ, the only weight value (w_t) will be set to 1. Method 2 has 3 subtasks and applies a .75 weight to the aggregate subtask of LOTZ because it is clearly the most important component to learning the overall task. The LO and TZ subtasks evenly split the remaining .25 weight, where each have .125 weight for forgetting equations DFM and MFM.

5. Results

Experiment method 1 demonstrates that PD and the two proposed forgetting metrics are equivalently well suited for classifying and measuring forgetting for the monolithic learning method. Because there is no decomposition in the monolithic method, the three forgetting metrics return the same values for each measured time-step, displayed in Figure 1 (b). Also, because the performance of the sole LOTZ task is never decreasing, the max function always returns the t -1's performance value (max returns t's performance value when $t = 0$), making all three metrics always return the same value.

The monolithic approach's use of a never decreasing performance value means only neutral and positive forgetting can be tested with the experiment setup. This never decreasing trend is displayed in Figure 1 (a), which plots the average LOTZ performance of the 10 trials over time. To test for neutral and positive forgetting, 10 independent trials of method 1

are performed and their average performance at each time-step is collected. For neutral forgetting verification, time-steps with unchanged performances for each trial and their average are compared with the metrics' returned values. With an accuracy of 100% for each trial, the metrics correctly return zero when performance does not improve. For positive forgetting, time-steps with an increase in performance are expected to result in positive forgetting values. Again, with 100% accuracy for each trial, the forgetting metrics correctly return forgetting values that correlate to positive forgetting when performance increased. For instance, at time-step 1,112, performance stagnates at .9996 for 35 time-steps. On the 36th time-step after reaching .9996, the bit string is modified to the optimal string for a performance value of 1. From this minuscule performance increase, the metrics return a small but correct .00039 positive forgetting value. From these results, it is concluded that given this experiment configuration, the proposed metrics are accurate in classifying and measuring performance-based forgetting in monolithic learning.
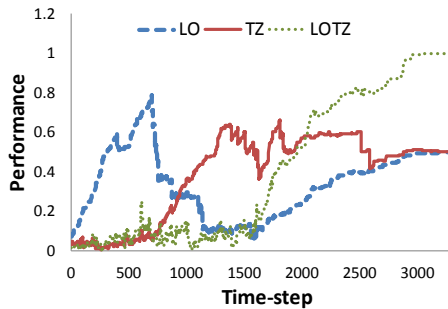
Fig. 2

METHOD 2: AVERAGED SUBTASK PERFORMANCE OVER TIME.

To validate the proposed metrics in a decomposition-based approach, method 2 collects and averages each subtask's performance per elapsed time-step for 10 independent trials. Figure 2 displays each subtask's average performance over all observed time-steps of the 10 trials. Further, subtask performance changes are noted and compared to their corresponding forgetting metrics' values. PD's averages are graphed in Figure 3 (a), DFM in Figure 3 (b), and MFM in Figure 4.

(a) PD   (b) DFM
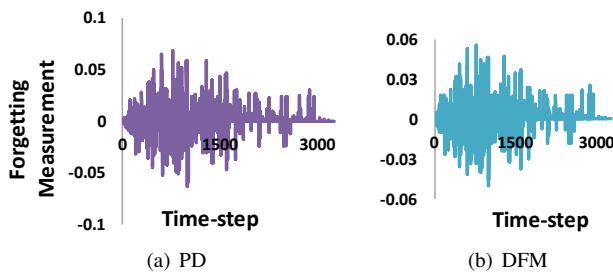
Fig. 3

METHOD 2: AVERAGED FORGETTING VALUES OVER TIME.
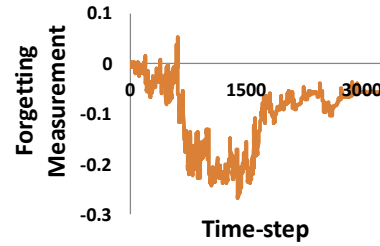
Fig. 4

METHOD 2: AVERAGED MFM VALUES OVER TIME.

By evaluating the 10 independent trials, it is expectedly observed that PD, DFM, and MFM can generate different classifications and magnitudes at the same time-step. Because PD only calculates the difference between two task performances, DFM's calculation is based on the weighted sum of subtask performances between two performances, and MFM uses the difference between each subtask performance at t and their respective maximum observed subtask performance through t-1, the forgetting calculations will differ at times.

For neutral forgetting validation, it is observed that for each of the 10 runs, PD and DFM remained unchanged when no change took place in performance. This neutral forgetting always results in PD and DFM returning 0 when subtask performance was not altered. Because MFM uses the entire history of each of the subtasks, the metric returns 0, the indicator of neutral forgetting, when the weighted aggregated value of all of the subtask is even with the best observed performance. In terms of the average of the 10 trials, MFM only detects neutral forgetting for a few instances early in the learning of LO and right before performance evaluation is switched to TZ. This neutral instance is caused by the bit string stagnating with the current best observed string while waiting to flip the correct bits for higher performance.

The ability to measure positive and negative forgetting is also validated through method 2. From the trials, PD always awards a positive value when task performance increases between two immediate time-steps and generates a negative value when there is performance decrease between t and t-1. Similarly, DFM always generates a positive value, which denotes positive forgetting, when the weighted differences in subtask performance is positive between t and t-1; negative forgetting takes place when the differences are below 0. DFM's trend is followed by MFM with the difference of comparing the best observed subtask performances with the performances at time-step t. Because the aggregate subtask of LOTZ is heavily weighted in these experiments, changes in LOTZ dominate the other subtasks in DFM and MFM calculations.

Finally, there are times where classification from the three metrics all differ. Again, these differences are caused by changes at the subtask level between PD and the proposed metrics and whether or not subtask performance at t-1 in-

cludes a best observed performance. For example, in the average of the 10 decomposition-based trials, at time-step 173, all three metrics returned different measurements. PD indicated neutral forgetting occurred with a value of 0 because LOTZ performance did not change between t and t-1. DFM indicated positive forgetting occurred with a value of 0.0016 due to a positive increase in LO performance while the other 2 subtasks maintained their values. MFM indicated negative forgetting because none of the subtask performances at t met or exceeded the observed best subtask performances. Because PD does not use the performance of each subtask, it misses subtask changes which can be responsible for performance degradation.

From methods 1's monolithic approach and 2's decomposition-based approach, we can conclude that all three metrics correctly classify forgetting at their respective levels; their differences lie in their subtask fidelity. PD is a simple metric that only considers direct changes in task performance. Although the easiest to calculate and requires the least amount of input, PD ignores changes at the subtask level. Even though it is more complex than PD and requires pre-defined subtask weights, DFM allows for direct comparison between two learning time-steps and factors in subtask changes. Finally, MFM compares each subtask performance at one time to the best observed.

6. Conclusion

Two proposed performance metrics were examined for accuracy and validity of forgetting classification and magnitude measuring. The first metric, direct forgetting metric, uses subtask performance and compares performance difference between two policies to determine the type of forgetting that has occurred. With the use of subtask performance weights and comparing performance changes between two policies, this metric is best used when a direct policy comparison is desired that compares two decomposition-based policies. The second metric, maximize forgetting metric, considers the entire history of subtask performance to determine which type of forgetting a policy has experienced. This metric is optimized to measure forgetting based on the best observed subtask performances.

As a control, both proposed metrics were compared to a performance difference measure. For monolithic learning, all three metrics are equivalent and prove accurate at classifying the type of forgetting that has occurred. Under decomposition-based learning, the performance difference measure does not factor all of the knowledge that is lost through the learning process. Instead, it ignores performance changes at the subtask level and solely relies on task performance for its calculation. This oversimplification can result in inaccuracies if subtasks are heavily weighted. On the contrary, the proposed metrics are shown to be robust enough to inform both a monolithic and decomposition-based approaches, capture subtask performance changes, and support direct and history-based calculations.

Although the metrics were not designed to detect forgetting, they are intended to be used when it is known that forgetting has occurred and is the cause of performance change in decomposition-based reinforcement learning. The metrics become valuable assets when the learning system struggles to retain important information used for solving older subtasks when new knowledge is obtained. With these metrics, developers can pinpoint times of the learning process where important knowledge is forgotten and determine its impact on performance at the task and subtask level.

References

[1] P. Stone and M. Veloso, "Layered learning," in *Proceedings of the Eleventh European Conference on Machine Learning*. Springer Verlag, 1999, pp. 369–381.

[2] S. Markovitch and P. D. Scott, "The role of forgetting in learning," in *In Proceedings of the Fifth International Conference on Machine Learning*. Morgan Kaufmann, 1988, pp. 459–465.

[3] B. Smyth and M. T. Keane, "Remembering to forget: A competence-preserving case deletion policy for case-based reasoning systems." Morgan Kaufmann, 1995, pp. 377–382.

[4] I. Koychev, "Gradual forgetting for adaptation to concept drift," in *In Proceedings of ECAI 2000 Workshop Current Issues in Spatio-Temporal Reasoning*, 2000, pp. 101–106.

[5] H. Nakayama and K. Yoshii, "Effectiveness of active forgetting in machine learning applied to financial problems," pp. 24–29, March 2002.

[6] G. Widmer and M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Mach. Learn.*, vol. 23, no. 1, pp. 69–101, Apr. 1996. [Online]. Available: http://dx.doi.org/10.1023/A:1018046501280

[7] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: a survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

[8] V. Heidrich-Meisner and C. Igel, "Evolution strategies for direct policy search," in *Proceedings of the 10th international conference on Parallel Problem Solving from Nature: PPSN X*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 428–437.

[9] J. H. Holland, *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, 1975.

[10] K. A. D. Jong, *Evolutionary computation - a unified approach*. MIT Press, 2006.

[11] D. Jackson and A. Gibbons, "Layered learning in boolean gp problems," pp. 148–159, 2007.

[12] W. H. Hsu and S. M. Gustafson, "Genetic programming for layered learning of multi-agent tasks," in *2001 Genetic and Evolutionary Computation Conference Late Breaking Papers*, E. D. Goodman, Ed., San Francisco, California, USA, 9-11 July 2001, pp. 176–182.

[13] ——, "Genetic programming and multi-agent layered learning by reinforcements," in *In Genetic and Evolutionary Computation Conference*. Morgan Kaufmann, 2002, pp. 764–771.

[14] P. Dayan and G. E. Hinton, "Feudal reinforcement learning," in *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann, 1993, pp. 271–278.

[15] R. Parr and S. Russell, "Reinforcement learning with hierarchies of machines," in *Advances in Neural Information Processing Systems 10*. MIT Press, 1998, pp. 1043–1049.

[16] N. A. Gorski and J. E. Laird, "Evaluating evaluations: A comparative study of metrics for comparing learning performances," Computer Science Department, Rutgers University, Center for Cognitive Architecture, University of Michigan, 2260 Hayward Ave, Ann Arbor, Michigan 48109-2121, Tech. Rep., 2009.

[17] C. T. Morrison, Y. han Chang, P. R. Cohen, and J. Moody, "Experimental state splitting for transfer learning," in *Proceedings of the ICML-06 Workshop on Structural Knowledge Transfer for Machine Learning*, 2006.

[18] P. A. Borisovsky and A. V. Eremeev, "A study on performance of the (1+1)-evolutionary algorithm," in *FOUNDATIONS OF GENETIC ALGORITHMS, 7*. Morgan Kaufmann, 2003, pp. 271–287.

[19] I. Wegener and C. Witt, "On the behavior of the (1+1) evolutionary algorithm on quadratic pseudo-boolean functions," 2000.

An Extraction of Influential Lightings for Illuminance Sensors and Lighting Off Mechanism in An Intelligent Lighting System

Miki Mitsunori[1], Yohei Azuma[2], and Ikegami Hisanori[2]
[1]Department of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *The authors have researched and developed an intelligent lighting system for achieving various desired brightness levels for various office workers while reducing power consumption. In an intelligent lighting system, it is necessary to dynamically estimate the influence of lighting and illuminance sensors in order to efficiently carry out an optimum lighting pattern. In some cases, however, it is not possible to accurately estimate the influence of the lighting and illuminance sensors. In this case, more time is necessary to achieve an optimal lighting pattern; therefore, there are problems that need to be resolved with regards to the feasibility of achieving the desired brightness and energy conservation. In light of this, the authors propose a method for more accurately estimating lightings that are located near illuminance sensors. Verification experiments showing the effectiveness of the proposed method verified that the lightings near illuminance sensors could be accurately extracted. Furthermore, since the lighting that has no influence can be taken into consideration, the verification experiments also verified that energy conservation can be improved as the lighting that had no influence could be switched off.*

Keywords: Lighting Control, Optimization, Office, Energy Conservation

1. Introduction

An improvement of office workers' intelligent productivity, creativity and comfortableness in offices has been focused on in recent years[1], [2]. And it is clarified in the study by Boyce, etc. that to provide illuminance most suitable for execution of work for each individual is effective from the viewpoint of improving the lighting environment[3]. To provide brightness most suitable for execution of work for each individual is easily realized with task and ambient lighting. However, ceiling lighting fixtures which provide even brightness on a floor are common in office buildings in Japan, and it is not easy to adopt task and ambient lighting. Therefore, the lighting control system to provide brightness most suitable for each office worker is necessary by using ceiling lighting fixtures.

With this background in mind, the authors proposed an intelligent lighting system that focuses on the lighting environment in order to provide a personalized illuminance for office workers while reducing power consumption[4].

In an intelligent lighting system, an optimization method is used to achieve an illuminance desired by an office worker by randomly changing the brightness (luminance) of the lighting within a range that is imperceptible to the office worker[5]. Further, regression analysis is performed on the basis of the lighting variation of the luminance as well as the variation of the illuminance detected by the illuminance sensor in the search process of the optimal lighting pattern. This analysis is used to estimate the positional relationship (degree of influence) of the lighting and the illuminance sensor. The directionality of the variation of the luminance of the lighting can be determined, and the optimal light diming pattern can therefore be efficiently achieved, as well.

In some cases, however, the influence of lighting near to the location where illuminance is measured cannot be accurately ascertained, and the lighting may be dimly lit for a moment. From the perspective of viability and energy conservation required for obtaining a desired illuminance, this phenomenon must be addressed in order to even entertain the idea of practical use of such a system.

In light of this, the authors propose a method for more accurately estimating illuminance of lighting that is near an illuminance sensor. The desired illuminance can thereby be achieved with lower power consumption even in cases where the influence of lighting near to the illuminance sensor is not accurately estimated. Furthermore, the proposed method can also be used to dim the lighting that does not influence the illuminance sensor, and it is thus possible to further enhance energy conservation by turning off such lighting. In this paper, we construct a system that incorporates the proposed method, perform verification experiments for this system in a simulated environment of an actual office, and demonstrate the effectiveness of the system.

2. Intelligent Lighting System

2.1 Construction of Intelligent Lighting System

An intelligent lighting system realizes an illuminance level desired by the user while minimizing energy consumption by changing the luminous intensity of lightings. The intelligent lighting system, as indicated in Figure 1, is composed of lights equipped with microprocessors, portable illuminance sensors, and electrical power meter, with each element connected via a network. Control devices installed in the
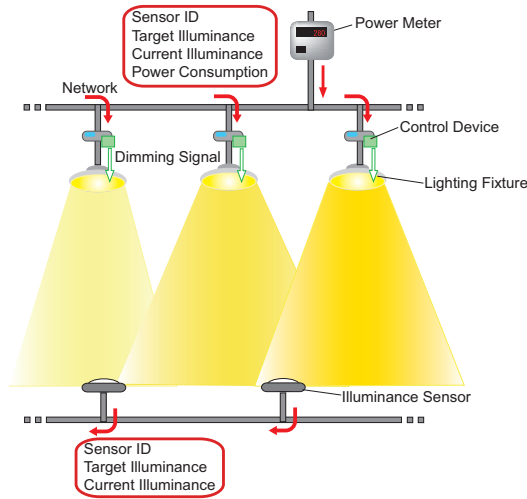
Figure 1: The construction of a intelligent lighting system

lightings continuously vary the luminance using an optimization method on the basis of the illuminance information and the power consumption information. It is thereby possible to achieve the illuminance desired by a user with low power consumption.

2.2 Control Algorithm using Regression Analysis

In Intelligent Lighting System, the algorithm where Simulated Annealing (SA) is improved for lighting control (Adaptive Neighborhood Algorithm using Regression Coefficient: ANA/RC) is used to control luminance intensity for each lighting fixture[4].

It is possible with ANA/RC to provide the target illuminance with minimum power consumption by making luminance intensity for lighting fixtures the design variable and by using the difference between the current illuminance and target illuminance as well as power consumption as objective functions. Furthermore, by learning the influence of each lighting fixture on each illuminance sensor using the regression analysis and by changing the luminance intensity depending on the results, it is possible to promptly change to the optimal luminance intensity. This algorithm is effective to solve the problem which the objective function is near monomodal function and changes in real time. The objective function is indicated in the Equation (1).

$$f \quad = \quad P + w \times \sum_{j=1}^{n} g_j \qquad (1)$$

$$P \quad = \quad \sum_{i=1}^{m} L_i$$

$$g_j \quad = \quad \left\{ \begin{array}{ll} 0 & (Ic_j - It_j) \geq 0 \\ R_j \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{array} \right.$$

$$R_j \quad = \quad \left\{ \begin{array}{ll} r_j & r_j \geq T \\ 0 & r_j < T \end{array} \right.$$

n:Number of illuminance sensors, m:Number of lighting fixtures, w:Weight, P:Power consumption, Ic:Current illminance, It:Target illuminance, L:Luminance intensity, r:Regression coefficient, T:Threshold value

As indicate in the Equation (1), the objective function f consists of power consumption P and constraint g_j. The difference between the current illuminance and target illuminance is used for the constraint g_j, and a penalty is imposed only if the target illuminance is not achieved. As a result, the objective function value largely increases as the target illuminance goes further than the current illuminance. $R_j = 0$ is multiplied if the regression coefficient is less than the threshold. With this, if the illuminance sensor with a lower regression coefficient does not achieve the target illuminance, the objective function value does not increase. Therefore, objects for optimization are successfully limited to illuminance sensors to which the lighting gives a strong influence. Furthermore, the weight w value is multiplied for constraint g_j, and it is possible to switch whether or not to prioritize the convergence to the target illuminance over minimization of power consumption by setting the weight w value.

2.3 Problem in Regression Analysis

The intelligent lighting system varies the luminance of the lightings in the process for searching for the target function show in Equation (1). At this point, the luminance of all of the lightings that are switched on is varied, and the amount of variance in illuminance detected by the illuminance sensors is thus not determined by only a small variance in luminance. Regression analysis is then carried out on the basis of the luminance variation of the lightings and the illuminance variation detected by the illuminance sensors. The variation in luminance of one lighting therefore acts as a disturbance component for another lighting. With regards to this problem, errors in the regression coefficient of the lightings and illuminance sensors can be minimized by using a sufficient number of steps. However, there is a possibility that large errors will occur in the calculated regression coefficient when there are too few steps.

In view of the forgoing, the authors carried out verification experiments in order to determine the accuracy of the regression coefficient under conditions where the number of steps is few. The verification experiment environment consisted of 15 white fluorescent lamps made by Panasonic (model FHP45EN) and four illuminance sensors installed as depicted in Figure 2. The regression coefficient of the lightings and illuminance sensors was calculated using 60 steps (approximately 120 seconds) of illuminance and luminance data. A regression coefficient that did not include
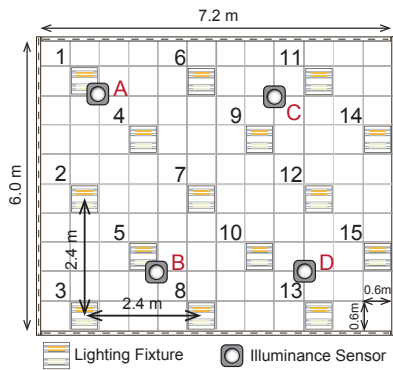
Figure 2: Experiment environment of verifying accuracy (ground plan)

Figure 3: Accuracy of the regression coefficient for each illuminance sensor (the lightings ranked the first and the second)
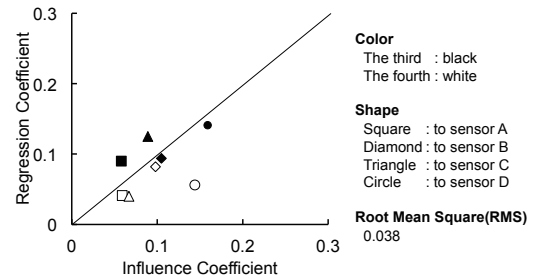
Figure 4: Accuracy of the regression coefficient for each illuminance sensor (the lightings ranked the third and the fourth)

disturbance was used as a target for comparison with the calculated regression coefficient. This value was calculated by varying the luminance of the lightings one at a time, and is considered to be the true value of the regression coefficient. The regression coefficient is defined as the influence coefficient. Table 1 shows the four lightings with the highest influence coefficient in the illuminance sensors.

Figure 3 shows the regression coefficients and influence coefficients of the first and second lightings shown in Table 1. Figure 4 shows the regression coefficients and influence coefficients of the third and fourth lightings. Table 2 shows the four lightings with the highest regression coefficients in the illuminance sensors.

As can be confirmed from Table 1 and Figure 3, the regression coefficients and influence coefficients of the first and second lightings are calculated to be nearly the same values. Further, as can be confirmed from Table 1 and Figure4, there is a great deal of error between the regression coefficients and the influence coefficients of the third and fourth lightings. The root mean square (RMS) shown in Figure 3 and 4 also makes it clear that error increases in the

sequential order of the lightings. Next, the sequential order of the two lightings that have the highest influence coefficient in illuminance sensors can be accurately determined using the regression coefficients as shown in Table 1 and Table 2. However, it is evident that the error included in the regression coefficients of the third and fourth causes the sequential order of these two lightings to be reversed. These results demonstrate that a large degree of error occurs between the influence coefficient and the regression coefficient.

It is thus possible to perform precise control according to the position of the illuminance sensors in cases where the number of steps is few because the regression coefficient is used in relation to the target coefficient shown in Equation (1) and the orientation of variation in lumination of the lightings. For example, the lightings may be turned down low even when positioned near the illuminance sensor, and likewise the lightings may be brightly lit even when positioned at some distance from the illuminance sensor.

2.4 Validation Experiment in the Actual Offices

In order to verify the practicality of the intelligent lighting system, the authors introduced this system into an actual office setting[6]. In this office setting, the desks of the office workers were in fixed locations and the illuminance sensors were also not moved. There was therefore no need to dynamically estimate the degree of influence. The issues presented in Section 2.3 were therefore addressed by calculating the

Table 1: The lighting with a large influence coefficient

	Top four lightings (No.)			
	the first	the second	the third	the fourth
Sensor A	1	4	2	6
Sensor B	5	8	7	3
Sensor C	11	9	6	14
Sensor D	13	10	12	15

Table 2: The lighting with a large regression coefficient

	Top four lightings (No.)			
	the first	the second	the third	the fourth
Sensor A	1	4	6	2
Sensor B	5	8	7	3
Sensor C	11	9	6	14
Sensor D	13	10	12	6

influence coefficient when the system was introduced and storing these values as a static database.

On the other hand, many workplaces are now adopting a non-territorial office configuration, in which there are no personal desks. In a non-territorial office configuration, office workers can freely determine where they sit, and such an office requires the illuminance sensors to move. In order to calculate the influence coefficient, the lumination of the lightings must be varied one by one, and the control algorithms of the intelligent lighting system used to simultaneously vary the lumination of all the lightings cannot be processed in parallel. Therefore, it is difficult to calculate the influence coefficient in an environment where office workers move about. Regardless, in order to achieve behavior similar to previously introduced intelligent lighting systems, it is necessary to adequately control lightings even when the regression coefficient of lightings near illuminance sensors is not accurate. In light of this, the authors propose a method whereby the lightings that influence the illuminance sensors can be more accurately extracted.

3. Extraction of Lightings That Influence Illuminance Sensors

3.1 Method for Extracting Effective Lightings Using Schematic Lighting Layout Diagram

Even in cases where the number of steps is few, it is possible to accurately determine the two lightings that exhibit the greatest influence on the illuminance sensors by using regression analysis in the control algorithm (ANA/RC) of the intelligent lighting system. The purpose of the method proposed in this study is to more accurately extract the lightings that are near the illuminance sensor by using the two lightings with the highest regression coefficient and the schematic lighting layout diagram.

Figure5 shows a conceptual diagram of the method proposed in this study. As shown in Figure 5, two lightings with a high influence (lighting 7 and lighting 10) can be identified by regression analysis for each illuminance sensor. It can be assumed that other lightings which have influence on the illuminance sensor are also located in the vicinity of these two lightings. Therefore, those lightings near the two lightings that have the highest regression coefficient can be extracted from the schematic lighting layout diagram. These lightings may be determined by calculating the linear distance between the two lightings with the highest regression coefficient and the other nearby lightings, and selecting the lightings that are near both of the top two lightings. Lighting is considered to be ạÈnearạÉ if the calculated linear distance is equal to or less than the maximum distance between adjacent lightings (2.4 m in the environment shown in Figure 2 and Figure 5). In the environment depicted in Figure 5, it is possible to extract lightings in which the linear distance is considered to be close to the top two lightings
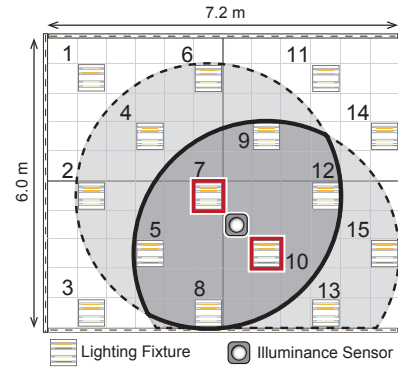
Figure 5: Concept of proposal method

(i.e., lightings 5, 8, 9, and 12). The lightings not included in the extraction results are lightings far away from either of the top two lightings, and can be considered to have no influence on the illuminance sensors.

A conventional control algorithm and the schematic lighting layout diagram are combined in order to extract three or fewer lightings based on the schematic lighting layout diagram and the placement of the two lightings with the highest regression coefficient. It is therefore conceivable that the influence of lightings can be accurately determined, using the regression coefficient. However, in cases in which partitions are disposed on the desks, lighting may be blocked even when the lightings are near the illuminance sensor. The proposed method cannot thus be used in offices in which partitions are used.

The problems described in section 2.3 occur when the regression coefficients of extracted lightings are not accurate. These problems can therefore be solved by providing the extracted lightings with a coefficient that replaces with the regression coefficient. In the control algorithm of the intelligent lighting system, the target of optimization is focused on the nearest lightings by assigning a threshold value T to the regression coefficient. The provided value must therefore be greater than or equal to the threshold value T. The influence of the extracted lightings will never be greater than that of the two lightings with the highest regression coefficient that were used in the extraction process. For this reason, these lightings are provided with a value that is between the retained regression coefficient and the threshold value T.

Furthermore, it is possible that the regression coefficient of a lighting that does not influence the illuminance sensor may exceed the threshold value T. Therefore, the lightings that are not included in the extraction results in the proposed method are provided with a value which is less than the threshold value T. It is thereby possible to search for an optimal lighting pattern that is not influenced by error in the regression coefficients.

3.2 Improving Energy Conservation Through Dimming Control of Lightings That Have No Influence On Illuminance sensors

In the intelligent lighting system, in order to address movement of illuminance sensors and to improve the safety of the office environment, the lightings that have no influence on illuminance sensors are turned down to the lowest possible illumination level. However, turning off the lightings that have no influence on illuminance sensors is an effective measure when improving energy conservation is a priority.

In the proposed method, the lightings that have an influence on illuminance sensors can be extracted, while, at the same time, the lightings that have no influence on illuminance sensors can be identified. Turning off these lightings makes it possible to retain only those lightings necessary to achieve the desired illuminance while still improving energy conservation.

On the other hand, moving illuminance sensors in locations, where lightings have been switched off, makes it impossible to achieve an appropriate illuminance. Therefore, after illuminance sensors are moved, all of the lightings that have been switched off are switched on to the lowest luminance level, and the lightings that have an influence on illuminance sensors are once again extracted, using the proposed method. Movement of the illuminance sensors is thus determined on the basis of the amount of variation in illuminance. Office workers are determined to have moved illuminance sensors when inexplicable variation in illuminance occurs in the illuminance of the lightings.

4. Verification Experiment

4.1 Overview of Verification Experiment

A verification experiment was carried out in order to demonstrate the effectiveness of the proposed method. In the experiment, fifteen lightings and two illuminance sensors were arranged as depicted in Figure 6. White fluorescent lamps made by Panasonic (FHP45EN), which are capable of being dimmed from 30 to 100%, were used as the lightings. The number of illuminance sensors was kept to a minimum in order to clearly demonstrate the turning-on of lightings in optimal locations.

In the experiment environment, an illuminance convergence experiment was carried out using a control algorithm (the proposed method) in which the proposed method was incorporated with a conventional control algorithm (conventional method: ANA/RC). The regression coefficient was calculated from 60 steps of data (approximately 120 seconds), and this regression coefficient was used to extract the lightings that influence the illuminance sensors. Additionally, in the proposed method, the lightings that have no influence on the illuminance sensors are switched off.

The lightings are initially switched on at 100% and the target illuminance of the illuminance sensors A and B are
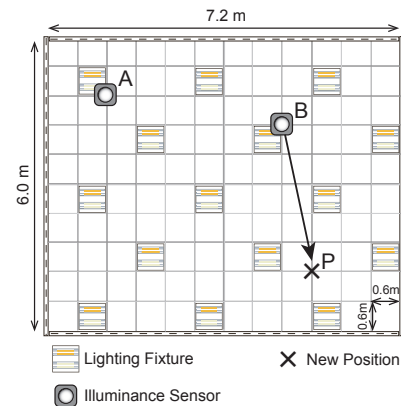
Figure 6: Experiment environment

set to 400 lx and 600 lx, respectively. Furthermore, the illumination sensor B is moved to the location P shown in Figure 6 after 300 steps. A single step takes two seconds.

4.2 Verification of Feasibility of Achieving a Desired Illuminance

Figure 7 and 8 show the history of the illuminance detected by the illuminance sensors in both the conventional method and the proposed method. The data for the conventional method only shows cases where the regression coefficient was accurately obtained by the illumination sensors.

It can be confirmed from Figure 7 and 8 that the desired illuminance was achieved. In the proposed method, the illuminance detected by the illuminance sensor B suddenly dropped in comparison to the conventional method after 300 steps had elapsed. However, this is because the illuminance sensor B had been moved to a location in which the lightings were switched off. The illuminance immediately increased after the illuminance sensor B was moved, which confirms that the movement of the illuminance sensor B was detected and that the lightings which had been switched off were then switched on. The lightings that influenced the illumination sensor were then once again estimated, and the desired illuminance was again achieved. The above results confirm that the control algorithm incorporating the proposed method has the same capability as the conventional method to achieve a desired illumination.

4.3 Verification of Extraction of Lightings That Influence the Illuminance Sensors

In order to confirm the effectiveness of extracting the lightings that influence lighting sensors, Figure 9 and Figure 10 show the states of the lightings before and after the illumination sensor is moved. These figures show the lighting states at the 250th step prior to the illumination sensor being moved and at the 550th step after the illumination sensor is moved.
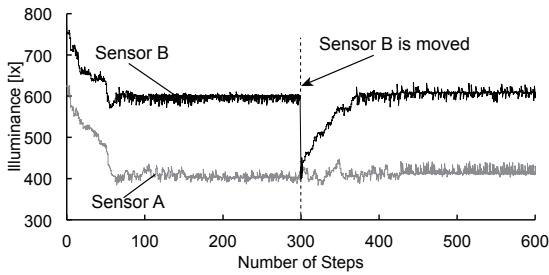
Figure 7: History of the illuminance data (conventional method)
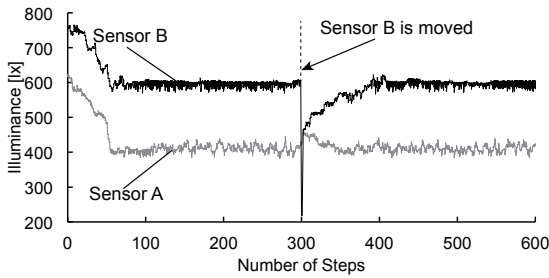
Figure 8: History of the illuminance data (proposed method)

Figure 9: Status of lightings (before moving the illuminance sensor B)
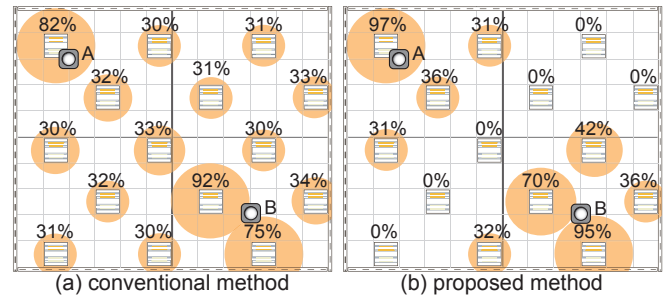
Figure 10: Status of lightings (afrer moving the illuminance sensor B)

In the conventional method shown in Figure 9-(a) and 10-(a), the closer they were to the illumination sensor, the higher level lightings were switched on at, and the lightings that were further away from the illumination sensor were switched on at the lowest possible lumination. This occurred in response to the movement of the illumination sensor, and illustrates that in this method even lightings, which have very little influence on the illumination sensor, must remain on. On the other hand, in the proposed method shown in Figure 9-(b) and 10-(b), lightings in the vicinity of the illumination sensor were switched on, lightings further away from the illuminance sensor were switched off. The lightings that were switched off are the ones not included in the extraction results of the proposed method, and are determined to have no influence on either of the illuminance sensors.

As illustrated in Figure 8, the desired illuminance can be achieved using only the switched on lightings. And this confirms that the lightings which influence the illumination sensors can be precisely extracted.

4.4 Verification of Energy Conservation

Figure 11 shows the power consumption history in the conventional method and the proposed method. As can be confirmed from Figure 11, the power consumption in the initial state of the experiment was set to 100%, and the power consumption in the proposed method can be reduced by approximately 15%, compared to the conventional method. This demonstrates that switching off the lightings which

Figure 11: History of power consumption

have no influence on the illumination sensors improves energy conservation.

5. Summary

In this paper, the authors have proposed a method for more accurately extracting lightings that influence illumination sensors in an intelligent lighting system. The experimental results confirm that the lightings which influence illumination sensors can be precisely extracted according to the position of the illumination sensors, and that the proposed method has the same capability to achieve a desired illumination as the conventional method. Furthermore, it was also confirmed that energy conservation is improved by switching off the lightings which have no influence on the illuminance sensors. From this, the proposed method is expected to provide the same behavior as intelligent lighting

systems that have previously been introduced into actual offices, even in office environments where office worker movement is unpredictable.

The authors will be introducing an intelligent lighting system that incorporates the proposed method at the Tokyu Corporation's RISE Office in August of 2012, and will carry out detailed verification experiments.

References

[1] O. Seppanen, and W. J. Fisk, A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control, Proceedings of ASHRAE, 2005

[2] M. J. Mendell, and G. A. Health, Do indoor pollutants and thermal conditions in schools influence student performance? A critical review of the literature, Indoor Air, Vol. 15, No.1, pp.27-52, 2005

[3] P. R. Boyce, N. H. Eklund, and S. N. Simpson, Individual Lighting Control, Task Performance, Mood and Illuminance JOURNAL of the Illuminating Engineering Society, pp.131-142, 2000

[4] S. Tanaka, M. Miki, T. Hiroyasu, and M. Yoshikata, An Evolutional Opti- mization Algorithm to Provide Individual Illuminance in Work-places, Proc IEEE Int Conf Syst Man Cybern, Vol.2, pp.941-947, 2009

[5] T.Shikakura, H.Morikawa, and Y.Nakamura, Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment, Journal of the Illuminating Engineering Institute of Japan, Vol.85 5, pp.346-351, 2001

[6] M. Miki, Y. Azuma, K. Ono, T. Yoshii, M. Suzuki, and K. Yoshida, Construction of Intelligent Lighting System Using In-office Frames, IEEE CIS, pp.24-29, 2012

Computational Methods for Determining the Similarity between Ancient Greek Manuscripts

Eddie Dunn[1]**, Curry Guinn**[1]**, and George Zervos**[2]
[1]Department of Computer Science, University of North Carolina Wilmington, United States
[2]Department of Philosophy and Religion, University of North Carolina Wilmington, United States

Abstract - *This paper describes research applying computational document classification methods to the domain of biblical paleography. Ancient manuscripts were preserved by the laborious process of hand-copying from prior versions. The scribes often made changes in spelling, word usage and syntax. Further, scribes might delete or alter passages that reflect a theology or understanding different from the scribe's contemporary view. Additional material not found in the prior version(s) might be inserted, perhaps combining information from other sources. In this paper, we examine over a hundred different versions of a single apocryphal gospel, the Protoevangelium of James, in order to group the documents into families of related documents in order to better understand the history of the document and how it evolved over time. This research uses the computational techniques of k-means analysis, hierarchical clustering, and correspondence analysis to find similarities and differences between documents. These results are then compared to the work of Daniels and Zervos, scholars in the field of biblical paleography who have studied this gospel.*

Keywords: Natural language processing, machine learning, clustering.

1   Introduction

Paleography is the study of ancient writing including deciphering, understanding, and dating manuscripts. Prior to the printing press, documents were preserved by hand-copying existing sources. In the ancient world, because spelling, vocabulary, and syntax were not standardized and changed over time, scribes often made modifications in order to make the text more accessible to current readers. If multiple prior versions existed, a scribe might decide to merge content from those documents or choose content from one source over others. Further, because of theological issues, a scribe might decide to delete or amend passages that seemed to contradict the scribe's current theological understanding. Passages might also be inserted from other (perhaps unrelated) sources in order to introduce or reinforce a particular theological point. As a result of this process, an ancient biblical manuscript like Genesis or The Gospel of Mark exists today in hundreds of different versions. Which version is the "correct" version? Which version most closely resembles the earliest (perhaps no longer extant) version? Can we tell which documents were the source documents for later versions? Can we trace the evolution of a document, and thus see the influences of evolving linguistics and theologies?

In this paper, we will explore techniques that have been used in authorship attribution, document classification, and data visualization in order to explore these questions.   We will analyze a collection of documents which are all versions of the *Protoevangelium of James*, an apocryphal manuscript whose original dates may be from 200 CE or earlier.   These documents have been studied by biblical paleographers, B. Daniels [1] and George Zervos [9], and we will compare the results using these computational techniques with these researchers' prior analyses.   The goal of this study is to 1) corroborate past results, 2) discover new connections between documents, and 3) suggest to paleographers particular features or passages that deserve more exploration.

2   Textual criticism and the *Protoevangelium of James*

Textual criticism is the area that aspires to remove errors (whether intentional or unintentional) in an attempt at coming as close to creating the "original" or source documents(s) as possible. The hallmark of this type of work is a critical apparatus to show variant readings alongside a primary text (also called a base text).

The *Protoevangelium of James* (PJ) most likely dates back to the middle to later part of the 2nd Century CE [10]. This document has been known by several names. In the earlier years of its life it was likely called Book of James as it is referred to in the writings of Origen who died in the middle of the 3rd Century CE [2]. As its most commonly known name in the literature today implies, proto-gospel means just that -- it is a story before the gospels or life of Jesus. It seems to have been composed largely in reaction to the accusations by contemporary critics that were assaulting the burgeoning religion on the grounds that the parents of its messiah were commoners. The writer of this document portrays Joseph as a rich building contractor and Mary as herself being immaculately conceived and brought up (with her chastity protected) as a revered temple virgin perhaps in direct response to these accusations.

Figure 1: A scanned page of the Protoevangelium of James from Bodmer V, a, dated c. 200.

This document is also intriguing as the oldest existing manuscript is complete and it is significantly different from its closet contemporary versions as well as the majority of the later surviving examples (Figure 1). PJ also enjoys the luxury of, while not being part of the canon, being widely copied and distributed throughout the ancient world, especially in those eastern traditions with highly developed Mariological themes. Mariology is, as its name implies, the study of Mary the mother of Jesus. This term has a much more profound meaning to the traditions that evolved in the eastern world. In fact, there are Eastern Orthodox feast days established based on information in this document. While all of the documents we will be performing computations on are in

Greek, there are surviving copies of this document in many languages including Coptic, Syriac, Ethiopic, Armenian, Georgian, and Slavonic. There is also a scholarly notion that an Arabic copy might have "influenced Qur'anic and later Islamic understandings of the place of Mary in the Christian tradition" [5], yet another way this text has impacted western religion.

For this study, the focus is on two separate collections of this gospel. These collections that are the basis for the dataset come from the dissertations from Duke University of Daniels (BD) [1] and Zervos (GZ) [9]. These collections are both presented in their own critical apparatus. A critical apparatus in this context is an accepted way of showing how different copies of the same documents vary (called variant readings). There are over 167 extant Greek versions of the PJ. Scholars have found that the earliest copy (from the Bodmer V collection [8]) is decidedly different from the base text used by Daniels and the base text used by GZ.

3 Authorship attribution and document classification techniques

3.1   Authorship attribution techniques

While this research does not attempt to identify particular authors, the techniques employed in authorship attribution are relevant as an introduction. While there were previous attempts dating back to the 19th century at using statistical measures in attributing authorship, it was not until the publication of *Inference and Disputed Authorship: The Federalist* by Mosteller and Wallace in 1964 that this area of "non-traditional" authorship attribution study gained widespread attention [5]. Previous work had attempted to use features such as average sentence length and rate of use of articles and pronouns. They found that while the rates of use in the case of some words such as "the" did not vary in a statistically significant manner from author to author, the use of what they refer to as connector words, such as "upon", can vary by as much as 3 standard deviations. Mosteller and Wallace used such features as word counts and rate of use of specific, non-article or pronoun words. By examining the distributions of individual words it was discovered that some word rates were best described by a Poisson distribution and others were better approximated with a negative binomial distribution. Bayesian inference was then applied using the probabilities calculated using the appropriate distribution. Their analysis was ultimately to come down on the side of supporting the historical notion that Madison was likely the author of the 12 then-disputed Federalist papers. Their study also outlines a basic work flow of technique application that is still followed.

3.2   Document classification techniques

More recently, with the profusion of massive amounts of textual data via the internet, document classification techniques have been used to compute the similarity between documents. The core of these techniques relies on using term frequency (TF) and inverse document frequency (IDF).   Term frequency can be computed as simply taken a count of a term within a document. To prevent a bias towards longer documents, this value may be normalized by a variety of techniques. In this study, we normalize term frequency by dividing by the maximum frequency of any word in the set of documents.   f(t, d) is the frequency of a word in a document"

$$\text{tf}(t,d) = \frac{\text{f}(t,d)}{\max\{\text{f}(w,d) : w \in d\}}$$

Inverse document frequency weights each feature in inverse proportion to its relative occurrence, thus giving infrequently used words higher importance. We employ the standard IDF formula:

$$\text{idf}(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

where t is the term or word, D is the set of documents, $|D|$ is the number of documents, and the denominator is the number of documents where the term t appears.
These two measures can them be combined to compute the relative importance a word is to a document in a measure known as the term frequency-inverse document frequency (tf-idf):

$$\text{tfidf}(t,d,D) = \text{tf}(t,d) \times \text{idf}(t,D)$$

For each feature selected (and as described below in our Research Experiment we will use features other than complete words), we compute a log normalized vector in multi-dimensional space of TF-IDF values.

3.3   Machine learning techniques

We then employ a number of well-established techniques in machine learning/document classification such as k-means analysis, hierarchical clustering, as well as DCA correspondence analysis, a technique often used in ecologists in the study of populations [6]. For all of these techniques, we use the statistical programming package, R, which contains libraries to support all of these analyses plus corresponding visualization tools.

3.4   Similar research applied to ancient texts

Research has been done by Finney [3] in the study of ancient manuscripts where he employs similar techniques in his analysis of ancient documents. His work focuses on a difficult (and different) problem in comparing

different versions of the same document: namely, automatic alignment of the texts.   The alignment problem is also commonly encountered in machine translation.

4   Research Experiment

4.1   Data set

All of the versions of the Protoevangelium of James (PJ) used in this study are in Greek. Our analyses examine 135 documents: 89 manuscripts analyzed by Daniels [1], 45 by Zervos [9], and Bodmer V (the oldest known version of PJ) [8]. Using OCR-software (Read Iris Pro), the original documents are scanned and converted to UTF-8 character codes.   Then each document was put into an HTML-like document format for review by human readers where proofreading was done in multiple passes by multiple persons to better ensure data accuracy and integrity.

Our experiments examined the full set of documents (135) as well as a subset of the oldest 32 manuscripts (those definitively dated before 1100 CE).

4.2   Feature selection

This research uses a composite feature space consisting of unigrams (single Greek words), bigrams (neighboring pairs of words), and character sequence n-grams of length 2, 3, 4, and 5.   The reason for the inclusion of character n-grams was to capture variations in letter patterns and spelling that occurred in the millennia from the time of the first document to the most recent documents in the collection.   Obviously, there will be overlap in the feature set (as some words are 5 characters or less); however, an analysis conducted by removing some of those character n-gram features produced slightly worse results (not included in this article).   The feature space is very large, over 85,000 unique tokens. Fortunately, the techniques employed all typically work very well with large feature spaces.

4.3   Analysis techniques

Using the "vegan" library within R [6], we computed several analyses:
- Sorensen (Bray/Curtis) similarity index cluster analysis
- Detrended correspondence analysis (DCA)
- Nonmetric Multidimensional Scaling (NMDS)
- Canonical correspondence analysis (CCA)
- K-means clustering

for each document and for each chapter within the document.

5   Results

Using the ordination plotting function (ordiplot in R), we can visualize the results of DCA, CCA and NMDS

analyses. For brevity, we present only the ordiplot from the DCA in this paper. For the full set, the DCA ordination plot is presented in Figure 2. For the subset including only the oldest 32 manuscripts, refer to Figure 3. Note that the document numbers used are the ones given by BD and GZ.

Cluster analysis visualization using dendrogram (tree) is also an intuitive appealing technique for examining differences and simulations. Using Bray-Curtis and the hclust function within R we obtained the results for the complete set (Figure 4) and the oldest 32 documents (Figure 5).

5.1 Comparison with Daniels and Zervos

In their respective PhD dissertations, Daniels [1] and Zervos [9] analyze the documents based on their own observations focusing on the inclusion or absence of various passages. Further, they made use of where the documents were located or found. Several subsets of documents were found at particular monasteries. Often, multiple versions of the document found at one monastery were found to be extremely similar, presumably because these documents were copies of each other and earlier documents. The computational analysis we performed does not take into account any of this information.

One group of documents is described as the largest by both BD and GZ consisting of 003, 005, 103, 115, 118, 201, 204, 206, 214, 502, and 609. The hierarchical plot (Figure 4) shows all of these manuscripts as being in the same cluster. This corresponds to groups E and G from the DCA groupings (Figure 2). GZ suggests that 612 and 409 might also be close, and we do place them in the same group in the hierarchical clustering as well as being in Group C on the DCA groupings (Figure 2). However, it is not near the rest of the mentioned manuscripts. This is intriguing and should be examined with the knowledge of the Greek language.

Another family widely agreed upon is the one made up from 112, 208, 212, 402, 407, 511, 616, 702, 705, 709, and 901. All with the exception of 702 and 709 came from the St. Panteleimonus monastery in Athens (so we identify this group by the same name). It is interesting to note that GZ specifies that there are two sub-groups in this family consisting of 511, 702, and 709 that follow 212 and 616, 705, 901 that follow 208,402. The hierarchical plot confirms all of this information (Figure 4). Also notice this group corresponds with Group D in our DCA groupings (Figure 2).

The group 601, 606 is also highlighted. Document 601 has in its sub-group 512, 615, 619 and 606 with sub-group 617, 703, 707, 803, 805, and 902. Our plot also confirms these observances (see DCA Group I, Figure 2).

In looking at the tree it seems proper to place the 601, 606 group and its associated documents into the Panteleimon family. It is also worth noting that 621, 631, and 704 (Group H) are mentioned by GZ as being in this group as well. It was found that while these three documents were indeed found to be similar, they were placed a good distance from the rest of its other neighbors and should be examined by experts for further analysis.

Next we examine the group from the monastery of Vatopedi on Mt. Athos. This group consists of 111, 218, 501, 513, 801, and 802. Again this is all confirmed in both the hierarchical (Figure 4) and DCA (Figure 2) plots where it shows up as group E. The group that is now being called the Jerusalem group consists of 202, 508, 603, 622, and 708. These are lumped in with our DCA Group A (Figure 2). As an aside, GZ notes the similarity of 509 and 604 and also 210 and 220. This information is also confirmed in our plots.

5.2 Chapter by chapter results

The results of the full document analysis while providing a great deal of information and striking results also shows some confusion with respect to some documents. This is especially clear with the DCA Only Old plot. There are two dynamics that contribute to this effect: The first is that the letter groupings were established from the full set which includes the very tightly grouped but also very different traditions that do not seem to be present in the earliest documents in our set. The second reason is that a more detailed examination of the chapters shows that there is a great deal of variation contained within each manuscript in certain sections versus others. This is where visualization of the correlation matrices (corrplot in R) truly shines. It provides a way to see how each chapter breaks down, and that indeed we have situations where in one chapter the scribe is using content from one tradition and then another in different sections. Figure 6 presents the correlation plots for each chapter of the oldest 32 documents. Clearly, there are families that emerge per chapter that are not as apparent when examining the entire document as a whole.

6 Conclusion and future work

The identification of "families" of related copies of the same base document has traditionally required laborious and detailed study of the documents including some knowledge of the physical history of the documents. The computational techniques described in this paper produced results that were remarkably similar to scholars' analyses. What makes this result particularly exciting to the paleographers studying this document is that there are scores and perhaps hundreds

more copies of the *Protoevangelium of James* that have not been carefully analyzed yet. These algorithms can automatically suggest which documents are related and which are dissimilar. Further, these algorithms can actually point to which features are most relevant for distinguishing the documents. Such tools will be invaluable to researchers as they incorporate new versions of the documents into their study.

While this study has focused on the PJ document, there are no limitations to language or document set. This methodology can be used to explore any collection of ancient texts to suggest document family histories.

7   References

[1] Daniels, B. (1956) *The Greek Manuscript Tradition of Protoevangelium Jacobi*, Unpublished PhD Dissertation, Duke University Durham, NC.

[2] Ehrman, B and Plese, Z. (2011) *The Apocryphal Gospels Texts and Translation*s, Oxford University Press, New York, NY.

[3] Finney, T. (2012) How to Discover Text Groups, http://www.tfinney.net/Groups/index.xhtml.

[4] Foster, P (2009) The *Apocryphal Gospels: A very short introductio*n, Oxford University Press, New York, NY.

[5] Mosteller, F & Wallace, D. (1964). *Inference and disputed authorship: The Federalist*. Addison Wesley, Boston, MA.

[6] Oksanen, J. (2013) *Multivariate Analysis of Ecological Communities in R: vegan tutorial*, http://cc.oulu.fi/~jarioksa/opetus/metodi/vegantutor.pdf.

[7] Schaps, D. (2011) *Handbook for Classical Research*, Routledge, New York, NY.

[8] Testuz, M. (1958) *Papyrus Bodmer V: Nativite de Marie*, Geneva: Bibliotheca Bodmeriana.

[9] Zervos, G. T. (1986) *Prolegomena to a Critical Edition of the Genesis Maria (Protoevangelium Jacobi)*, Unpublished PhD Dissertation, Duke University Durham, NC.

[10] Zervos, G. (1994) – Dating the Protoevangelium of James: The Justin Martyr Connection – *SBLSP*, pp. 415-34.
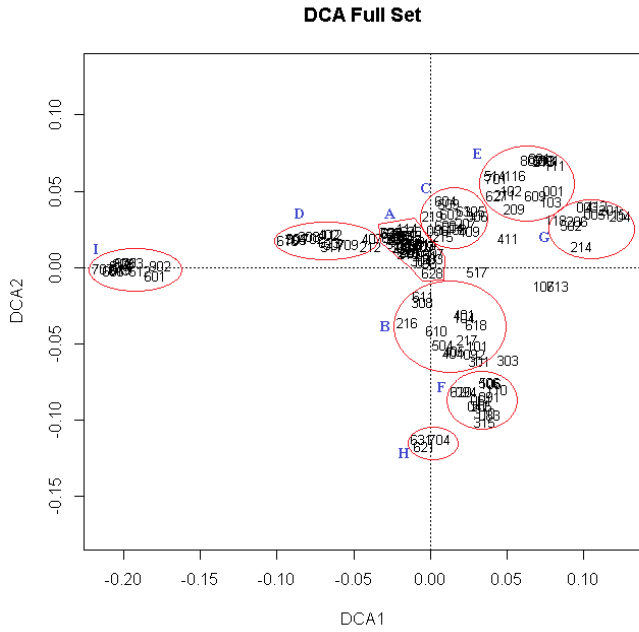
Figure 2 Detrended Correspondence Analysis (DCA) plot of the full data set
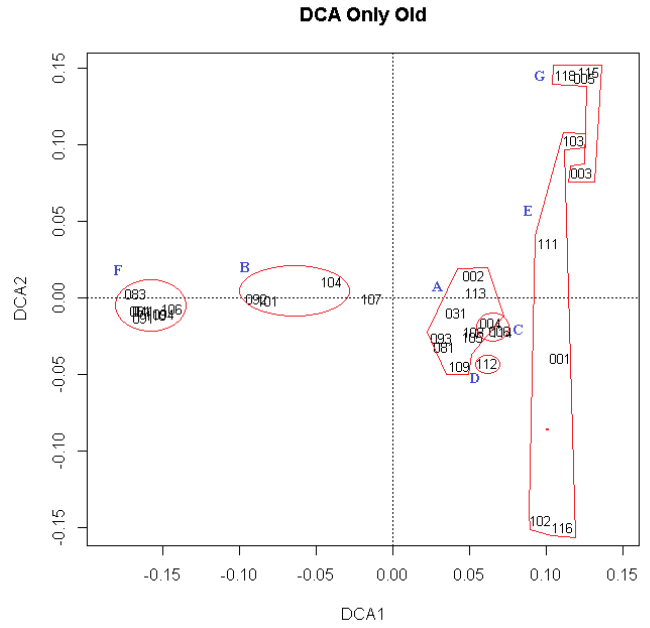
Figure 3 DCA plot of the subset of 32 oldest manuscripts
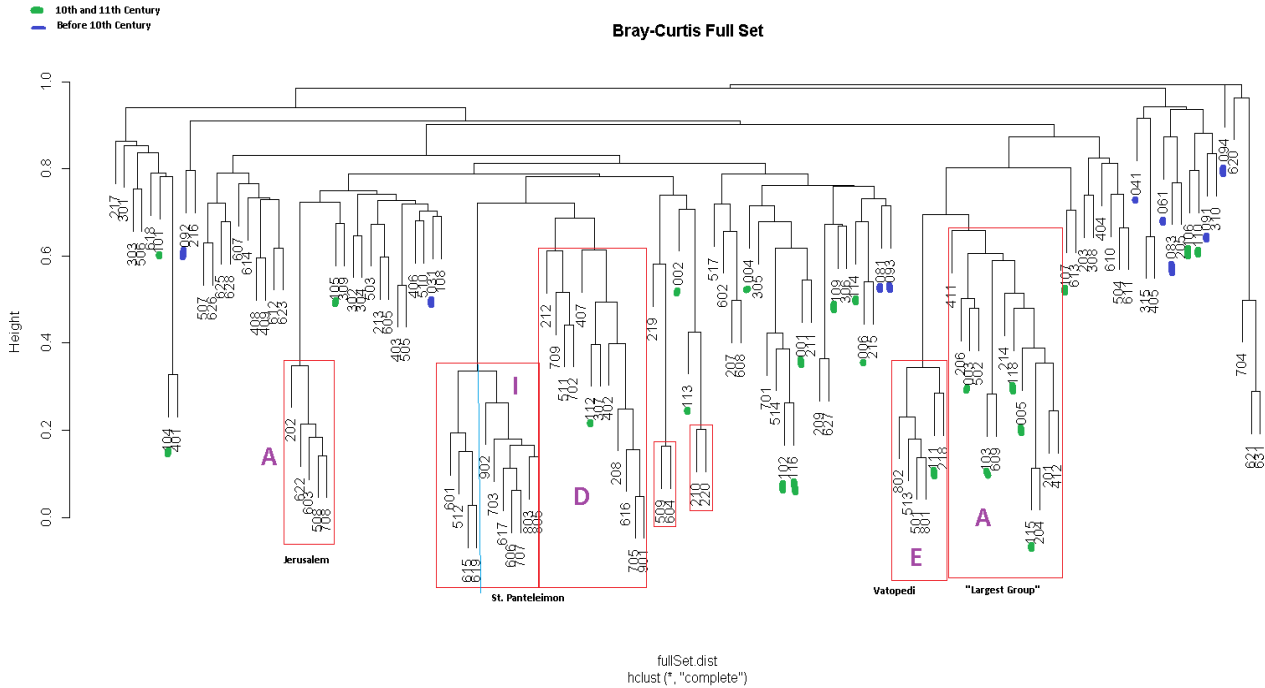
Figure 4 Hierarchical Clustering with Bray-Curtis on full data set

Only Old Hierarchical Cluster Tree

onlyOld.dist
hclust (*, "complete")
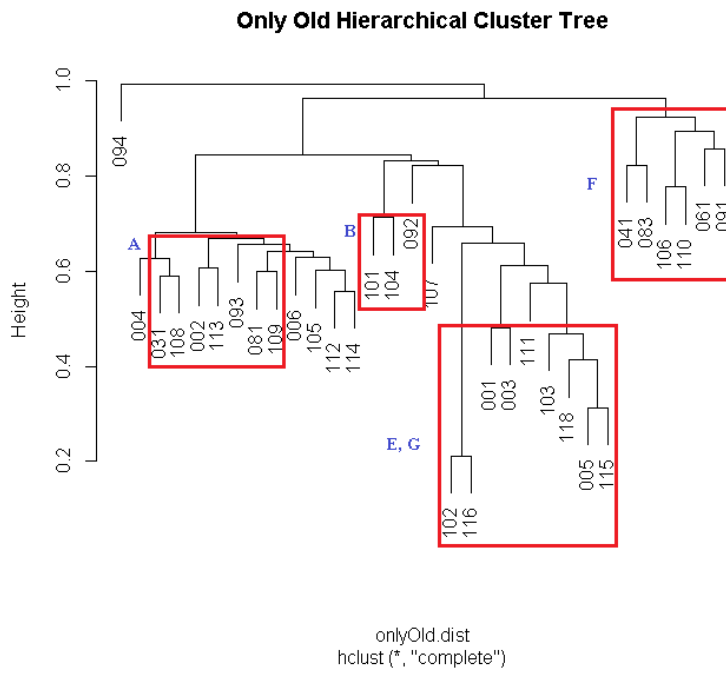
Figure 5 Hierarchical clustering using Bray-Curtis on the 32 oldest documents
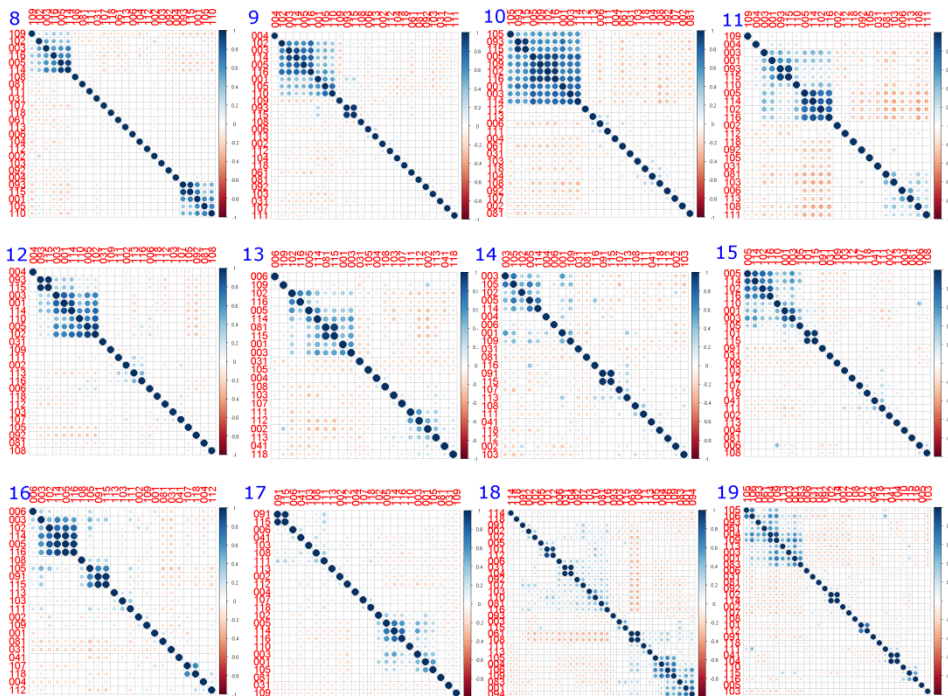
Figure 6 Correlation plot chapter-by-chapter of the 32 oldest documents

Two-level Default Theories

Liangjun Zang[1,2]**, Cungen Cao**[1]**, and Yuefei Sui**[1]

[1]Key Laboratory of Intelligent Information Processing, Institute of Computing Technology,
Chinese Academy of Sciences
No.6 Kexueyuan South Road, Hai Dian Distinct, Beijing 100190, China
[2]University of Chinese Academy of Sciences
No.19 Yu Quan Road, Shi Jing Shan Distinct, Beijing 100049, China
ljzang.ict@gmail.com, cgcao@ict.ac.cn, yfsui@ict.ac.cn

Abstract— *Default Logic employs assumption-based default rules to draw plausible consequences in face of incomplete information. In ontology representation, there are two kinds of relations between concepts: subsumption relation and default subsumption relation. Subsumption relation is transitive, whereas default subsumption relation is transitive by default. Both default transitivity of default subsumption and default inheritance of default property should be represented as defaults about defaults, i.e. two-level defaults. None of existing default logics can represent two-level defaults. In this paper, we propose two-level default theories which augment default theories with two-level defaults. A two-level default theory can be divided into two levels and its extensions can be generated by two steps. We prove that normal two-level default theories cannot reduce to normal default theories. Specifically, there is a normal two-level default theory such that there exists no normal default theory such that they share the same set of extensions.*

Keywords: Default subsumption, Default Transitivity, Two-level default, Extension, Irreducibility

1. Introduction

Default Logic enables plausible reasoning based on default rules when faced with incomplete information. Since Default Logic was originally introduced by Raymond Reiter [1], many variants have been developed either to respect some intended properties, or to realize more appealing intuitions, or to solve computational difficulty.

• Prioritized versions of Default Logic implicitly or explicitly specify a priority relation over defaults and thereby tend to the extensions derived by the preferred rules only, such as [2][3].

• Justified Default Logic [4] and Constrained Default Logic [5] guarantee the existence of extensions. Constrained Default Logic and Rational Default Logic [6] enforce the joint consistency of the justifications of all applied defaults in an extension. Cumulative Default Logic [7] respects a "cumulativity" property.

• In order to generate intuitively more appealing extensions, Disjunctive Default Logic [8] adopts a special strategy dealing with disjunctive formulae; Weak extensions [9] check the validity of the prerequisite of a default against the ultimate extension that will be generated rather than the consequents that have been generated.

• Some variants attempt to reduce computational difficulty by restricting their base language to decidable subcases of First-Order Logic language, say propositional logic language [10][11], terminological logic language $\mathcal{ALCF}$[12], and OWL DL language [13].

Default Logic has a wide range of applications because incomplete information and defaults can be naturally found in many application areas, such as commonsense reasoning [14], nonmonotonic inheritance networks [15][16], and information retrieval [17].

An ontology is a formal, explicit specification of a shared conceptualization [18]. In ontology representation, the backbone of an ontology consists of a hierarchy of concepts (unary predicates), which is formed by subsumption relation between concepts. In addition, there is default subsumption relation between concepts, which means any instance of a concept is also an instance of another concept by default. Formally, a ontology can be represented as a triple $(\mathcal{C}, \sqsubseteq, \sqsubseteq_d)$ where $\mathcal{C}$ is a set of concepts, $\sqsubseteq$ is the subsumption relation between concepts, and $\sqsubseteq_d$ is the default subsumption relation between concepts. For example, consider the following information about birds, stated in the form of a theory:

$$\left\{ \begin{array}{l} Bird \sqsubseteq Wing, \ Bird \sqsubseteq_d Fly, \\ Sparrow \sqsubseteq Bird, \\ Ostrich \sqsubseteq Bird, \ Ostrich \sqsubseteq \neg Fly \end{array} \right\}$$

Subsumption relation is transitive; formally,

$$\frac{X \sqsubseteq Y, Y \sqsubseteq Z}{X \sqsubseteq Z} \qquad (1)$$

where X, Y, Z can be any concepts. Using this default in the theory derives "both sparrow and ostrich have wings". In contrast, default subsumption relation is not transitive but transitive by default; formally

$$\frac{X \sqsubseteq Y, Y \sqsubseteq_d Z : X \sqsubseteq_d Z}{X \sqsubseteq_d Z} \qquad (2)$$

where X, Y, Z can be any concepts. Applying this default to the theory make it possible to infer $Sparrow \sqsubseteq_d Fly$. On the contrary introducing $Ostrich \sqsubseteq_d Fly$ would cause inconsistency with fact $Ostrich \sqsubseteq \neg Fly$, and thus application of the default is blocked. Let $X = S$parrow$, Y = B$ird$, Z = F$ly. Since statement $B \sqsubseteq_d F$ can be interpreted as a default $\frac{B(x):F(x)}{F(x)}$, the default (2) is equivalent to the form

$$\frac{\forall x(S(x) \to B(x)), \frac{B(x):F(x)}{F(x)} : \frac{S(x):F(x)}{F(x)}}{\frac{S(x):F(x)}{F(x)}}$$

which is called a *two-level default*. As in the example, the two-level default can be used to represent default inheritance of default property. That is, the default property "can fly by default" of the notion "Birds" can be defaultly inherited by its subconcepts "Sparrow" and "Ostrich". None of existing default logics can represent such kind of two-level defaults. Therefore, it is interesting and significant to study the default theories that contain both defaults and two-level defaults.

In this paper, we introduce the notion of *Two-level Default Theories* (TDTs), which augments a default theory with two-level defaults. A two-level default theory is a triple $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$, where $\mathbf{W}$ is a set of formulas, $\mathbf{D}$ a set of defaults and the negation of defaults, and $\mathbf{D^2}$ a set of two-level defaults. A two-level default theory is *normal* if all defaults are normal and all constituents of two-level defaults are normal defaults, called *Normal Two-level Default Theories* (NTDTs). A two-level default theory can be divided into two levels, and its extensions are generated by two steps. Firstly, $(\mathbf{D}, \mathbf{D^2})$ constitute a default theory, called *default-level default theory*. The two-level defaults in $\mathbf{D^2}$ are applied to extend $\mathbf{D}$, generating a set of defaults and the negation of defaults, called *default-level extensions*. Secondly, $\mathbf{W}$ and all defaults in a default-level extension constitute a default theory, called *formula-level default theory*. The defaults in a default-level extension are applied to extend $\mathbf{W}$, generating a set of formulas, called *formula-level extensions*. Finally, any formula-level extension is regarded as an extension of the two-level default theory $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$.

Normal two-level default theories (NTDTs) are more expressive than normal default theories (NDTs). All NDTs are NTDTs, whereas there are some NTDTs that cannot reduce to NDTs. More specifically, NTDTs have property of *irreducibility*: there is a NTDT such that there is no NDT such that they share the same set of extensions.

Our contributions include the following aspects:
• introduce the notions of two-level defaults and two-level default theories;
• analyze the properties of extensions of a (normal) two-level default theory;
• present a procedure of generating all extensions of a normal two-level default theory;
• prove that there exist some normal two-level default theories that cannot reduce to normal default theories.

The rest of this paper is structured as follows: In section 2, we review Default Logic and Prioritized Default Logic. In section 3, we present the basic definitions and general properties about extensions of default-level default theories. In section 4, we present the basic definitions and general properties about extensions of two-level default theories, and present a procedure of computing extensions. In section 5, we present and prove the *irreducibility* property of normal two-level default theories. Finally, we conclude our work and discuss some promising directions.

2. Preliminaries

In this section we review Reiter's Default Logic[1] and Brewka's Prioritized Default Logic[3].

2.1 Reiter's Default Logic

Definition 1: (**Defaults**) A *default* is any expression of the form

$$\delta = \frac{\alpha : \beta_1, ..., \beta_n}{\gamma},$$

where
• $\alpha, \beta_1, ..., \beta_n, \gamma$ are all first order formulae and $n \geq 1$;
• α is called the *prerequisite* of δ, denoted as $pre(\delta)$;
• $\beta_1, ..., \beta_n$ is called the *justifications* of δ, denoted as $just(\delta)$;
• γ is called the *consequent* of δ, denoted as $cons(\delta)$;
• δ is a *closed default* if none of $\alpha, \beta_1, ..., \beta_n, \gamma$ contains a free variable; otherwise it is a *open default*;
• δ is *normal* if it has the form $\alpha : \beta/\beta$.

A default can be intuitively interpreted as follows: if α is known, and if it is consistent to assume $\beta_1, ..., \beta_n$, then conclude γ.

Definition 2: (**Default Theories**) A *default theory* is a pair

$$\mathbf{T} = (\mathbf{W}, \mathbf{D})$$

where
• $\mathbf{W}$ is a set of closed formulae and $\mathbf{D}$ is a set of defaults;
• $\mathbf{T}$ is a *closed* default theory if every default of $\mathbf{D}$ is closed; otherwise it is an *open* default theory;
• $\mathbf{T}$ is a *normal* default theory if every default of $\mathbf{D}$ is normal.

Given a default theory, an extension of it can be derived by applying as many defaults as consistently possible. Intuitively, an extension represent a acceptable set of beliefs that an agent might have with the given default theory. Extensions were originally given a fixed point definition; here we present a more intuitive definition, also due to Reiter.

Definition 3: (**Extensions**) Let $\mathbf{T} = (\mathbf{W}, \mathbf{D})$ be a closed default theory, and define $\mathbf{E_0} = \mathbf{W}$ and

$$\mathbf{E_{i+1}} = Th(\mathbf{E_i}) \cup \{\gamma \mid \frac{\alpha : \beta_1, ..., \beta_n}{\gamma} \in \mathbf{D}, \alpha \in \mathbf{E_i}, \neg\beta_1, ..., \neg\beta_n \notin \mathbf{E}\}.$$

Then, $\mathbf{E}$ is an extension of $(\mathbf{W}, \mathbf{D})$ if $\mathbf{E} = \bigcup_{i=0}^{\infty} \mathbf{E_i}$.

In the case of an open default theory, $\mathbf{W}$ and $\mathbf{D}$ are converted into skolemized forms, and any open default is converted into a set of closed defaults by assigning ground terms (over the Herbrand universe [1]) to all free variables occurring in the open default. For example, given an open default $bird(x) : fly(x)/fly(x)$ and two facts $bird(tweety)$ and $bird(clyde)$, the default theory represented is equal to

$$\left(\begin{array}{c} \{bird(tweety), bird(clyde)\}, \\ \left\{ \frac{bird(tweety):fly(tweety)}{fly(tweety)}, \frac{bird(clyde):fly(clyde)}{fly(clyde)} \right\} \end{array} \right)$$

In the future discussions, we implicitly assume that default theories are closed when computing extensions.

The following propositions describe general properties of the extensions of a general default theory.

Proposition 1: (**Existence**) A default theory may have zero, one, multiple extension(s).

Proposition 2: (**Consistency**) A default theory $(\mathbf{W}, \mathbf{D})$ has an inconsistent extension *iff* $\mathbf{W}$ is inconsistent. Furthermore, if $(\mathbf{W}, \mathbf{D})$ has an inconsistent extension, then it has no other extension.

Proposition 3: (**Minimality**) Let $\mathbf{E_1}$ and $\mathbf{E_2}$ be two extensions of a default theory $(\mathbf{W}, \mathbf{D})$. If $\mathbf{E_1} \subseteq \mathbf{E_2}$, then $\mathbf{E_1} = \mathbf{E_2}$.

If a default theory is normal, there are some additional properties about extensions, including *semi-monotonicity*, *existence* and *orthogonality*.

Proposition 4: (**Semi-monotonicity**) Let $\mathbf{D_1}$ and $\mathbf{D_2}$ be two sets of normal defaults and $\mathbf{D_1} \subseteq \mathbf{D_2}$. If $\mathbf{E_1}$ is an extension of $(\mathbf{W}, \mathbf{D_1})$, then there exists an extension $\mathbf{E_2}$ of $(\mathbf{W}, \mathbf{D_2})$ such that $\mathbf{E_1} \subseteq \mathbf{E_2}$.

Proposition 5: (**Existence**) A normal default theory has at least one extension.

Proposition 6: (**Orthogonality**) Let $\mathbf{E_1}$ and $\mathbf{E_2}$ be two extensions of a default theory $(\mathbf{W}, \mathbf{D})$. If $\mathbf{E_1} \neq \mathbf{E_2}$, then $\mathbf{E_1}$ is inconsistent with $\mathbf{E_2}$.

2.2 Prioritized Default Logic

Brewka present a prioritized version of Reiter's Default Logic for normal defaults[3]. A prioritized normal default theory supplements a normal default theory with a priority order over the defaults. The priority order defines some constraints about which defaults should precede which other defaults when there are multiple applicable ones. As a result, defaults of lower priority are applied only if this does not lead to a conflict with those of higher priority. Following a tradition in the field of nonmonotonic reasoning, $\delta_1 < \delta_2$ is used to represent that δ_1 has a higher priority than δ_2.

Definition 4: (**Prioritized Default Theories**) A *prioritized default theory* is a triple

$$\mathbf{T}^< = (\mathbf{W}, \mathbf{D}, <)$$

where
- $(\mathbf{W}, \mathbf{D})$ is a normal default theory;

- $<$ a strict partial order on $\mathbf{D}$;
- a *fully prioritized default theory* is a prioritized normal default theory whose priority order is a strict well order.

Definition 5: (**Active Defaults**) A default $\delta = \alpha : \beta/\beta$ is *active* in a deductively closed set of formulas $\mathbf{E}$ (i.e. $\mathbf{E} = Th(\mathbf{E})$) if $\alpha \in \mathbf{E}$ and $\neg\beta \notin \mathbf{E}$ and $\beta \notin \mathbf{E}$.

Definition 6: (**Prioritized Extensions**) Let $(\mathbf{W}, \mathbf{D}, \ll)$ be a fully prioritized normal default theory, and $\mathbf{E}$ a set of formulas. $\mathbf{E}$ is the *prioritized extension* of $(\mathbf{W}, \mathbf{D}, \ll)$ if $\mathbf{E} = \bigcup_{i=0}^{\infty} \mathbf{E}_i$ where $\mathbf{E}_0 = Th(\mathbf{W})$ and

$$\mathbf{E_{i+1}} = \begin{cases} \mathbf{E_i} & \text{if no default is active in } \mathbf{W}_i, \\ Th(\mathbf{E_i} \cup cons(\delta)) & \text{o.w., where } \delta \text{ is } \ll\text{-minimal} \\ & \text{default that is active in } \mathbf{E_i}. \end{cases}$$

$\mathbf{E}$ is a prioritized extension of a prioritized normal default theory $(\mathbf{W}, \mathbf{D}, <)$ if $\mathbf{E}$ is the prioritized extension of a fully prioritized default theory $(\mathbf{W}, \mathbf{D}, \ll)$ where $< \subseteq \ll$.

The following theorem states the relation between prioritized extensions and extensions.

Theorem 1: For any set of formulas $\mathbf{E}$, $\mathbf{E}$ is an extension of a normal default theory $(\mathbf{W}, \mathbf{D})$ iff there is a well order $\ll$ on $\mathbf{D}$ such that $\mathbf{E}$ is the prioritized extension of a fully prioritized normal default theory $(\mathbf{W}, \mathbf{D}, \ll)$.

3. Default-level Default Theories

In accordance with the definitions and properties of normal default theories, we define *default-level normal default theories* and *default-level extensions*, present the properties of default-level extensions.

3.1 Basic Definitions

Definition 7: (**The Negation of Defaults**) *The negation of a normal default* is any expression of the form

$$\neg \frac{\alpha : \beta}{\beta},$$

where α, β are both first order formulae, and $\neg$ is a logical connective. Sometimes we use $\alpha \nrightarrow_d \beta$ as an abbreviation of the negation of a normal default.

Consider the example $bird(x) \nrightarrow_d swim(x)$, which means "Abnormally, birds can swim". Notice the difference between $bird(x) \nrightarrow_d swim(x)$ and $bird(x) \rightarrow_d \neg swim(x)$. The latter means "Normally, birds cannot swim". Intuitively, $bird(x) \rightarrow_d \neg swim(x)$ entails $bird(x) \nrightarrow_d swim(x)$, but $bird(x) \rightarrow_d \neg swim(x)$ is not implied by $bird(x) \nrightarrow_d swim(x)$. Notably, $\neg\neg\delta$ means δ, i.e. the negation of the negation of a normal default is the default itself.

Definition 8: (**Consistency**) Let $\mathbf{D}$ be a set of normal defaults and the negation of normal defaults. $\mathbf{D}$ is *inconsistent* if there exists a normal default δ such that $\delta \in \mathbf{D}$ and $\neg\delta \in \mathbf{D}$; otherwise $\mathbf{D}$ is *consistent*.

Definition 9: (**Normal Two-level Defaults**) A *normal two-level default* is any expression of the form

$$\theta = \frac{\delta_1, \ldots, \delta_n : \delta}{\delta}$$

where

- $\delta_1, \ldots, \delta_n, \delta$ is either a normal default or the negation of a normal default, and $n \geq 0$;
- $\delta_1, \ldots, \delta_n$ is called the *prerequisite* of θ, denoted as $pre(\theta)$;
- δ is called the *justification* and the *consequent* of θ, denoted as $just(\theta)$ and $cons(\theta)$ respectively;
- If $\delta_1, \ldots, \delta_n, \delta$ are all closed, then θ is *closed*, otherwise θ is *open*.

Intuitively a two-level default can be interpreted as follows: if $\delta_1, \ldots, \delta_n$ are known, and if it is consistent to assume δ, then conclude δ.

Definition 10: (**Default-level Default Theories**) A *default-level default theory* is a pair

$$\mathbf{T}_\delta = (\mathbf{D}, \mathbf{D^2})$$

where $\mathbf{D}$ is a set of normal defaults and the negation of normal defaults, and $\mathbf{D^2}$ is a set of two-level normal defaults.

Definition 11: (**Default-level Extensions**) Let $\mathbf{T}_\delta = (\mathbf{D}, \mathbf{D^2})$ be a default-level default theory, and define

$$\begin{cases} \mathbf{E_0} = \mathbf{D} \\ \mathbf{E_{i+1}} = \mathbf{E_i} \cup \{\delta \mid \frac{\delta_1, \ldots, \delta_n : \delta}{\delta} \in \mathbf{D^2}, \delta_1, \ldots, \delta_n \in \mathbf{E_i}, \neg\delta \notin \mathbf{E_\delta}\}. \end{cases}$$

Then, $\mathbf{E}_\delta$ is an extension of $(\mathbf{D}, \mathbf{D^2})$ if $\mathbf{E}_\delta = \bigcup_{i=0}^\infty \mathbf{E_i}$.

Corresponding to *Prioritized Default Theories* and their extensions, we can define *Default-level Prioritized Default Theories* and their extensions.

Definition 12: (**Active Two-level Defaults**) Let $\theta = \delta_1, \ldots, \delta_n : \delta/\delta$ be a normal two-level default, and $\mathbf{D}$ a set of normal defaults and the negation of normal defaults. θ is *active* w.r.t. $\mathbf{D}$ if $\delta_1, \ldots, \delta_n \in \mathbf{D}$ and $\neg\delta \notin \mathbf{D}$ and $\delta \notin \mathbf{D}$.

Definition 13: (**Default-level PDTs**) A *default-level prioritized default theory* is a triple

$$\mathbf{T}_\delta^< = (\mathbf{D}, \mathbf{D^2}, <)$$

where

- $(\mathbf{D}, \mathbf{D^2})$ is a default-level normal default theory;
- $<$ is a strict partial order on $\mathbf{D^2}$;
- a *default-level fully prioritized normal default theory* is a default-level prioritized normal default theory whose priority order is a strict well order.

Definition 14: (**Prioritized Extensions**) Let $(\mathbf{D}, \mathbf{D^2}, \ll)$ be a fully prioritized default-level normal default theory, $(\mathbf{D}, \mathbf{D^2}, <)$ a prioritized default-level normal default theory, $\mathbf{E}_\delta$ a set of normal defaults and the negation of normal defaults.

$\mathbf{E}_\delta$ is the extension of $(\mathbf{D}, \mathbf{D^2}, \ll)$ if $\mathbf{E}_\delta = \bigcup_{i=0}^\infty \mathbf{E_i}$ where

$$\begin{cases} \mathbf{E_0} = \mathbf{D} \\ \mathbf{E_{i+1}} = \begin{cases} \mathbf{E_i} & \text{if no default is active in } \mathbf{E_i}, \\ \mathbf{E_i} \cup cons(\theta) & \text{o.w., where } \theta \text{ is } \ll\text{-minimal} \\ & \text{default that is active in } \mathbf{E_i}. \end{cases} \end{cases}$$

$\mathbf{E}_\delta$ is an extension of $(\mathbf{D}, \mathbf{D^2}, <)$ if $\mathbf{E}_\delta$ is the prioritized extension of a fully prioritized default theory $(\mathbf{D}, \mathbf{D^2}, \ll)$ where $< \subseteq \ll$.

3.2 General Properties

A default-level default theory can be viewed as a propositional default theory. Hence, the extensions of a default-level default theory satisfy *consistency* and *minimality*; the extensions of a normal default-level default theory satisfy *semi-monotonicity*, *existence*, and *orthogonality*.

Proposition 7: (**Consistency**) Let $\mathbf{E}_\delta$ be an extension of a default-level default theory $\mathbf{T}_\delta = (\mathbf{D}, \mathbf{D^2})$. If $\mathbf{D}$ is consistent, then $\mathbf{E}_\delta$ is consistent.

Proposition 8: (**Minimality**) Let $\mathbf{T}_\delta = (\mathbf{D}, \mathbf{D^2})$ be a default-level default theory. If $\mathbf{E_1}$ and $\mathbf{E_2}$ are extensions of $\mathbf{T}_\delta$ and $\mathbf{E_1} \subseteq \mathbf{E_2}$, then $\mathbf{E_1} = \mathbf{E_2}$.

Proposition 9: (**Semi-monotonicity**) Let $\mathbf{D_1^2}$ and $\mathbf{D_2^2}$ be two sets of normal two-level defaults and $\mathbf{D_1^2} \subseteq \mathbf{D_2^2}$. If $\mathbf{E_1}$ is an extension of $(\mathbf{D}, \mathbf{D_1^2})$, then there exists an extension $\mathbf{E_2}$ of $(\mathbf{D}, \mathbf{D_2^2})$ such that $\mathbf{E_1} \subseteq \mathbf{E_2}$.

Proposition 10: (**Existence**) A normal default-level default theory always has an extension.

Proposition 11: (**Orthogonality**) Let $(\mathbf{D}, \mathbf{D^2})$ be a normal default-level default theory. If $\mathbf{E_1}$ and $\mathbf{E_2}$ are extensions of $(\mathbf{D}, \mathbf{D^2})$ and $\mathbf{E_1} \subseteq \mathbf{E_2}$, then $\mathbf{E_1}$ is inconsistent with $\mathbf{E_2}$.

The following theorem states the relation between prioritized extensions and extensions, which provides an operational method of constructing extensions for normal default-level default theories.

Theorem 2: For any set $\mathbf{E}_\delta$ of normal defaults and the negation of normal defaults, $\mathbf{E}_\delta$ is an extension of a normal default-level default theory $(\mathbf{D}, \mathbf{D^2})$ iff there is a well order $\ll$ on $\mathbf{D^2}$ such that $\mathbf{E}_\delta$ is the prioritized extension of a fully prioritized normal default-level default theory $(\mathbf{D}, \mathbf{D^2}, \ll)$.

4. Two-level Default Theories

In this section, we formally define two-level default theories and their extensions, discuss extension's properties, and propose a procedure of computing extensions of NTDTs.

4.1 Basic Definitions

Definition 15: (**Two-level Default Theories**) A *two-level default theory* is a triple

$$\mathbf{T} = (\mathbf{W}, \mathbf{D}, \mathbf{D^2})$$

where

- $\mathbf{W}$ is a set of closed formulae,

- **D** is a set of defaults and the negation of defaults,
- **D²** is a set of two-level defaults,
- **T** is a *normal two-level default theory* (NTDT) if all defaults in **D** are normal and all two-level defaults in **D²** are normal.

A two-level default theory can be divided into two levels. The motivation can be presented in terms of syntax and computation. On the syntactic side, two-level defaults and defaults lie in different syntactic levels. Two-level defaults are defaults about defaults, which are applied to extend a set of defaults, whereas defaults are defaults about formulas, which are applied to extend a set of formulas.

On the computational side, dividing two-level defaults and defaults into different levels simplifies the computation of extensions. As the fixed point definition of extension in Default Logic, an extension is generated after applying as many defaults as possible. After computing the extensions of the default theories at default level and at the formula level respectively, it is guaranteed that neither two-level default nor default or can be applied.

Intuitively we can describe the procedure of computing all extensions of a two-level default theory $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ as follows. Firstly, **D** and **D²** constitute the default-level default theory. Applying the two-level defaults of **D²** to **D** generates default-level extensions. Secondly, for each default-level extension $\mathbf{E^2}$, **W** and the defaults in $\mathbf{E^2}$ constitute a formula-level default theory. Applying the defaults in $\mathbf{E^2}$ to **W** derives formula-level extensions. Finally, any formula-level extension is viewed as an extension of the two-level default theory .

A default-level extension may contain the negation of normal defaults. The negation of normal defaults do not take part in generating extensions. This property meets human intuitions. Suppose we know "$Student(Chandler)$" and "$Student(x) \not\rightarrow_d Swim(x)$". Intuitively we conclude neither "Chandler can swim" nor "Chandler cannot swim". One function of the negation of a default is to prevent a two-level default from generating a new default.

Definition 16: Let **D** be a set of defaults and the negation of defaults. We define

$$\mathbf{D}^+ = \left\{ \delta \in \mathbf{D} \,\middle|\, \delta = \frac{\alpha : \beta}{\beta} \right\}.$$

Definition 17: (**Extensions**) A set of formulas **E** is an extension of a two-level default theory $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ if there is a set of defaults and the negation of defaults $\mathbf{E}_\delta$ such that

$$\begin{cases} \mathbf{E}_\delta \text{ is a default-level extension of } (\mathbf{D}, \mathbf{D^2}), \\ \mathbf{E} \text{ is a formula-level extension of } (\mathbf{W}, \mathbf{E}_\delta^+). \end{cases}$$

4.2 General Properties

Below we present the properties of a two-level default theory, including *consistency* and *non-minimality*; and the properties of a normal two-level default theory, including *semi-monotonicity*, *existence*, and *non-orthogonality*.

Proposition 12: (**Consistency**) Let **E** be an extension of a two-level default theory $\mathbf{T} = (\mathbf{W}, \mathbf{D}, \mathbf{D^2})$. If **W** is consistent, then **E** is consistent.

Proof: Since **E** is an extension of **T**, there must be a default-level extension $\mathbf{E}_\delta$ of **T** such that **E** is an extension of $(\mathbf{W}, \mathbf{E}_\delta^+)$. According to consistency property of extensions of DL default theories, we have **E** is consistent if **W** is consistent. ∎

Proposition 13: (**Non-minimality**) There is a two-level default theory such that it has two extensions $\mathbf{E_1}$ and $\mathbf{E_2}$ such that $\mathbf{E_1} \subset \mathbf{E_2}$.

Proof: We prove this proposition by creating a two-level default theory that has two strictly inclusive extensions. See the example in the *irreducibility* theorem (section 5). ∎

Proposition 14: (**Semi-monotonicity**) Let $\mathbf{T_1} = (\mathbf{W}, \mathbf{D}, \mathbf{D_1^2})$ and $\mathbf{T_2} = (\mathbf{W}, \mathbf{D}, \mathbf{D_2^2})$ be two normal two-level default theories such that $\mathbf{D_1^2} \subseteq \mathbf{D_2^2}$. If $\mathbf{E_1}$ is an extension of $\mathbf{T_1}$, then there is an extension $\mathbf{E_2}$ of $\mathbf{T_2}$ such that $\mathbf{E_1} \subseteq \mathbf{E_2}$.

Proof: Since $\mathbf{E_1}$ is an extension of $\mathbf{T_1}$, there must be a default-level extension $\mathbf{D_1^*}$ such that $\mathbf{E_1}$ is an extension of $(\mathbf{W}, \mathbf{D_1^*})$. By the semi-monotonicity property of default-level extensions, there exists a default-level extension $\mathbf{D_2^*}$ of $\mathbf{T_2}$ such that $\mathbf{D_1^*} \subseteq \mathbf{D_2^*}$. By the semi-monotonicity property of formula-level extensions, there exists a formula-level extension $\mathbf{E_2}$ of $\mathbf{T_2}$ such that $\mathbf{E_1} \subseteq \mathbf{E_2}$. That is, $\mathbf{E_2}$ is an extension of $\mathbf{T_2}$ and $\mathbf{E_1} \subseteq \mathbf{E_2}$. ∎

It worth noting that the same conclusion is not true for $\mathbf{T_1} = (\mathbf{W}, \mathbf{D_1}, \mathbf{D_1^2})$ and $\mathbf{T_2} = (\mathbf{W}, \mathbf{D_2}, \mathbf{D_2^2})$ such that $\mathbf{D_1} \subseteq \mathbf{D_2}$ and $\mathbf{D_1^2} \subseteq \mathbf{D_2^2}$. This can be illustrated by the following example.

Example 1: Let $\mathbf{T_1} = (\{\alpha\}, \emptyset, \{\frac{:\delta}{\delta}\})$, $\mathbf{T_2} = (\{\alpha\}, \{\neg\delta\}, \{\frac{:\delta}{\delta}\})$. $\mathbf{T_1}$ have one default-level extension $\{\delta\}$ and one formula-level extension $\mathbf{E_1} = Th(\{\alpha, \beta\})$. $\mathbf{T_2}$ have one default-level extension $\{\neg\delta\}$ and one formula-level extension $\mathbf{E_2} = Th(\{\alpha\})$. That is, we have $\mathbf{E_2} \subseteq \mathbf{E_1}$. □

Proposition 15: (**Existence**) A normal two-level default theory always has an extension.

Proof: Let $\mathbf{T} = (\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ be a normal two-level default theory. By the existence property of default-level extensions, $(\mathbf{D}, \mathbf{D^2})$ always has an extension, say $\mathbf{E}_\delta$. By the existence property of formula-level extensions, $(\mathbf{W}, \mathbf{E}_\delta^+)$ has an extension, say **E**, which is an extension of **T**. ∎

Different default-level extensions are inconsistent, and distinct formula-level extensions corresponding to identical default-level extension are inconsistent. However, distinct formula-level extensions corresponding to different default-level extensions might be consistent.

Proposition 16: (**Non-orthogonality**) There is a normal two-level default theory such that it has two extensions $\mathbf{E_1}$ and $\mathbf{E_2}$ such that $\mathbf{E_1}$ is consistent with $\mathbf{E_2}$.

Proof: We prove this proposition by creating a normal two-level default theory which has two consistent extensions. See the example in the *irreducibility* theorem (section 5). ∎

4.3 Extension Computation

In this section we present a procedure for computing all extensions of a given normal two-level default theory and prove its correctness.

Procedure 1 Computing-Two-Level-Extensions

Input: $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$; // normal two-level default theory
Output: $\mathcal{E}$; // the set of all extensions

1: $\mathcal{E} \leftarrow \emptyset$;
2: **for all** well order $\ll_2$ over $\mathbf{D^2}$ **do**
3:      $\mathbf{E^2} \leftarrow \mathbf{D}$;
4:      **while** there is an active two-level default w.r.t. $\mathbf{E^2}$ in $\mathbf{D^2}$ **do**
5:          $\theta \leftarrow \ll_2$-minimal active two-level default w.r.t. $\mathbf{E^2}$;
6:          $\mathbf{E^2} \leftarrow \mathbf{E^2} \cup cons(\theta)$;
7:      **end while**
8:      $\mathcal{D} \leftarrow \mathcal{D} \cup \mathbf{E^2}$;
9: **end for**
10: **for all** $\mathbf{E^2} \in \mathcal{D}$ **do**
11:     **for all** well order $\ll_1$ over $(\mathbf{E^2})^+$ **do**
12:         $\mathbf{E^1} \leftarrow Th(\mathbf{W})$;
13:         **while** there is an active default w.r.t. $\mathbf{E^1}$ in $(\mathbf{E^2})^+$ **do**
14:             $\delta \leftarrow \ll_1$-minimal active default w.r.t. $\mathbf{E^1}$;
15:             $\mathbf{E^1} \leftarrow Th(\mathbf{E^1} \cup cons(\delta))$;
16:         **end while**
17:         $\mathcal{E} \leftarrow \mathcal{E} \cup \mathbf{E^1}$;
18:     **end for**
19: **end for**
20: **Return** $\mathcal{E}$;

Procedure 1 takes three parameters $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ which denotes a normal two-level default theory , and outputs all extensions of the normal two-level default theory.

Lines 2-9 generate all default-level extensions of default-level default theory $(\mathbf{D}, \mathbf{D^2})$. To generate all default-level extensions, it needs to enumerate all possible well orders over $\mathbf{D^2}$ (line 2). Lines 3-8 compute the extension of $(\mathbf{D}, \mathbf{D^2}, \ll_2)$. The variant $\mathbf{E^2}$ is used to store the value of the extension, which is initialized with $\mathbf{D}$ (line 3). Lines 4-7 iteratively apply the $\ll_2$-minimal active default to $\mathbf{E^2}$ until there is no active two-level default in $\mathbf{D^2}$. Then a default-level extension $\mathbf{E^2}$ is generated and put into $\mathcal{D}$ (line 8).

For each default-level extension $\mathbf{E^2} \in \mathcal{D}$, the defaults in $\mathbf{E^2}$ and the set of formulas $\mathbf{W}$ constitute a TDL formula-level default theory $(\mathbf{W}, (\mathbf{E^2})^+)$ (line 10).

Lines 11-18 generate all formula-level extensions of formula-level default theory $(\mathbf{W}, (\mathbf{E^2})^+)$. To generate all formula-level extensions, it needs to enumerate all possible well orders over $(\mathbf{E^2})^+$ (line 11). Lines 12-17 compute the

extension of $(\mathbf{W}, (\mathbf{E^2})^+, \ll_1)$. The variant $\mathbf{E^1}$ is used to store the value of the extension, which is initialized with $Th(\mathbf{W})$ (line 12). Lines 13-16 iteratively apply the $\ll_1$-minimal active default to $\mathbf{E^1}$ until there is no active default in $(\mathbf{E^2})^+$. Then a formula-level extension $\mathbf{E^1}$ is generated and put into $\mathcal{E}$ (line 17). Any formula-level extension is regarded as an extension of the two-level default theory $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$.

Procedure 1 is not a decidable procedure because the First-Order Logic inference is undecidable. Our objective is to propose an operational method of deriving extensions for two-level default theories. The following theorem prove the correctness of our algorithm.

Theorem 3: The output of the procedure *Computing-Two-Level-Extensions* with input parameters $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ are exactly all extensions of the two-level default theory $(\mathbf{W}, \mathbf{D}, \mathbf{D^2})$.

Proof: The correctness of the procedure follows from theorem 2 and theorem 1 immediately. ∎

5. Irreducibility of NTDTs

The following theorem states that there exists a normal two-level default theory (NTDT) whose extensions cannot be exactly generated by any single normal default theory (NDT), which implies that NTDTs have stronger expressiveness than NDTs.

Theorem 4: (**Irreducibility**) There is a normal two-level default theory $\mathbf{T^2}$ such that there is no normal default theory $\mathbf{T}$ such that, for any set of formulas $\mathbf{E}$, $\mathbf{E}$ is an extension of $\mathbf{T^2}$ iff $\mathbf{E}$ is an extension of $\mathbf{T}$.

Proof: Construct a NTDT $\mathbf{T^2}$ such that
(1) $\mathbf{T^2}$ has 3 default-level extensions $\mathbf{E_1}, \mathbf{E_2}, \mathbf{E_3}$;
(2) $\mathbf{T^2}$ has 7 default-level extensions $\mathbf{E_{11}}, \mathbf{E_{12}}, \mathbf{E_{21}}, \mathbf{E_{22}}, \mathbf{E_{31}}, \mathbf{E_{32}}, \mathbf{E_{33}}$;
(3) $\mathbf{E_{11}}$ is consistent with $\mathbf{E_{21}}$.

Let A=Archaeopteryx, B=Bird, D=Dinosaur, F=Feather, T=Teeth.

Let $\mathbf{T^2} = (\mathbf{W}, \mathbf{D}, \mathbf{D^2})$ is a TDL normal default theory, where

$$\mathbf{W} = \{A(x)\},$$

$$\mathbf{D} = \left\{ \begin{array}{l} B \sqsubseteq_d F, \ \ F \sqsubseteq_d \neg T, \ \ A \sqsubseteq_d T, \\ D \sqsubseteq_d T, \ \ T \sqsubseteq_d \neg F, \ \ A \sqsubseteq_d F \end{array} \right\},$$

$$\mathbf{D^2} = \left\{ \begin{array}{cc} \dfrac{true : A \sqsubseteq_d B}{A \sqsubseteq_d B}, & \dfrac{A \sqsubseteq_d B : A \not\sqsubseteq_d D}{A \not\sqsubseteq_d D}, \\[3mm] \dfrac{true : A \sqsubseteq_d D}{A \sqsubseteq_d D}, & \dfrac{A \sqsubseteq_d D : A \not\sqsubseteq_d B}{A \not\sqsubseteq_d B} \end{array} \right\}.$$

Then, $\mathbf{T^2}$ has three default-level extensions:

$$\mathbf{E_1} = \mathbf{D} \cup \{A \sqsubseteq_d B, A \not\sqsubseteq_d D\};$$

$$\mathbf{E_2} = \mathbf{D} \cup \{A \sqsubseteq_d D, A \not\sqsubseteq_d B\};$$

$$\mathbf{E_3} = \mathbf{D} \cup \{A \sqsubseteq_d B, A \sqsubseteq_d D\}.$$

$\mathbf{E_1}$ corresponds to two formula-level extensions:

$$\mathbf{E_{11}} = Th(\{A(x), B(x), F(x), T(x)\}),$$
$$\mathbf{E_{12}} = Th(\{A(x), B(x), F(x), \neg T(x)\});$$

$\mathbf{E_2}$ corresponds to two formula-level extensions:

$$\mathbf{E_{21}} = Th(\{A(x), D(x), F(x), T(x)\}),$$
$$\mathbf{E_{22}} = Th(\{A(x), D(x), \neg F(x), T(x)\});$$

$\mathbf{E_3}$ corresponds to three formula-level extensions:

$$\mathbf{E_{31}} = Th(\{A(x), B(x), D(x), F(x), T(x)\}),$$
$$\mathbf{E_{32}} = Th(\{A(x), B(x), D(x), \neg F(x), T(x)\}),$$
$$\mathbf{E_{33}} = Th(\{A(x), B(x), D(x), F(x), \neg T(x)\}).$$

We can see that $\mathbf{E_{11}}, \mathbf{E_{21}}$; $\mathbf{E_{11}}, \mathbf{E_{31}}$; $\mathbf{E_{21}}, \mathbf{E_{31}}$; $\mathbf{E_{12}}, \mathbf{E_{33}}$; and $\mathbf{E_{22}}, \mathbf{E_{32}}$ are all consistent. According to the orthogonality property of extensions of normal default theories, there is no normal default theory $\mathbf{T}$ such that both $\mathbf{E_{11}}$ and E_{21} are extensions of $\mathbf{T}$. ∎

6. Conclusion and Discussions

In an attempt to represent the default transitivity of default subsumption between concepts in ontologies, we introduced two-level defaults and two-level default theories as a hierarchical variant of classical defaults and default theories. We also present the properties and computing procedure of extensions of two-level default theories. In addition, we prove that normal two-level default theories cannot reduce to normal default theories. To represent default reasoning in ontologies, two-level default description logic need to be further considered in the future.

Acknowledgment

References

[1] R. Reiter, "A logic for default reasoning," *Artificial intelligence*, vol. 13, no. 1-2, pp. 81–132, 1980.

[2] F. Baader and B. Hollunder, "Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic," *Journal of Automated Reasoning*, vol. 15, no. 1, pp. 41–68, 1995.

[3] G. Brewka, "Adding priorities and specificity to default logic," in *Proceedings of the European Workshop on Logics in Artificial Intelligence*. Springer-Verlag, 1994, pp. 247–260.

[4] W. Łukaszewicz, "Considerations on default logic: an alternative approach1," *Computational intelligence*, vol. 4, no. 1, pp. 1–16, 1988.

[5] T. Schaub, "On constrained default theories," in *Proceedings of the 10th European Conference on Artificial Intelligence*. John Wiley & Sons, Inc., 1992, pp. 304–308.

[6] A. Mikitiuk and M. Truszczynski, "Constrained and rational default logics," in *Procceedings of International Joint Conference on Artificial Intelligence (IJCAI'95)*, vol. 14, 1995, pp. 1509–1517.

[7] G. Brewka, "Cumulative default logic: In defense of nonmonotonic inference rules," *Artificial Intelligence*, vol. 50, no. 2, pp. 183–205, 1991.

[8] M. Gelfond, V. Lifschitz, H. Przymusinska, and M. Truszczynski, "Disjunctive defaults," *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, vol. 91, pp. 230–237, 1991.

[9] V. Marek, M. Truszczyński, and R. Reiter, *Nonmonotonic logic: context-dependent reasoning*. Springer, 1993, vol. 393.

[10] T. Schaub, S. Brüning, and P. Nicolas, "Xray: A prolog technology theorem prover for default reasoning: A system description," *Proceedings of the Thirteenth International Conference on Automated Deduction (CADE'96)*, pp. 293–297, 1996.

[11] P. Cholewinski, V. Marek, and M. Truszczynski, "Default reasoning system deres," in *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR'96)*, 1996, pp. 518–528.

[12] F. Baader and B. Hollunder, "Embedding defaults into terminological knowledge representation formalisms," *Journal of Automated Reasoning*, vol. 14, no. 1, pp. 149–180, 1995.

[13] V. Kolovski, B. Parsia, and Y. Katz, "Implementing owl defaults," DTIC Document, Tech. Rep., 2006.

[14] G. Brewka, *Nonmonotonic reasoning: logical foundations of commonsense*. Cambridge University Press, 1991, vol. 12.

[15] D. Etherington and R. Reiter, "On inheritance hierarchies with exceptions," in *Proceedings of AAAI Conference on Artificial Intelligence (AAAI'83)*, vol. 83, 1983, pp. 104–108.

[16] F. Horty, R. Thomason, and D. Touretzky, "A clash of intuitions: The current state of nonmonotonic multiple inheritance systems," *Proceedings of the 10th International Joint Conference on Artificial Intelligence (IJCAI'87)*, 1987.

[17] A. Hunter, "Using default logic in information retrieval," *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pp. 235–242, 1995.

[18] R. Studer, V. Benjamins, and D. Fensel, "Knowledge engineering: principles and methods," *Data & Knowledge Engineering*, vol. 25, no. 1, pp. 161–197, 1998.

Research on Bounded Nussbaum Gain Adaptive Method

Junwei Lei, Guorong Zhao,Yong Liang , Huali Wu , Shixing Wang and Xiuxia Yang

Department of Control Engineering, Naval Aeronautical and Astronautical University, Yantai, Shandong, China

Abstract - *For common Nussbaum gain method, there exists a disadvantage that the nonlinear gain of the controller is unbounded essentially, which will cause many weaknesses because of the high gain of the controller. A new concept, which is called bounded Nussbaum gain function, is firstly proposed in this paper. What is different from the common Nussbaum gain function is that it can reduce the gain of the whole controller. And even better than that, it can ensure the gain of the controller is bounded. So the gain of the whole system can be controlled as small as possible, which is meaningful for the realization of many controllers in many actual nonlinear physical systems. Also, the new Nussbaum gain function is applied in a one order control system to give a example. Finally, The stability of the control system with the new Nussbaum function is proved and the simulation is done to testify the rightness and effectiveness of the new method.*

Keywords: Unknown control direction; Nussbaum gain; Bounded function; Adaptive

1   Introduction

The Nussbaum gain method can deal with the unknown control direction problem, which is discussed in many papers recently[1-18]. But the use of common Nussbaum gain function also cause that the gain of the whole system will increase nonlinearly, so the common Nussbaum function is unbounded essentially[1-18]. Although the controller with high gain can increase the stability of the system, it also cause many problems such as: the high gain will make the system highly sensitive to the interference of noise, or it will make the system reach the saturation area earlier, or for some actual systems, the high gain will make the controller too difficult to be realized[19-21]. In one word, if the gain of a system is too high, the control method will lost its physical meaning(because it can not be realized). So it is only a mathematically meaningful method.

These weaknesses caused by high gain are fatal for control systems. For example, methods in paper [20] and [21] are very effective for many ideal mathematical models, but it still have many problems to be solved if it is applied in some real and practical objects such as missiles and other weapon systems. That it just because of the using of high gain observer ,which do brings some advantages, that may cause many bad and dangerous effect and even unknown risk. So it is meaningful for a responsible engineer to analyze some

characteristics of a controller such as boundedness. Some other factors such as the sampling period or system damping characteristics are also very important and they have close relationship with the stability of a nonlinear system. Unfortunately, it is too difficult to make clear all those relationships. So in this paper we only study the easy boundedness characteristic. We will try to research other complex factors in our future work.

To solve the unknown control direction problem, a increase sequence is skillfully constructed in [18] and it can take the place of the Nussbaum gain. Meanwhile, a switching mechanism is adopted to decide the time of gain switching based on a energy function. If the energy of the system is increase, then switching mechanism will change the gain of the controller into another bigger gain with an opposite sign according to the above increase sequence. The system is guaranteed to be stable and the same conclusion as the Nussbaum gain method is achieved through a different angle of study. In this paper, a new concept of bounded Nussbaum gain function is adopted to improve the increase sequence method. Finally, we use a simple one order system to show the effect of the proposed method. According to the simulation result, we can make a conclusion that not only the control law is simplified, but also the system is more stable. Most important of all, the gain of the whole system is guaranteed to be bounded. So the problems caused by high gain method in [18] will not appear with the bounded Nussbaum gain method.

2   Definition and lemma

In this paper, the concept of bounded Nussbaum gain method is proposed firstly. Then to draw a comparison between the common Nussbaum method and the bounded Nussbaum gain method, a simple one-order system is research by using the above two different methods. So the definition and advantages and weaknesses of the bounded Nussbaum gain method can be testified.

Definition 1:   A bounded function $N(\chi)$ is called a bounded Nussbaum-type function, if it has the following characteristics

$$\lim_{s \to \infty} \sup \frac{1}{s} \int_0^s N(x)dx \to \infty$$

$$\liminf_{s \to \infty} \frac{1}{s} \int_0^s N(x)dx \to -\infty$$

Definition 2: A bounded function $N(\chi)$ is called a bounded Nussbaum-K type function, if it has the following characteristics

$$\limsup_{s \to \infty} \frac{1}{s} \int_0^s N(x)dx > K$$

$$\liminf_{s \to \infty} \frac{1}{s} \int_0^s N(x)dx < -K$$

Lemma 1: If $N(\chi)$ is belonged to Nussbaum-1 type function, then $KN(\chi)$ is belonged to Nussbaum-K type function.

Lemma 2: Assume $V(t)$ and $\chi(t)$ are smooth function defined on the interval $[0, t_f)$, for $\forall t \in [0, t_f)$, $V(t) \geq 0$, and if the following inequality holds

$$V(t) \leq c_0 + \int_0^t (g(\tau)N(\chi)+1)\dot{\chi}d\tau$$

Where c_0 is a proper constant, $c_1 > 0$, q is a positive odd constant $g(\tau)$ is an time-varying parameter in an unknown closed interval, then $V(t)$, $\chi(t)$ is bounded in the closed interval $[0, t_f)$.

3   Bounded Nussbaum Gain Method

To make it simple, we take a one order system as an example to illustrate the bounded Nussbaum gain method. Considering the following one order system

$$\dot{x} = 3x + gu \tag{1}$$

Where g is a time-varying parameter in an unknown closed interval, we define a new variable as

$$z = x - x^d$$

Then we have

$$\dot{z} = \dot{x} = 3x + gu \tag{2}$$

Design a virtual control law as

$$u^d = -3x - k_z z \tag{3}$$

And design a common Nussbaum gain method and a bounded Nussbaum gain method respectively as follows:

$$u = -N(k)u^d \tag{4}$$

$$N_1(k) = k^2 \cos k \tag{5}$$

$$N_2(k) = K_b \cos k^{2/4} \tag{6}$$

It is obvious that if we choose $N(k)$ as $N_1(k)$, the control law is a common Nussbaum gain control law and if we choose $N(k)$ as $N_2(k)$, the control law is belonged to the bounded Nussbaum gain control law. So we have

$$\dot{z} = -k_z z - gN(k)u^d - u^d \tag{7}$$

Choose a Lyapunov function as

$$V = \frac{1}{2}z^2 \tag{8}$$

Define

$$\dot{k} = zu^d \tag{9}$$

Then we have

$$\begin{aligned} \dot{V} &= -k_1 z^2 - gN(k)u^d z - u^d z \\ &= -k_1 z^2 - (1 + gN(k))\dot{k} \end{aligned} \tag{10}$$

Do the integral computation on both side of the equation, we have

$$\begin{aligned} V(t) - V(0) &= -\int_0^t k_1 z^2 dt - \int_0^t (1 + gN(k))\dot{k}dt \\ &= -\int_0^t k_1 z^2 dt - \int_{k(0)}^{k(t)} gN(k)dk + k(0) - k(t) \end{aligned} \tag{11}$$

Then we got

$$\begin{aligned} V(t) &= -\int_0^t k_1 z^2 dt - \int_{k(0)}^{k(t)} gN(k)dk \\ &+ k(0) - k(t) + V(0) \end{aligned} \tag{12}$$

With the help of the characteristic of Nussbaum gain method, it is easy to prove the system is bounded and stable.

Now the stability of the one order system with common Nussbaum gain method and Bounded Nussbaum gain method

is proved , then the stability of using the Nussbaum-k type function is discussed as below.

First, we assume g is a bounded time-varying parameter in the close interval[-1/K,1/K], and we set $N(k)$ in Equ.(12) as a Nussbaum-K type function, then $k(t)$ can be proved bounded as follows.

We use the apagoge and assume $k(t)$ is unbounded. First, we assume $k(t) \to +\infty$, then according to the characteristic of $N(k)$, we have

$$\limsup_{s \to \infty} \frac{1}{s} \int_0^s N(x)dx > K$$

It means that there exists $k(t)$ such that

$$\int_{k(0)}^{k(t)} N(k)dk > Kk(t)$$

Meanwhile according to the mean value theorem of integration and the characteristic of Nussbaum gain function, there exists $k(t)$ such that

$$\int_{k(0)}^{k(t)} gN(k)dk > k(t) > 0$$

Then we get

$$V(t) \le -k(t) + V(0) < 0 \qquad (13)$$

So it is obviously contradict with a definition of Lyapunov function.

Then we assume $k(t) \to -\infty$,

$$\liminf_{s \to \infty} \frac{1}{s} \int_0^s N(x)dx < -K$$

Then there exists $k(t)$ such that

$$\int_{k(0)}^{k(t)} N(k)dk > -Kk(t)$$

Meanwhile, according to the mean value theorem of integration and the characteristic of Nussbaum gain function , there exists $k(t)$ such that

$$-\int_{k(0)}^{k(t)} gN(k)dk - k(t) < 0$$

Then we have

$$V(t) \le -k(t) + V(0) < 0 \qquad (14)$$

So it is also contradict with the definition of the Lyapunov function. Then the above assumption that $k(t)$ is unbounded does not hold.

Above all , we get $k(t)$ is bounded. And with the help of Baralat lemma, it is easy to prove that the system is stable.

4   Example and simulation

According to the above control law design, we choose $k_z = 12$ and set the desired value as $x^d = 5$, and the unknown control gain is defined as follows

$$g = \begin{cases} 1.5 & t \le 10 \\ -1.5 & t > 10 \end{cases}$$

The simulation result can be showed as Fig. 1 to Fig.4 by using the common Nussbaum gain method. And choose $K_b = 2$ and use the bounded Nussbaum gain method, the simulation result can be showed as Fig.5 to Fig.8.
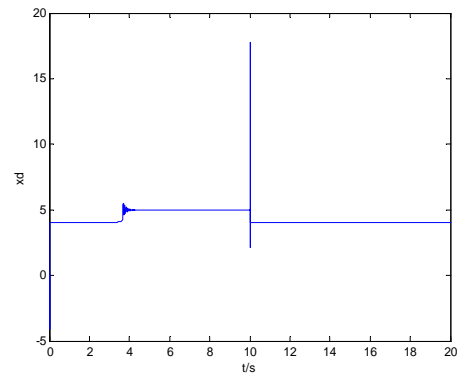
Fig. 1 Curve of tracking response
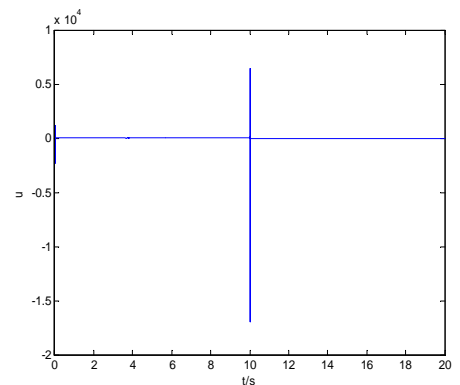
Fig. 2 Curve of control
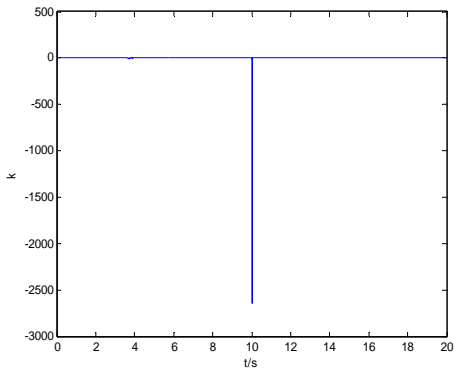
Fig. 3 Gain of K
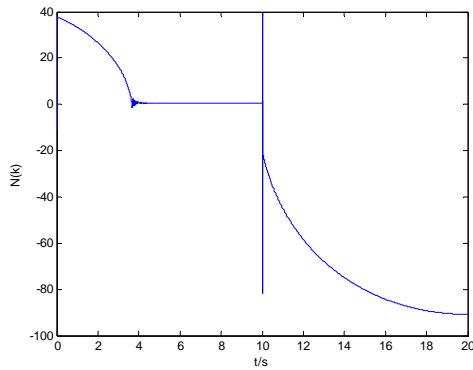
Fig. 3 Curve of $N(k)$
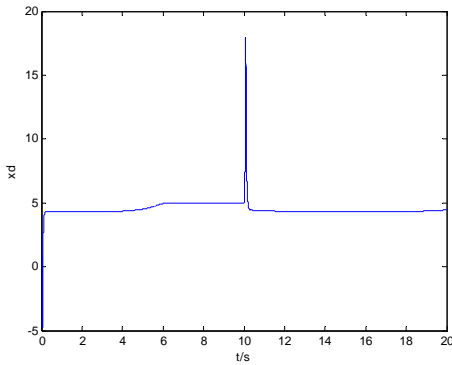
Fig. 5 Curve of tracking response
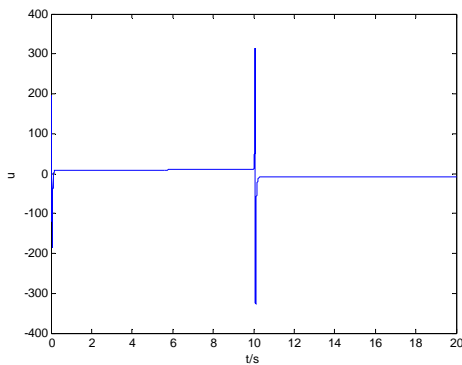
Fig. 6 Curve of control
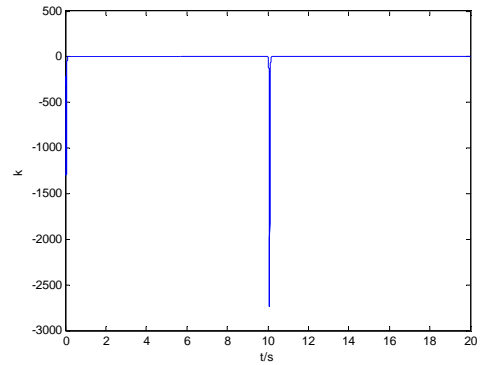
Fig7 Gain of K
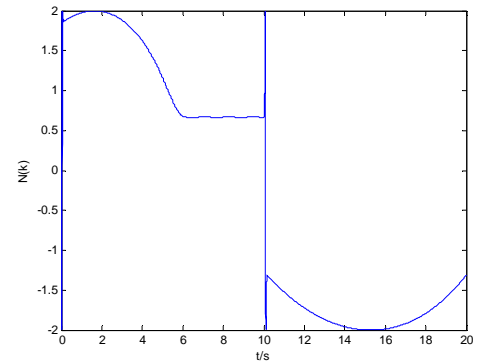
Fig. 8 Curve of $N(k)$

By the comparing of the above two methods, we can make a conclusion that the control signal is obviously reduced by using the bounded Nussbaum gain method. Also the Nussbaum gain $N(k)$ is reduced and meanwhile the gain of the whole system is reduced.

With further simulation, we find that when we set the unknown control direction as

$$g = \begin{cases} 3 & t \le 10 \\ -3 & t > 10 \end{cases}$$

Although we use the same simulation algorithm and simulation step, it is necessary to reduce the simulation step when the simulation program with common Nussbaum gain method reach the time of 10.021 second   but the simulation can be successfully completed without reducing the simulation step by using the bounded Nussbaum gain method.

5   Conclusions

In this paper, a novel concept, which is called Bounded Nussbaum gain function, is proposed. It is different from the common Nussbaum gain functions that the gain of the function is bounded. Also, it still satisfies the characteristics of Nussbaum gain method. So it is a totally special Nussbaum

gain function which can reduce the gain of the whole system. Also the stability of the controller with the bounded Nussbaum gain method is given in this paper. And the comparison in simulation between the common Nussbaum gain method and the bounded Nussbaum gain method obviously show the advantage and effectiveness of later method. We will try to research other complex factors such as system damping characteristics in our future work.

Acknowledgment

6   References

[1] Nussbaum, R. D. (1983). Some remarks on the conjecture in parameter adaptive control. Systems and Control Letters, 3(3), 243–246

[2] Ye, X. D., & Jiang, J. P. (1998). Adaptive nonlinear design without a prioriknowledge of control directions. IEEE Transactions on Automatic Control, 43(11), 1617–1621.

[3] Ge, S. S., Hong, F., & Lee, T. H. (2004). Adaptive neural control of nonlinear time-delay system with unknown virtual control coefficients.IEEE Transactions on Systems, Man, and Cybernetics-PartB: Cybernetics, 34(1), 499–516.

[4] Yan Li , YangQuan Chen When is a Mittag_Leffler function a Nussbaum function? Automatica 45 (2009) 1957-1959

[5] Ge, S. S., & Wang, J. (2003). Robust adaptive tracking for time-varying uncertain nonlinear systems with unknown control coefficients. IEEE Transactions on Automatic Control, 48(8), 1463_1469.

[6] T.P.Zhang , S.S.Ge Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs, Automatica 43(2007) 1021-1033

[7] Xu H J and Mirmirani M.Robust adaptive sliding control for a class of MIMO nonlinear systems [A].In: AIAA Guidance,Navigation,and Control Conference and Exhibit[C].Montreal, Canada, AIAA 2001-4168.

[8] Weisheng Chen, Adaptive NN control for discrete-time pure-feedback systems with unknown control direction under amplitude and rate actuator constraints, ISA Transactions 48 (2009) 304-311

[9] Yang C, Ge SS, Xiang C, Chai TY, Lee TH. Output feedback NN control for two classes of discrete-time systems with unknown control directions in a unifiedapproach. IEEE Transactions on Neural Networks 2008;19(11):1873_86.

[10] Jagannathan S, He P. Neural-network-based state-feedback control of a nonlinear discrete-time system in nonstrict feedback form. IEEE Transactions on Neural Networks 2008;19(12):2073-87.

[11] Charalampos P. Bechlioulis, George A. Rovithakis, adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems , Automatica 45(2009)532-538

[12] Tsinias, J. (1991). A theorem on global stabilization of nonlinear systems by linear feedback. System and Control Letters, 17, 357-362

[13] C. C. Fuh and P. C. Tung, "Controlling Chaos Using Differential Geometric Approach," Physical Review Letters, vol. 75, no. 16, pp. 2952-2955, 1995.

[14] M. Krstic, J. Sun, and P. V. Kokotovic, "Control of Feedback Linearizable Systems with Input Unmodeled Dynamics," Proc. of the 33rd Conference on Decision and Control, Lake Buena Vista, FL, pp. 1633-1638, 1994

[15] Qian, C., & Lin, W. (2002). Output feedback control of a class of nonlinear systems: A nonseparation principle paradigm. IEEE Transactions on Automatic Control, 47, 1710–1715

[16] Xudong Ye, Global adaptive control of nonlinear systems with unknown control directions overparameterization design, Journal of China Jiliang University   Jan, 2005 Vol.14,No,1

[17] Liu Yunguang, Output-feedback Adaptive Control for a Class of Nonlinear Systems with Unknown Control Directions, ACTA AUTOMATICA SINICA, Vol. 33, No. 12, 1306-1312

[18] KE Hai-sen, YE Xu-dong, Robust adaptive controller design for a class of nonlinear systems with unknown high frequency gains, Univ SCIENCE A 2006 7(3):315-320

[19] Khalil, H. K., & Saberi, A. (1987). Adaptive stabilization of a class of nonlinear systems using high-gain feedback. IEEE Transactions on Automatic Control, 32, 1031-1035

[20] Hao Lei,Wei Lin   Universal adaptive control of nonlinear systems with unknown growth rate by output feedback   Automatica 42 (2006) 1783-1789

[21] Hao Lei,Wei Lin   Adaptive regulation of uncertain nonlinear systems by output feedback: Auniversal control approach , Systems & Control Letters 56 (2007) 529– 537

Intelligent Sensorless Monitoring Dual System to Detect the Tool Condition in CNC Milling Machines

G. Mota-Valtierra[1], L.A. Franco-Gasca[1], H. Aguirre-Becerra[1] and **F. Ruiz-Vazquez[2]**
, [1]Facultad de Ingeniería, Universidad Autónoma de Querétaro, Querétaro, México
[2]Departamento de Mantenimiento, Instituto Tecnológico de Celaya, Celaya, Guanajuato, México

Abstract: *Tool state in the CNC milling machines will determine the product quality. An efficient tool state monitoring system will protect machinery from severe damages. For determining the state of the cutting tools in a milling machine there is a great variety of models in the industrial market, however those systems are not available for all companies because of their high costs and requirements of modifying the machining tool in order to attach the system sensors in the machine. This paper presents a sensorless intelligent dual system which classifies the cutters status in a CNC milling machine. The tool state is mainly determined through the analysis of the cutting forces drawn from the spindle motors currents. The tool classification is made by a Supervised SOM (Self Organized Maps), a MLP (Multilayer Perceptron) or both, achieving a reliability of 98%.*

Keywords: tool ware, wavelet transform, supervised SOM, Multilayer Perceptron, tool monitoring system.

1 Introduction

One of the main objectives of any company is to satisfy the customer needs by producing high quality products, optimizing costs by improving the manufacturing processes. In order to achieve the quality specifications it is important to eliminate variations during the production processes. For manufacturing companies the use of on line tool condition monitoring systems is essential in order to detect either breakage or tools ware to avoid poor quality production pieces due to the state of cutting tools and even preventing damage of machines.

Neural Networks (ANN) is one of the most common and reported methods used in monitoring systems that classifies tool state, it is widely used because of its adaptive learning, self-organization, fault tolerance and real-time operation, providing good solutions for classification or decision making problems. Examples of ANNs applied to tool condition classification are found in [1], [2], [3] and [4].

Literature suggests that exists a correlation between cutting forces (static and dynamic) and tool wear [5], and those parameters may be studied in several forms, such as the based on changes of friction force between cutting tools and workpieces [6]. In several works it has been decided to analyze the cutting forces in order to determine the level of tool wear [7], [8], [2]

and [3]. In order to evaluate cutting forces [9] and [10] developed simulation models that determine cutting forces with more precision than analytical models due to the application of Multi Layer Perceptron (MLP).

The use of sensors is common, however, their application is limited because of the narrow operating range defined by the manufacturer, and usually system designs are made considering specific work conditions, which do not allow adjustments of manufacturing operations. In many occasions it is necessary to make machine modifications to place sensors. All those negative aspects are not presented in the proposed system because of its sensorless operation. It is presented the proposal of an intelligent system, with low cost and easy incorporation to the original process to classify physical condition of the cutting tool in a milling machine, helping to prevent defects in the working pieces and avoiding severe damage on the machine tool.

This paper is organized as follows: Section 1 introduction, Section 2 is related to the monitoring system, and finally in Section 3 conclusions are presented.

2. Development

2.1. Monitoring System.

A retrofitted CNC milling machine, model FNK25, with a head tool of two carbide inserts was used for testing the intelligent classifier. The correct set up of cutting parameters is an important step in the milling process. Parameters selection depends on the material hardness, type of cutter and work piece finish, among others. This choice will determine whether or not the final product meets the quality specifications (dimensions, finish, etc.). The stages that compose a milling process are shown in figure 1.

The milling parameters are shown in Table 1. The milling process was made on ASTM-4130 steel, and using cutters of different states such as new (good conditions), worn (with several degrees of wear) and broken.
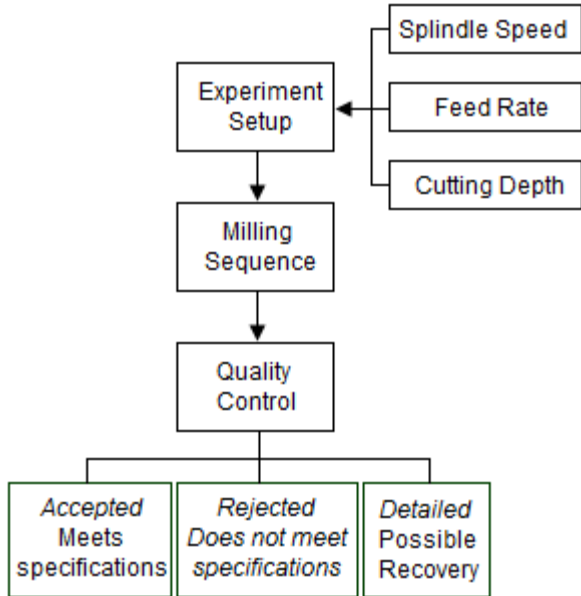
Figure 1. Stages of a milling process

Table 1. Milling parameters

Parameter	Value	
Spindle speed	300 – 450	rpm
Cutting depth	1 – 1.5	mm
Feed rate	100 - 120	mm / min

Values of spindle speed, depth of cut and feed rate were varied between the ranges shown in table 1 for each case. Signals from the motor spindle driver were acquired to determine the cutter status; these signals are a direct representation of the cutting forces. A Tektronik MSO4000 oscilloscope was the instrument used to acquire the signals. Finally, to obtain the neural networks inputs, a features extraction from the acquired data was made by digital signal processing techniques. Figure 2 represents the general system stages.
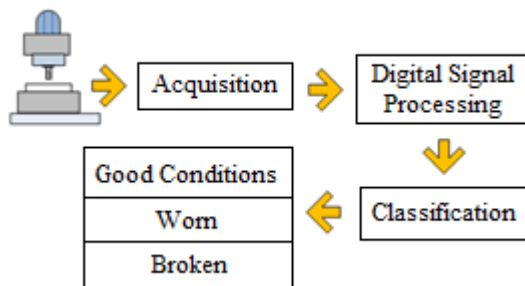
Figure 2. Monitoring system stages

2.2. Data Acquisition.

As mentioned before, the cutting forces will be the main parameter to be analyzed [11]; one of the points where is possible to acquire these signals is the spindle motor. Thus for avoiding the use of sensors, in this research the motor driver is proposed as the data source. The original signal presents the cutting force as its main component; however, it is important to mention that signals from the servo driver have severe noise interference by the ball-screw and the switching noise due to the associated digital systems [12]. Because of this, previously to the digital processing is necessary to filter the spindle current signals. Figure 3 shows the signal processing system, which was used to obtain the cutting forces.
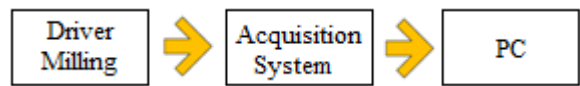
Figure 3. Experiment setup

For noise elimination, the original signal was filtered using a band-pass filter. A Finite Impulse Response (FIR) digital filter was chosen because it has a linear phase response. Table 2 presents the parameters of the applied filter; this was designed using the filter design and analysis tool from MatLab.

Table 2. Filter parameters

Filter characteristics	
Filter type	Bandpass
Design method	Kaiser window
Sampling frequency	6250 Hz
Order Filter	20
Cutoff frequencies	fc1 20 Hz
	fc2 138 Hz

Figure 4 displays an unfiltered signal (a) and its corresponding filtered signal (b). The filtering process removes noisy components and preserves the embedded cutting force.

To consolidate the classification process a data compression procedure was performed. This was done by the Wavelet Transform (WT), which implements a mapping of the time-domain to a time-scale representation, preserving the temporal aspect. The figure 5 shows a signal with different compression levels by applying a Daubechies-5 Wavelet function, it can be seen that those levels have the same pattern but a different resolution. The maximum transformation level is determined by the desired resolution, for this study the fifth level was selected, this allows to achieve a data reduction from 1024 to only 32 points per sample.
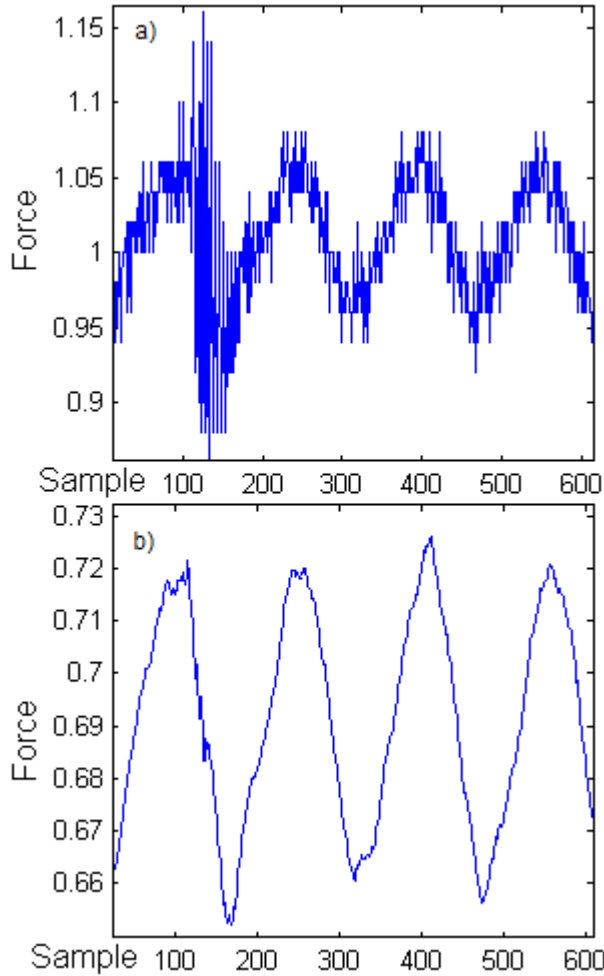
Figure 4. (a) Unfiltered signal. (b) Filtered signal.
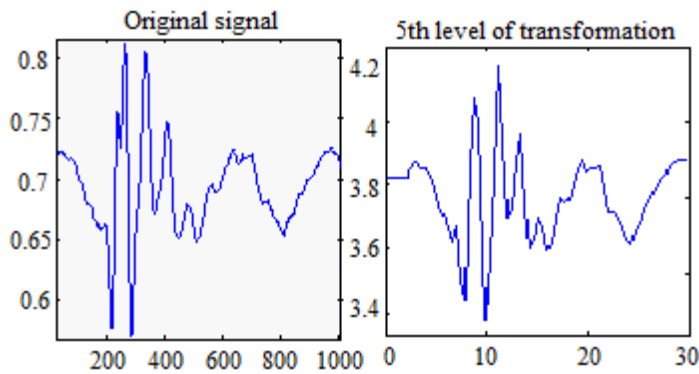
Figure 5. Data compression levels. The Y-axis represents the cutting force and the X-axis is the sample number

2.3. Intelligent Classification.

In order to select the optimal networks for tool state classification, several Multi Layer Perceptron (MLP) type networks with supervised training were tested. Self-Organizing Maps (SOM) with supervised and unsupervised training were supervised, this represents a significant variation because SOM networks are usually unsupervised trained. Some of the tested MLP networks included [3, 10, 10, 3], [3, 8, 8, 3], [3, 8, 12, 3] and [3, 8, 10, 3] structures. Figure 6 shows one of the MLP tested. The activation function is a sigmoid function and learning rate of 0.2.

It was decided to test a SOM network as classifier, due to its low sensitivity to noise, it is an appropriate tool to classify this kind of signals. Some of the analyzed structures were [4, 4], [4, 8] and [8 8], these were trained using both supervised and unsupervised learning. The supervised training was made adding a supervisor agent, which is an array of [N 1], where N is the number of existing classes. Figure 7 shows a [4, 4] SOM network with a neighborhood of 1.
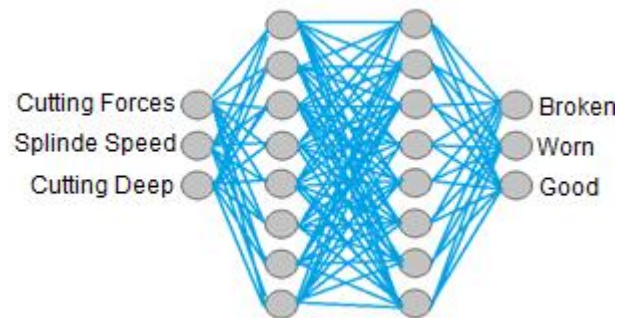
Figure 6. Multilayer Perceptron [4 8 8 3].
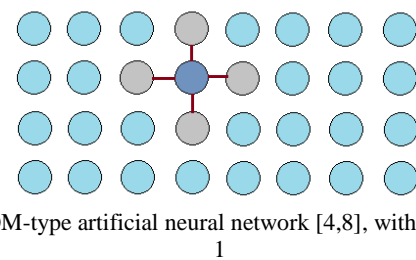
Figure 7 SOM-type artificial neural network [4,8], with neighborhood 1

The classifier was tested using two different kinds of ANNs with different size and training types. There are differences in the networks performance, but they are not significant. As summary, the Table 3 shows the achieved error during the training of the MLP networks and Table 4 shows the error using ANN SOM supervised.

Table 3. Error of MLP networks

Cutting depth (mm)	Feed rate (mm/min)	Spindle speed (rpm)	Epoch	Neuron in hidden layers	Error
1	100	300	1000	[8,8]	0.001397
1	100	450	1000	[8,8]	0.001298
1	120	300	1000	[10,10]	0.00099
1	120	450	1000	[10,10]	0.00099
1.5	100	300	1000	[8,10]	0.00099
1.5	100	450	1000	[8,10]	0.00099
1.5	120	300	1000	[8,12]	0.00099
1.5	120	450	1000	[8,12]	0.00099

Table 4. Error of MLP networks

Cutting depth (mm)	Feed rate (mm/min)	Spindle speed (rpm)	Epoch	Structure	Error
1	100	300	1000	[4,4]	0.001197
1	100	300	1000	[4,8]	0.001318
1	100	300	1000	[8,8]	0.00099
1.5	120	450	1000	[4,4]	0.00199
1.5	120	450	1000	[4,8]	0.00099
1.5	120	450	1000	[8,8]	0.00099

Figure 8 shows the way as the error decreases, there is not an important difference among the obtained errors when the number of neurons in the hidden layers is bigger than 8. The neural network that is considered as suitable for using in the proposed intelligent classifier should have at less eight neurons in the hidden layers. For validating the network performance two types of inputs were tested, signals previously used during the training and signals not used for the training process. Figure 9 shows the convergence of the ANN SOM supervised and unsupervised.

Figure 9. Error during neural networks training

For testing the monitoring system we can use one, two or both ANNs (MLP and SOM supervised). Some of the worn cutters were used during the training. To guarantee a good classification system it must identify either a broken or worn cutter. Figure 10 shows some results of correct identifications.

Figure 8. Error during neural networks training
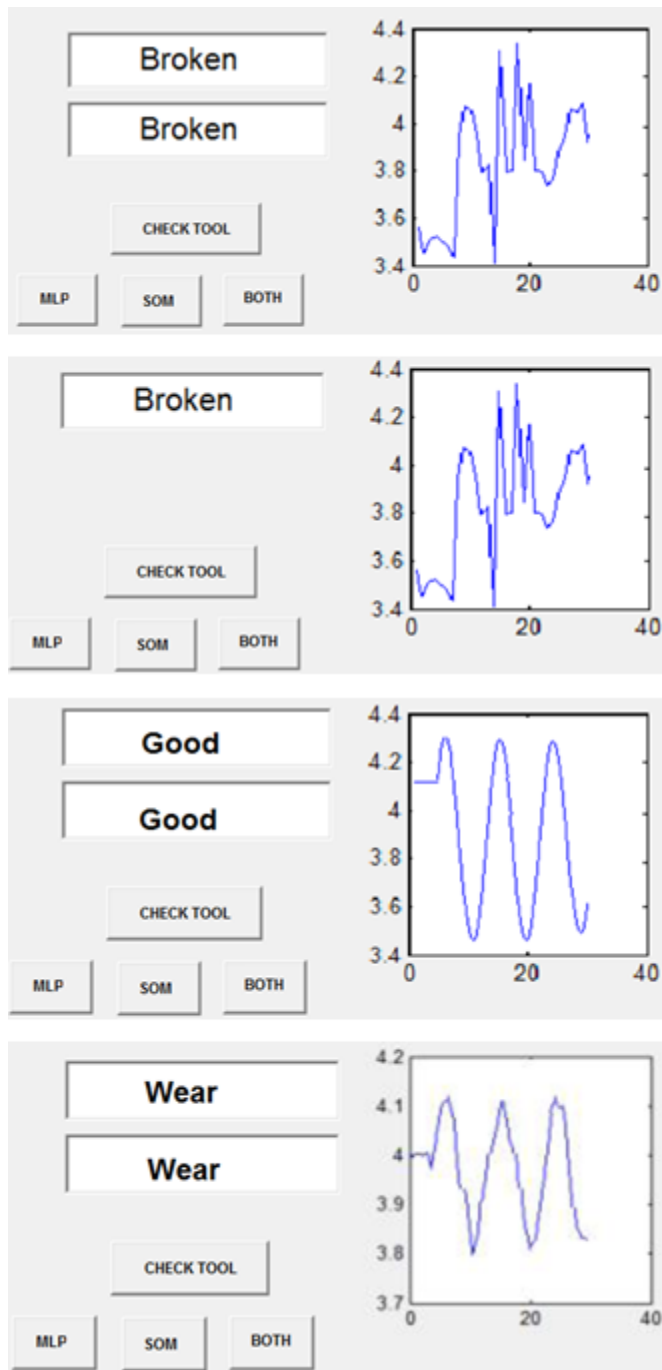
Figure 10. Status classification of tool using two and one ANN.

3. Conclusions

Cutting force variations have been correlated to the tool wear by using one or two Artificial Neural Network, as a consequence the correct classification of the cutting tool condition is achieved. The ANNs approach had made possible the online and fault tolerant monitoring of the tool, besides the system presents the advantage of not having to stop the machine for knowing the tool condition. From test using both training types, the SOM supervised obtained a faster convergence than SOM unsupervised. The presented tool condition monitoring system is sensorless, thus the machinery will not be modified if the system is attached to its structure.

There is not a significant difference in the achieved error when the MLP networks have a size more than [8, 8] neurons in the hidden layer, or when del SOM has an structure [8,8]; however, computationally it represents a considerable difference in resources consumption, for that reason is not appropriate to comprise more neurons. So this is the suitable size for a network which is going to be considered as classifier in the proposed system.

The proposed neural network is able to classify breakage levels greater than 0.3 mm with a confidence level of 98%, with the same confidence level it also determines the good condition of the tool. The wear tool can be resolved with an efficiency of 94% when the wear is greater than 0.25 mm. Its reliable confidence level avoids the damage to the machinery, the tool and the piece, which is the main objective of a tool condition monitoring system. In the worst case if the monitoring system fails the workpiece will need to be re-worked with a new cutting tool, since the damage will be just a piece of a lower quality finish. Future work will try to identify two levels of wear tool, in addition to the breakage and the good working conditions.

The maximum compression level of the signal was achieved with five levels of data processing. To verify that the original signal may be recovered, an inverse wavelet transformation was made, verifying that the signal is completely recoverable.

References

[1] K. Bhattacharyya, K. Pal Surjya and K. Patra, "Artificial neural network based prediction of drill flank wear from motor current signals", Applied Soft Computing, Vol. 7, pp.929–935, 2007.

[2] E. Kuljanic, M. Sortino and G. Totis, "Development of an intelligent multisensory chatter detection system in milling", Mechanical Systems and Signal Processing, Vol. 2, pp.1704–1718, 2009.

[3] L.N. Lopez de La calle, M.L. Penalva and A. Rivero, "Tool wear detection in dry high-speed milling based upon the analysis of machine internal signals", Mechatronics, Vol. 18, No. 10, pp.627–633, 2008.

[4] A. Mehrabian and B. Menha Mohammad, "A real-time neuro-adaptive controller with guaranteed stability", *Applied Soft Computing*, Vol. 8, pp.530–542, 2008. [7] P.J. Arrazola and K. Jemielniak, "Application of AE and cutting force signals in tool condition monitoring in micro-milling", CIRP Journal of Manufacturing Science and Technology, Vol. 1, pp.97–102, 2008.

[5] J.C. Romero-Troncoso, G. Herrera-Ruiz, R. J. Jáuregui-Correa and I. Terol-Villalobos, "Driver current analysis for sensorless tool breakage monitoring of CNC milling machines", International Journal of Machine Tools & Manufacture, Vol. 43, pp.1529–1534, 2003.

[6] I. Fernández de Bustos, A Lamikiz, L. N. López de La calle and J. A. Sánchez, "Recording of real cutting forces along the milling of complex parts", Mechatronics, Vol. 16, pp.21–32, 2006.

[8] P. Bhattacharyya, S. Mukhopadhyay and D. Sengupta, "Cutting force-based real-time estimation of tool wear in face milling using a combination of signal processing techniques", Mechanical Systems and Signal Processing, Vol. 21, pp.2665–2683, 2007.

[9] F. Cus and U. Zuperl, "Tool cutting force modelling in ball-end milling using multilevel perceptron", Journal of Materials Processing Technology, Vols. 153–154, pp.268–275, 2004.

[10] A. Kurt, "Modelling of the cutting tool stresses in machining of Inconel 718 using artificial neural networks", Expert Systems with Applications, Vol. 36, No. 6, pp.9645–9657, 2009.

Ackowledgments.

THE IMPORTANCE AND NEED FOR DIGITAL FORENSIC INVESTIGATIVE FRAMEWORK

Inikpi O. Ademu[1] Dr Chris O. Imafidon[2]

[1]School of Architecture, Computing and Engineering, University of East London, London, United Kingdom
[2]Formerly, Head of Management of Technology Unit, Queen Mary University of London, Currently Senior Academic, School of Architecture, University of East London, London, United Kingdom

Abstract - *Since digital devices such as computers are vulnerable to attack by criminals, digital forensics is very important. Understanding digital forensic procedures will help to capture vital information which can be used to prosecute an intruder that compromises a digital devices or network. A large amount of information is produced, accumulated, and distributed via electronic means. It is necessary for forensic experts to increase their abilities to gather evidence from digital devices. Also, implementing security mechanism and guidelines to digital forensic investigative process is needed to correctly analyze evidence. The advancement of the digital forensic investigation requires a new design, improved mechanism and processes. Forensic experts are faced with growth in data. Huge amount of data has expanded and grown in recent years and attempts to consume the storage space available. There is need to understand the importance of a digital forensic investigative framework.*

Keywords: autonomous coding, security mechanism, security threat, data processing, investigative framework

1. Introduction

Digital forensic has been defined as the use of scientifically derived and proven methods toward the identification, preservation, collection, validation, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations (Reith et al, 2002). This definition covers the broad aspects of digital forensics from data acquisition to legal action. It describes digital forensics as a synonym for computer forensics, and defines it as the use of scientific methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation, and presentation of digital evidence derived from digital sources for the purpose of facilitation helping to anticipate unauthorized actions exposed to be disrupting intended operations. The influence of information technology is pervasive in our private and professional lives. The use of the internet, email and chat groups has changed the ways in which interaction is carried out as a society. This important change is also apparent in organizations and the economy. The information technology has become the foundation for communications, banking, transformation etc. However, this has been capitalized on by the criminal in the society. Today, traditional criminal investigations need to be supported with digital evidence collection tools and techniques. This need has led to the development of digital forensic science and specifically computer forensic. The aim of this research is to study the main activities or processes necessary for digital forensic investigation and to examine existing digital forensic models and the importance and need for an investigative framework. The research also identifies the need of a security mechanism that contributes to a good security level of digital forensic investigation process and as a measurement tool for integrity of digital object.

2. Digital Evidence

Digital evidence is by nature fragile. It can be altered, damaged or destroyed by improper handling or improper examination. It is easily copied and modified, and not easily kept in its original state, precaution should be

taken to document, collect, preserve and examine digital evidence (Carrier, 2003). Buttressing this point is looking at one of the research carried out by (Sommer, 2009) he emphasized that data from computer can be accurately preserved and presented in court and like every other evidence digital evidence must be authentic, accurate, complete and convincing to juries, digital evidence is different from every other evidence in that it can change from moment to moment within a computer and along a transmission line, it can easily be altered without trace and can be changed during evidence collection. The main problem is how do expert measure the reliability of digital evidence?

Digital evidence is a data of investigative value that is stored on or transmitted by a digital device. Therefore digital evidence is "hidden" evidence in the same way that Deoxyribonucleic Acid (DNA) or fingerprint evidence is hidden. In its natural state, digital evidence cannot be known by the content in the physical object that holds such evidence. Investigative reports may be required to explain the examination process and any limitation (Pollitt, 2007). Digital forensic techniques are used primarily by private organisations and law enforcement agencies to capture, preserve and analyze evidence on digital devices. Digital evidence collected at a crime scene has to be analyzed and connections between the recovered information need to be made and proven. The search for digital evidence is a tedious task that consumes time. An extremely large amount of evidence needs to be processed in a very limited time frame which leads to delay in processing schedules.

3. Digital forensic investigation process

The investigative process is structured to encourage a complete, accurate investigation, ensure proper evidence handling and reduce the chance of mistakes created by preconceived theories and other potential pitfalls. This process applies to criminal investigations as well as public and private inquiries dealing with policy violations or system intrusion. Investigators and Examiners work hand in hand in a systematic and determined manner in an effort to present accurate and reliable evidence in the court. While in the court evidence are handed over to the prosecutors who scrutinize the findings and decide whether to continue or discontinue the case.

A good investigative framework should aim at providing to the investigators and examiners, structured and precise logical flow of event that collectively seeks to provide:

- Acceptance: steps or process and methods have earned professional consent

- Reliability: methods used can be trusted to support findings.

- Usability: the process can be repeated and applied by all regardless of time and place.

- Flexibility: the process or component is easily modified for use.

- Integrity: the state of evidence proven to be unaltered.

- Documentation: the process is recorded from start to finish for testimonial evidence.

4. Existing digital forensic model

The first DFRWS was held in Utica, New York (2001). The goal of the workshop was to provide a forum for a newly formed community of academics and practitioners to share their knowledge on digital forensic science. The audience were military, civilian, and law enforcement professionals who use forensic techniques to uncover evidence from digital sources. The group created a consensus document that drew out the state of digital forensics at that time. The group agreed and among their conclusions was that digital forensic was a process with some agreed steps. They outline processes such as identification, preservation, collection, examination, analysis, presentation and decision (Palmer, 2001). Some of their identified matrixes were identified by the group as fundamental processes, although many will debate the forensic nature of each step of the process. This can be called a comprehensive or an enhanced model of the Department of Justice (DOJ) model because it was able to cover stages that were not covered in any previous model, such as presentation stage. The main advantage of DFRWS is that it is the first large-scale organisation that is lead by academia rather than law enforcement, this is a good direction because it helps define and focus the direction of the scientific community towards the challenge of digital forensic, but the DFRWS model is just a basis for future work.

Reith et al (2002) examined a number of published models/framework for digital forensics. The basis of this model is using the ideas from traditional (physical) forensic evidence collection strategy as practiced by law enforcement (e.g. FBI). The authors argued that the proposed model can be term as an enhancement of the DFRWS model since it is inspired from it. Using this

model, future technologies and the technical details required to forensically analyse them can be instantiated to provide a standard methodology for providing electronic evidence (Reith et al, 2002). This will improve the science of forensic because it involves a basis for analysing new digital technology while at the same time provide a common framework for law enforcement and the judicial system to feasibly work within a court of law.

Carrier and Spafford (2003) proposed a model, which the authors provide a review of previous work and then map the digital investigative process to the physical investigation process. The model known as the Integrated Digital Investigation Process was organised into five groups consisting of 17 phases. End to End Digital Investigation adopted by (Stephson, 2003) consist of 9 activities. It combined an extended digital forensic investigation process. The model takes into account the source of the incident, destination of the incident and the intermediate devices along the path through the network

Baryamueeba and Tushaba (2004) suggested a modification to Carrier and Spafford's (2003) Integrated Digital Investigation Model. The model is known as Enhanced Digital Investigation Process, the authors described two additional phases which are traceback and dynamite which seek to separate the investigation into primary crime scene (computer) and secondary crime scene (the physical crime scene). The goal is to reconstruct two crime scenes to avoid inconsistencies.

Ciardhuain (2004) argues that the existing models are general models of cybercrime investigation that concentrate only on processing of evidence in cybercrime investigation. The author proposed an extended model for cybercrime. It provides a good basis for understanding the process of cybercrime investigation, tackling certain activities such as presenting the information flow in an investigation and captures its full scope and not just processing the evidence. Even though the model was generic, it concentrated on the management aspect. The author argues that the available models are generic model of cybercrime investigation focusing on investigative process such as gathering, analysing and presenting the evidence. The model is designed to assist public and corporate forensic investigations. The model assists in the development of model investigative tools. The investigative tools for conducting investigation are not provided.

Freiling and Schwittany (2007) proposed a model for the purpose of introducing a new process framework to investigate computer security incidents and its aim is to combine the two concepts of incident response and computer forensic to improve the overall process of investigation. The framework focuses generally on the analysis of digital evidence. Perumal (2009) proposed a model that clearly defines that the investigation process will lead into a better prosecution as the very most important stages such as live data acquisition and static data acquisition has been included in the model to focus on fragile evidence. Polli et al (2010) proposed a generic model for network forensic analysis based on various existing digital forensic model, it covers tools, process model and framework implementation. This was specifically for the network based investigation.

The Systematic Digital Forensic Investigation Model proposed by (Agawal et al, 2011). This model has been developed with the aim of helping forensic practitioners and organizations for setting up appropriate policies and procedures in a systematic manner. The proposed model in this paper explores the different processes involved in the investigation of cyber crime and cyber fraud in the form of an eleven stage model. The model focuses on investigation cases of computer frauds and cyber crimes. The application of the model is limited to computer frauds and cyber crimes.

The Relational Reconstruction model was proposed by (Ademu et al, 2011). The model identifies the need for reconstrcution and interaction. The investigator should have consistent interaction with all resources for carrying out the investigation. Knowing the need of the victim and determined to meet the need is important. Better case goal can be defined. Optimal interaction with tools used by an investigator is very important. Tools need to be used by people who knows how to use them properly following a methodology that meets the legal requirement associated with the particular jurisdiction. Investigators need to have the patience, to stay on the target and have to learn any new techniques while performing an investigation. Very little testing has been formalized in this field for the specific need of digital forensic, investigators wishing to be prudent should undertake their own testing methods and this should be a normal part of the process used in preparing for legal matters and this should also meet the leal requirement of the jurisdiction. The model also help capture the expertise of investigation as a basis to the development of advanced tools incorporating techniques such as identifying the Visual Basic Integrated Development Environment with a set of rich features which are likely to be required for developing tools that can assist digital investigators during digital forensic investigation.

5. Why the need for investigative framework

Digital evidence is admissible in court as long as the process used to produce the evidence is known to produce reliable results. According to Kruse II (2002), the basic forensic methodology known as the three A's is evidence must be acquired without altering or damaging the original, investigator must authenticate that recovered evidence is the same as the originally seized data and data must be analyzed without modifying it. Digital evidence must not be modified or damaged during any part of the investigation process. Hash sums should be calculated on collected digital evidence data, and also the source of the evidence and compared to ensure the authenticity and integrity of the data.

An investigative framework should provide a process for conducting a digital forensic investigation. There are multiple factors complicating the investigative process. The more clearly the investigative process is defined, the more likely an investigation will be successful if such process is used for investigation (Ademu et al, 2012). An investigative framework, properly thought out and constructed would give a step by step process for conducting an investigation into a suspected digital device. A clear and structured process will allow investigators and examiners determine early in the investigation that an attack has occurred and it can also lead them to a final conclusion Casey (2002). It is important to know that this does not imply that all such conclusions are successful. A worrying large percentage of investigation ends with the conclusion that the victim was attacked, but the source of the attack cannot be determined.

Casey (2004) discussed that the U.S. Supreme Court provides certain criteria in the Daubert vs. Merrel case that may be used as guidelines by courts to determine whether or not evidence is admissible in court. Conventional applications in the jurisdiction therefore have to adhere to the requirements stipulated by the Daubert standard to allow the evidence they collect to be admissible in court. Few investigators have the time and skill to evaluate and analyze their chosen tools to determine whether or not it obeys the rules of the criteria stated by the Daubert standard. Even though the tools obeys the rule to the criteria and perform well in a trusted environment, they may give inconsistent results in an untrustworthy environment Casey (2004). This is because some software applications rarely contain all the operating logic needed to perform basic functionality that can be supplied by external drivers or the operating system, the application rely rather on libraries and drivers may be compromised to produce results that are inconsistent with the digital evidence.

6. Impact of security threat on investigative framework

Since digital devices such as computers are vulnerable to attack by criminals, security inclined digital forensics is very important. Understanding digital forensic procedures and techniques will help to capture vital information which can be used to prosecute an intruder that compromises a digital device or network. Also, deciding on the specific tools for computers or other digital devices that are needed to correctly analyse evidence is crucial. There is great need for information security guidelines integrated in the investigation process that can assist in establishing digital evidence.

Chaikin (2007) raised the issue of reliability as a limitation of digital evidence. The author explained that cyber attackers are rarely held accountable for their illegal actions, and one explanation for the lack of successful prosecution of cyber attackers is damage on digital evidence. Digital evidence is different from evidence created, stored, transferred and reproduced from a non-digital format. Digital evidence is temporary (short-lived) in nature and can easily be manipulated and this characteristic of digital evidence raises issues as to its reliability. Since courts have become more familiar with the vulnerabilities of digital evidence, they scrutinise the reliability of digital evidence with specific emphasis on its content and processes of gathering the digital evidence. The defence counsels increasingly challenge both the admissibility and the weight of digital evidence. There is need for improved competencies in handling digital evidence. As the role of information technology expands, so has the importance of information security. The increase in the use of information technology has led to the appearance of new areas of vulnerability. Although more time and effort is being put into developing security products, the potential consequences of security failure are also growing.

As computer and network play an increasingly important role in businesses and for digital investigation, so have been the dangers of malicious software. Also, as the computing industry has grown larger, the market has become dominated by a few leading brands. A malicious program that exploits bugs or vulnerabilities in one of those market leaders is a threat to thousands of organisations (Champlain, 2003). Since digital devices such as computers are being used by digital investigators, there is need for investigators to understand the dangers of malicious code or the practices that carry a risk of digital evidence being contaminated and leading to unacceptability in the court or during internal hearings.

Infection of a network by malicious code can usually be traced back to some instances of careless or risky behaviour, for instance a floppy disk or other storage device infected with a virus can infect an investigation workstation. Once the workstation is infected the virus can spread either by itself or with human assistance to other workstation in the network. If a computer is infected with a worm for example, it may automatically send infected mail attachment to all of the addresses in the address book. Malicious code is an increasing problem because the increasing complexity of programs has made more susceptible to attack. Also a few program suites dominate the market, so viruses can spread very quickly. Malicious code can enter the local network through a single workstation, but most attacks come through a network gateway. The network can come under attack in different ways, but one common way is importation of executable code into the network. Digital devices connected to the internet are also vulnerable to attack through their internet connections.

One of the most important elements of information security is protecting the computer network and users from malicious code. The internet is infested with viruses and worms that can do a lot of damage to the computers and most importantly to the data they store. Increasingly is the continuous experience of cybercrime threats. The growth of threats indicates that more types of threats will be faced. Also, there will be continuous need for countermeasures of security to control them. The nature of this countermeasures might be technological operational. The following are important security measures required for digital forensic investigation process:

6.1 Application and Content Based Security

Security technologies have important role in securing the systems and applications supporting the major aspect of digital forensic investigation processes. Technologies such as Antivirus, Firewalls, Cryptography and Security Protocol (Anderson, 2001) contribute to the success of the digital forensic investigation process by providing those involve in digital investigation high trust for the digital evidence collected. In a case of not having all, some or any of the security measures will have a negative impact and can be considered a threat on the digital evidence. It is important that all attachment for example email attachment are scanned before being opened. Attachment may contain code that will infect other files, care must be taken in working on attachment and executable files.

6.2. Applying Software Security Patches

It's common for security flaws to be identified in software that has been released for sale to the public.

Some of these flaws can leave the system open to attacks by malicious code (Champlain, 2003). Software vendors offer free patches to eliminate known flaws. Digital Investigators should ensure that all relevant software is updated with patches as soon as they are made available. Occasionally, a number of patches may be issued together as a service park.

6.3. Avoid Using Unsecured Window Shared Folders

The use of unsecured Windows shared folders should be avoided. Some worms for instance worms can spread across the network through such shares, infecting all hosts that have unsecured shared folders.

6.4. Operational Procedure

A good security guideline will have incident response process, security operational procedures.

During the investigation, it was important to ensure that all the investigation process were conducted ensuring that the security mechanism were available and all necessary activities conducted. It is important for any investigation to be able to prove the integrity of digital evidence. One way that this can be achieved is for investigators to ensure that security mechanism and policies were followed during the investigation, And also the security mechanisms must be integrated in the investigation process to ensure that digital information are secured from security threats and vulnerabilities that can alter digital object such as the computer and its information hence altering the digital evidence.

7. Conclusions

Digital evidence must be precise, authenticated and accurate in order to be accepted in the court. Digital evidence is fragile in nature and they must be handled properly and carefully. Detailed digital forensic investigative processes provide important assistance to forensic investigators in establishing digital evidence admissible in the court of law.

There is need for a security mechanism that contributes to a good security level of digital forensic investigation process and as a measurement tool for the integrity of digital object. Malicious code or software is an increasing problem in digital forensic investigation because of the increasing complexity of programs and toolkits deployed has made them more susceptible to attack. Also a few program suites dominate the market, therefore viruses can spread very quickly. Malicious program can enter the

network through a single workstation, but most attacks come through a network gateway.

This research identifies that the digital forensic community needs a security mechanism that will contributes to a good security level for digital forensic investigation process. The security requirement are identified as application and content based controls that recommends the need for Antivirus software, firewall, cryptography and security protocol for the success of digital forensic investigation process. It's common for security flaws to be identified in software that has been released for sale in the public. Some of these flaws can leave the system open to attacks by malicious program, for this purpose the second step of security requirement is applying software security patches. It is recommended for relevant software to be updated with patches as soon as they are made available. Due to the nature of digital object, they can be easily contaminated by malicious program. It is recommended that the use of unsecured Windows Shared folders should be avoided. It is also recommended as good practice for digital forensic investigators to follow operational procedures during investigation.

The digital forensic community needs a structure framework as identified in this research for rapid development of standard operational procedures that can be peer – reviewed and tested effectively and validated quickly. Digital forensic practitioners can benefit from the iterative structure provided in this research to build a forensically sound case and also for the development of consistent and simplified processes of digital forensic investigation that can be a guideline for standard operational procedure and a framework for developing future technology in the digital forensic investigation.

8. Acknowledgement

9. References

[1] Ademu, I. Imafidon, C. Preston, D., (2012) Intelligent

Software Agent applied to Digital Forensic and its

Usefulness Vol. 2 (1) Available (online):

http://interscience.in/IJCSI_Vol2Iss1/IJCSI_Paper_21.pd

f Accessed on 10th April 2012

[2] Ademu, I. Imafidon, C. I. Preston, D. (2011) A New Approach of Digital Forensic Model for Digital Forensic Investigation Vol. 2, (12) Available (online): http://thesai.org/Downloads/Volume2No12/Paper%2026-A%20New%20Approach%20of%20Digital%20Forensic%20Model%20for%20Digital%20Forensic%20Investigation.pdf Accessed 28th April 2012

[3] Agawal, A. Gupta, M. Gupta, S. Gupta, C. (2011) Systematic digital forensic investigation model Vol. 5 (1) Available (online): http://www.cscjournals.org/csc/manuscript/Journals/IJCSS/volume5/Issue1/IJCSS-438.pdf Accessed on 30th April 202

[4] Anderson, R. (2001) Security Engineering: A guide to building dependable distributed systems. New York: Wiley

[5] Baryamureeba, V. Tushabe, F. (2004) The Enhanced digital investigation process (2004) Available (online): http://www.dfrws.org/2004/bios/day1/tushabeEIDIP.pdf Accessed on 15th May 2012

[6] Chaikin, D. (2007) Network investigation of cyber attacks: the limits of digital evidence. [Online] Available at: http://www.springerlink.com/content/g002057l013811gw/ (Accessed: 18 March 2011)

[7] Champlain, J. (2003) Auditing information systems 2nd ed. New York: Wiley

[8] Ciardhuain, S. (2004) An extended model of cybercrime investigation Available (online): www.ijde.org/citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80...

Accessed on 11th April 2012

[9] Carrier, B. Spafford, H. (2003), Categories of digital investigation analysis techniques based on the computer history model. Available (Online): http://dfrws.org/2006/proceedings/16-carrier.pdf Accessed on the 12th April 2012

[10] Carrier, B. (2003) Defining digital forensic examination and analysis tools using abstraction layers Vol. 1 (4) Available (online): http://www.cerias.purdue.edu/homes/carrier/forensics Accessed on 20th September 2011

[11] Casey, E (2004) Digital evidence and computer crime forensic science, computers and the internet 2nd Edition Pg 101 Academic Press – London

[12] Casey, E. (2002) Handbook of computer crime and investigation Pg 116 Academic Press - London

[13] Freiling, F. Schwittany, B. (2007) A common process model for Incident Response and computer forensic Available (online): http://whitepapers.hackerjournals.com/wp-content/uploads/2010/06/A-Common-Process-Model-for-Incident-Response-and-Computer-Forensics.pdf Accessed on 17th April 2012

[14] Kruse II, W. Heiser, J (2002) Computer Forensics Incident Response Essentials Pg 170 Addison - Indianapolis

[15] Palmer, G. (2001) a road map to digital forensic research Available (online): http://www.dfrws.org/2001/dfrws-rm-final.pdf Accessed on 25th April 2012

[16] Panda labs Annual Report (2009) Available (online):http://www.pandasecurity.com/img/enc/Annual_Report_Pandalabs2009.pdf Accessed on 5th May 2012

[17] Perumal, S. (2009) Digital forensic model based on Malaysian investigation process Vol. 9 (8) Available (online): http://paper.ijcsns.org/07_book/200908/20080805.pdf Accessed on 7th May 2012

[18] Polli, E, Joshi, R. Niyosi, R. (2010) Network Forensic Frameworks: Survey and research challenges

Available (Online): http://www.sciencedirect.com/science?_ob=MiamiImageURL&_cid=273059&_user=132444&_pii=S1742287610000113&_check=y&_origin=&_coverDate=31-Oct-2010&view=c&wchp=dGLzVlt-zSkzS&md5=f0345fa37fdcbc76113b1d98c9a83367/1-s2.0-S1742287610000113-main.pdf Accessed on 28th April2012

[19] Pollitt, M. (2007) An Ad Hoc Review of Digital Forensic Models, Vol. 10(12) Available (Online): http://www.ieeexplore.ieee.org/ie15/4155337/4155338/04155349.pdf? Accessed on the 17th April 2012

[20] Reith, M. Carr. C. Gunsch, G. (2002). An examination of digital forensic model. Department of Electrical and Computer Engineering Air force institute of technology. Wright-Patterson. Available (Online): http://www.utica.edu/academic/institudes/ecii/ijde/articles.cfm?action Accessed on the 7th May 2012

[21] Sommer, P. (2009) Directors' and Corporate Advisors' Guide to Digital Investigations and Evidence 2nd ed. London: Information Assurance Advisory Council [Online] Available at: http://www.iaac.org.uk/_media/DigitalInvestigations2009.pdf Accessed: 3 September 2012.

[22] Stephenson, P. (2003) A comprehensive approach to digital incident investigation, Elsevier Information Security Technical Report. Vol. 8(1) [Online] Available at: http://www.emich.edu/cerns/downloads/pstephen/Comprehensive-Approach-to-Digital-Investigation.pdf Accessed: 20 April 2010

Article: Methods of Field Theory and Wave Processes in the Models of Multiple Intelligences

James William Brooks[1]**, Dmitry Zhukov**[2]**, Irina Samoylo**[3] **and Victoria Hodges**[4]

[1]Chancellor, Salem International University, Salem, West Virginia, USA

[2]Professor, Consultant, Department of Medical and Biological Physics,
I.M. Sechenov First Moscow State Medical University, Moscow, Russia

[3]Professor, Department of Medical and Biological Physics,
I.M. Sechenov First Moscow State Medical University, Moscow, Russia

[4]Consultant, Department of Medical and Biological Physics,
I.M. Sechenov First Moscow State Medical University, Moscow, Russia

Abstract - *This study represents a vector model of intellect and it also demonstrates entered by the authors the notions of vectors of intelligence and preparedness for a given type of intellectual activity and the scalar action on implementation of intellectual activity. Based on Field Theory, the mathematical substantiation of the wave nature of intelligence and intellectual activity has been received. The boundary value problem has been formulated and solved in order to assess the change of the condition of the student's preparedness to certain intellectual activity. The solution of this problem will allow prediction of the development of the process of intellectual activity as well as to study the possibility of the processes of interference among separate kinds of intelligence.*

Keywords: Multiple intelligences, field theory, wave processes.

Using the terminology of the theory of complex systems, it can be said that the intelligence of each one of us is a complex self-organizing system which is growing out of the integration of multiple heterogeneous personality traits [1]. The state of this system is described by a huge number of individual characteristics (parameters) – the number of degrees of freedom of the intellect. Based on the theory of multiple intelligence proposed by H. Gardner [2], we can make an assumption that the order parameters of the human intellect are relatively autonomous cognitive abilities by which people differ from each other and which make a significant influence on their learning and work.

A remarkable confirmation of Gardner's theory was the result of a large-scale experiment conducted by the researchers from the University of Western Ontario in Canada and the London Museum of Science [3]. That experiment was to study the activity of various areas of human brain. It showed that while solving problems of the test, the volunteers had stepped up many areas of the brain responsible for completely different functions. «We propose that human intelligence is composed of multiple independent components», the authors said following Gardner's statement.

That is to say that the main feature of intelligence is that its manifestation can be expressed through a certain number of order parameters (or variables that can be observed in practice), defining the dimension of intelligence (a priori, it is possible to state that intellect has N manifestations). Any state of the object (intelligence) in an N − dimensional space can be considered as a vector whose components along the corresponding axes define the given intellectual state and express their characteristics (competences) for different types of intelligence.

Setting the parameter action is an expression of the internal motion of a human being, *the process of achieving a specific result while using available types of intelligence.* The scalar quantity (Д) of an activity as of a process leading to the result should be connected with the change in time of an intellectual activity. To describe an activity, the use of one of the concepts of the intelligence vector ($\vec{И}$) is insufficient. A competent (effective) human action in solving certain tasks can be defined by the preparedness of intellect to implement certain kinds of activity. If intelligence is a vector quantity ($\vec{П}$) in a space of dimension N, then the preparedness must also be a vector quantity ($\vec{П}$) in the same space (with the axes relevant to each type of intelligence).

The main difference between intelligence and preparedness is the problem of their measurability. The magnitude of the vector of intelligence is fundamentally not the measured value, while the magnitude of the vector of preparedness for some components of intelligence can be measured, for example, an individual's mathematical abilities can be measured in test points.

When searching for solutions, a person uses a whole set of different kinds of intelligence, the totality of which can be considered as a certain hypersurface (denoted S) in the space of the vector of intelligence. This allows us to introduce the concept of the vector flow of intelligence (let us designate it as ($Ф_и$) as the scalar multiplication of the vector of intelligence ($\vec{И}$) by the value area of the hypersurface S and the vector of normal ($\vec{n}$) to the hypersurface S:

$$\Phi_и = \vec{И}S\vec{n} = \vec{И}\vec{S} \qquad (1)$$

The value of the (Д) should be greater, the greater and faster the time variation of the flux of intelligence ($Ф_и$) is. In

the proposed approach of the values of Д and $\Phi_и$ should be connected as follows:

$$Д = \frac{\partial \Phi_и}{\partial t} \qquad (2)$$

where t – is time, and $\frac{\partial \Phi_и}{\partial t}$ defines the rate of the change in intellectual activity, and the value

$$\Phi_и = \int \vec{И} d\vec{S}. \qquad (3)$$

If the value of the action (Д) can be determined through the operations with the vector of intelligence ($\vec{И}$), then it must also be determined through the operations with the vector of readiness ($\vec{П}$):

$$\Phi_и = \vec{П} S \vec{n} = \vec{П} \vec{S} \qquad (4)$$

In the construction of the hypersurface S in N – dimensional space of the variables which express the manifestations of intelligence, the hypersurface should be limited to the closed–loop L. According to the models of vector analysis we can introduce the concepts of the circulation of the vector of intelligence ($\vec{И}$) and the vector of preparedness ($\vec{П}$) along the closed–loop L.

$Д = \alpha_1 \int \vec{П} d\vec{l}$ where $\int \vec{П} d\vec{l}$ determines the circulation of the vector of preparedness in the closed–loop L, and α_l – is a coefficient of proportionality and dimension which transmits the measure units of circulation into the activity units.

On the other hand

$$Д = \frac{\partial \Phi_и}{\partial t} = \frac{\partial}{\partial t} \int \vec{И} d\vec{S} = \int \frac{\partial \vec{И}}{\partial t} d\vec{S} \qquad (5),$$

the value $d\vec{S}$, does not depend on time t. In this case, we can use the value representation of $d\Phi_и = \int \vec{И} d\vec{S}$.

Using these relations we can write down that:

$$\int \frac{\partial \vec{И}}{\partial t} d\vec{S} = \alpha_1 \int \vec{П} d\vec{l} \qquad (6)$$

Application to circulation of the vector of preparedness of Stokes' Theorem allows us to write down the followed:

$$\alpha_1 \int \vec{П} d\vec{l} = \alpha_1 \int rot \vec{П} d\vec{S} = \int \frac{\partial \vec{И}}{\partial t} d\vec{S}. \qquad (7),$$

Since the variable and the region of integration in the last two integrals are the same, then:

$$\frac{\partial \vec{И}}{\partial t} = \alpha_1 rot \vec{П} \qquad (8),$$

It should be noted that the rate of the change of the flux of the vector of preparedness can be reduced in proportion to the size of the stream itself ($\frac{\partial \Phi_П}{\partial t} \sim -\Phi_П$), that is connected to the fact that the greater the stream of the vector of preparedness is, the harder and harder it is to maintain its change in time (it requires more and more resources).

In accordance with the above methodology, we can write down the followed:

$$\frac{\partial \Phi_П}{\partial t} = \int \frac{\partial \vec{П}}{\partial t} d\vec{S} = -\alpha_2 \int \vec{И} d\vec{l} - \alpha_3 \Phi_П =$$
$$-\alpha_2 \int rot \vec{И} d\vec{S} - \alpha_3 \int \vec{П} d\vec{S} \qquad (9)$$

where $\int \vec{И} d\vec{l}$ – is the circulation of the vector of intelligence in the closed-loop L, which characterizes the local state of intellect and stimulates further change of its preparedness. And α_2 – is a coefficient of proportionality and dimension taking the units of measure of the circulation of the vector of intelligence in the rate of change of the units of measure of the

flux of the change of the vector of preparedness, and α_3 – is the aspect ration and dimension transforming the units of measure of the vector of preparedness into the units of measure of the change of flux of the vector of preparedness. The use of Stokes' Theorem in this case leads to the following result:

$$\frac{\partial \vec{П}}{\partial t} = -\alpha_2 rot \vec{И} - \alpha_3 \vec{П} \qquad (10)$$

Let us act on both sides of the equation (1) using the operator rot and let us express $rot \vec{И}$ from the equation (10) and by having used differentiation in t, we obtain the equation (11):

$$grad\ div \vec{П} - \nabla^2 \vec{П} = -\frac{1}{\alpha_1 \alpha_2} \frac{\partial^2 \vec{П}}{\partial t^2} - \frac{\alpha_3}{\alpha_1 \alpha_2} \frac{\partial \vec{П}}{\partial t} \qquad (11)$$

Similarly,

$$grad\ div \vec{И} - \nabla^2 \vec{И} = -\frac{1}{\alpha_1 \alpha_2} \frac{\partial^2 \vec{И}}{\partial t^2} - \frac{\alpha_3}{\alpha_1 \alpha_2} \frac{\partial \vec{И}}{\partial t} \qquad (12)$$

The vector of preparedness also characterizes the fixed state of readiness, and in the physical sense $grad\ div \vec{П} = 0$, while $grad\ div \vec{И}$ can (and/or it should) be different from zero. Thus, the basic equations proposed on the basis of Field Theory Models of implementation of competences will be as follows:

$$\nabla^2 \vec{П} = \frac{1}{\alpha_1 \alpha_2} \frac{\partial^2 \vec{П}}{\partial t^2} + \frac{\alpha_3}{\alpha_1 \alpha_2} \frac{\partial \vec{П}}{\partial t} \qquad (13)$$

$$grad\ div \vec{И} - \nabla^2 \vec{И} = -\frac{1}{\alpha_1 \alpha_2} \frac{\partial^2 \vec{И}}{\partial t^2} - \frac{\alpha_3}{\alpha_1 \alpha_2} \frac{\partial \vec{И}}{\partial t} \qquad (14)$$

Let us dwell on the consideration of the equation (13). The equations of the form of (13) are known in Mathematics and Physics as the wave equations with damping which describe the propagation of waves in media with absorption – *the damped waves*. Thus, in this study we presented *a rigorous mathematical justification of the wave nature of intellectual activity.*

In addition, the received mathematical model allows solving a number of applied problems. In particular, learning processes can be described in specific situations for given boundary and initial conditions of individual learning as the foundation for the subsequent selection of an optimal method of teaching.

Education under the given conditions. A model for the formation of a point source of the given type of intelligence.

In the simplest one–dimensional case (where the preparedness П will be a scalar quantity), and when we consider the preparedness change, for example in cases of formation of logical-mathematical intelligence, the equation (13) can be written down in a scalar form:

$$\nabla^2 \Pi(x,t) = \frac{1}{\alpha_1 \alpha_2} \frac{\partial^2 \Pi(x,t)}{\partial t^2} + \frac{\alpha_3}{\alpha_1 \alpha_2} \frac{\partial \Pi(x,t)}{\partial t} \qquad (15)$$

Let the boundary and initial conditions are:

$$\left. \frac{\partial \Pi(x,t)}{\partial t} \right|_{x=0} = 0 \qquad (16)$$

$$\left. \frac{\partial \Pi(x,t)}{\partial t} \right|_{x=L} = 0 \qquad (17)$$

$$\Pi(x,0) = f(x) = \Pi_0 \delta(x - x_0) \qquad (18)$$

$$\left. \frac{\partial \Pi(x,t)}{\partial t} \right|_{t=0} = \varphi(x) = 0 \qquad (19)$$

We can introduce the expression for the function of students' preparedness during the formation of a given type of

intelligence as the function of a point source located at time $t=0$ in the point of $x = x_0$ (formula (20)) as follows:

$$\Pi(x,t) =$$

$$\frac{2\Pi_0}{L} e^{-bt} \sum_{n=1}^{\infty} cos\left\{\pi n \frac{x_0}{L}\right\} cos\left\{\pi n \frac{x}{L}\right\} \left\{ cos\left\{t\sqrt{\frac{\pi^2 n^2 \alpha^2}{L^2} - b^2}\right\} + \right.$$

$$\left. \frac{b}{\sqrt{\frac{\pi^2 n^2}{L^2}\alpha_1\alpha_2 - \frac{\alpha_3^2}{4}}} sin\left\{t\sqrt{\frac{\pi^2 n^2}{L^2}\alpha_1\alpha_2 - \frac{\alpha_3^2}{4}}\right\}\right\} \qquad (20)$$

Figure 1 shows one of the possible graphs illustrating the results of theoretical calculations performed by the formula (20).
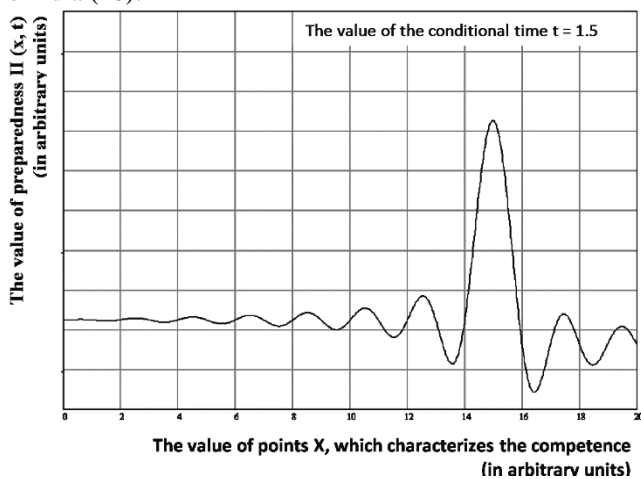
Fig. 1. The time dependence for the wave of the formation of a given type of intelligence at the point source

Figure 1 shows that the preparedness $\Pi(x,t)$ is a wave process with a variety of maxima of different sizes. It is important to note that the position of maxima of preparedness during the training process over time shifts to higher points, which is consistent with the observed in practice learning outcomes. The preparedness value spreads in waves in the domain of its definition from the initial skills (the low values of the parameter X) to the skills of higher level (the greater values of the parameter X). Such a wave can be seen as a process of structuring space competences in which at first, simple competences (the area of low values of their parameter X) are formed, and then more and more complex competences are formed (the area of high values of their parameter X). A point source model is a good approximation for study of the evolution of a given type of intelligence in the process of its formation as it allows to obtain solutions for different conditions. In particular it is possible to consider a periodically operating source of training if we set its function $S(x,t)$. In this case, the decision will be the convolution of the functions of the point source $\Pi(x,t)$, given by the formula (20) and $S(x,t)$.

Conclusions:
- Human intelligence can be viewed as a vector in a field and in the space which dimension is determined by the number of types of intelligence.
- Human intellectual activity can be seen as a wave process.
- The wave nature of intellectual activity on formation and implementation of competences involves the possibility of the appearance of interference processes of blending skills at which their essential mutual strengthening takes place (or easing can also take place for those skills which are negative by nature (a positive education)). The interference is characteristic of the so-called coherent waves. This will help understand *what features should be present in different processes of creating competencies so that they start mutually reinforcing each other.*
- The wave models of building skills and competences can develop new and efficient algorithms for control and management of intellectual development in particular areas.

Literature:
1. John Raven, *Competence in Modern Society: Its Identification, Development and Release (1984; London; H. K. Lewis& Co. Ltd)*
2. Howard Gardner, *Frames of Mind: The Theory of Multiple Intelligences* (1983; repr., New York: Basic Books, 2004).
3. Adam Hampshire, Roger R. Highfield, Beth L. Parkin, Adrian M. Owen, *Fractionating Human Intelligence*, Neuron (Volume 76, Issue 6, 20 December 2012, Pages 1225-1237).

A Behavioral Programming Approach to the Design-Development of Humanoid Robots

Alan Rosen
Machine Consciousness Inc.
1917 Clark Lane
Redondo Beach, CA 90278 USA

David B. Rosen
Machine Consciousness Inc.
1917 Clark Lane
Redondo Beach, CA 90278 USA

Abstract *An introduction to a radical new approach to solving the many challenges of Artificial Intelligence (AI) in humanoid robotics. Through the use of a Relational Robotic Controller (RRC) and Relational Correlation Sequencer (RCS) modules, including a self-location and identification coordinate frame, behavioral programming techniques may be employed to achieve human-like levels of AI for the identification, recognition, visualization, and comprehension of the input signals. Behavioral programming techniques may also be applied to auditory signals and to the control of a verbal-phoneme sound generator, to achieve human-like levels of AI for declarative-verbal capabilities of a humanoid robot. The approach described herein represents a paradigm shift in today's analytical-programming methodology.*

Keywords: human-like artificial intelligence; humanoid robots; thinking machines; behavioral-programming; behavioral speech processing.

1   Introduction

By the 1950's the theoretical foundation for Artificial Intelligence (AI) had been established [1] [2] and Alan Turing, arguably, the founder of AI, posed the question: "When is a machine thinking?" More than 60 years later, this question has not been answered satisfactorily. In the discipline of AI, the question may be reworded to: "When does a machine display human-like levels of AI?"
Alan Turing's approach to an answer was in terms of the behavior of the machine. He devised a "Turing test" based on the conversational behavior of the machine; and deemed any machine that passed the test to be a thinking machine [2] [3].
In this paper we describe a building path for a machine that can reach human-like levels of AI, defined in terms of the behavior of the machine. It describes a behavioral programming approach for the design of a smart RRC based humanoid robot, called an RRC-Humanoid Robot. The RRC-Humanoid Robot is a human-like robotic system, controlled by a proprietary Relational Robotic Controller (RRC) [4] [5] [6] [7].

1.1 But First, a Note About Human-like Levels of AI

1.1.1 Human intelligence is experiential intelligence. Humans learn from, and remember their experiences throughout their lifetime. A behaviorally programmed RRC-humanoid robot emulates the experiential intelligence of a human.

1.1.2 Humans have a self-location and identification coordinate frame that is trained from infancy to give the human brain a proprioceptive self-knowledge capability. Even a baby, with a self-knowledge capability, instinctively knows the location of every surface point on its body; the location of its flailing limbs; and by extension, the location of every coordinate frame point in the near space defined by its flailing limbs. The fundamental design characteristic of the RRC-Humanoid Robot is a centralized hub of intelligence, a proprietary module that is the centralized "self location and identification" coordinate frame of the system. This module gives the RRC-Humanoid Robot a robotic form of proprioceptive knowledge, similar to human proprioceptive intelligence. In the RRC-Robot, the self-knowledge capability is the basis for all knowledge.

1.1.3 In order to achieve contextual, or 'self-knowledge' of visual data, auditory data, olfactory data, gustatory data, and vestibular data, all the data obtained from those other human-like sensors must be related and correlated with the self-knowledge, self location and identification coordinate frame.

1.1.4 The human brain relates, correlates, prioritizes and remembers sensory input data. Similarly, to achieve human-like intelligence, relating, correlating, prioritizing and remembering input patterns must be the essential analysis tool. In contrast, most robotic computers are designed to calculate, compute and solve problems related to sizes, distances, shapes, and colors of objects recorded in the Field

Of View (FOV) of the visual system or sensed by the other sensors.

1.1. 5 Human intelligence is gained only from the 6 external sensors: tactile, visual, auditory, olfactory, gustatory, and vestibular sensors. These sensors provide for the consciousness associated with human 'feeling,' 'seeing,' 'hearing,' 'smelling,' 'tasting,' and 'balancing.' The recording monitors of the RRC-Humanoid Robot emulate the external sensors of humans.

1.1.6 In a smart RRC-Humanoid Robot, the mechanical robotic body and associated sensors simulate the human body and the human sensors. Also the control system operates in a manner similar to the human brain; that is, by relating and correlating the input data rather than computing and calculating distance, size, and shape.

2.0 Behavioral Programming and the Development of Human-like AI

Behavioral programming is achieved by training the humanoid robot to control its body, limbs, and verbal-phoneme sound generator on the basis of input data from the six external sensors. It is an experiential supervised programming technique analogous to the autonomous learning of a human. Behavioral programming techniques are employed for all the sensory input signals of the humanoid robot.

For example, the tactile input signals are used to define the central hub of intelligence, the self-nodal map/coordinate frame, of the humanoid robot. The behavioral programming technique employed for the self location and identification 'self-knowledge' coordinate frame is an itch-scratch methodology, wherein the robot is fully trained and remembers how to a) reach and touch (scratch) all points located on the surface of the robotic body, and all points in the near space surrounding the robotic body, b) to identify and locate all such points, and c) to identify and locate all the "end joint" body parts (ends of fingers, elbow, knee etc.) used to scratch all the itch points. When the level of training reaches the threshold of 'self-knowledge,' the Self Nodal Map Module and associated TSMs will facilitate the robotic identification and recognition of all body parts, and the navigation of all movable parts of the robot towards any and every itch point located on the surface of the robotic body and all points in the near space surrounding the robotic body.

The totality of the programmed "self location and identification" data, stored in a TSM-memory module, is the basis for the "self-knowledge" level of intelligence. Analogous to the proprioceptive knowledge of a human, a RRC-Robot with a fully programmed 'self-knowledge' capability 'knows,' behaviorally, the location of every surface point of the robotic body, the location of flailing limbs, and by extension, the location of every coordinate frame point in the near space defined by flailing limbs.

In the visual and auditory RRC-Humanoid systems, experiential intelligence is obtained by performing behavioral programming on the processed raw data coming from the video visual recording monitor and the auditory recording monitor. The raw data is processed in an Interface Circuit, inputted to the RRC and then behaviorally programmed to reach human-like levels of AI. Behavioral programming reaches the level of experiential human-like intelligence, when the RRC-Humanoid Robot demonstrates behaviorally that it has identified, recognized, visualized and comprehended in the same manner as does a human, the signals coming from the visual sensors, or the auditory sensors. The following sections will describe the processing of the raw data in the Interface Circuit, and the behavioral programming of the processed data within the RRC-Humanoid Robot. On completion of behavioral programming, the RRC-Humanoid Robot demonstrates behaviorally human-like levels of AI for the identification, recognition, visualization or comprehension of the processed raw data. Length

3.0 Processing the Visual Raw Data in the Interface Circuit and Behaviorally Programming the Processed Data.

3.0.1 The Interface Circuit stage: The visual raw data consists of the output of the two CCD arrays of the 2-video visual cameras of the visual recording monitor. This data is inputted to the Interface Circuit. The Interface Circuit generates a real time 3D-photometric image that is a high fidelity representation of the real objects that gave rise to that image. The Interface Circuit includes a processing stage for the calibration and projection of the 3D-photometric image onto the self-location and identification Nodal Map Module, the centralized 'self-knowledge' coordinate frame of the system.

3.0.2 Behavioral Programming of the Calibrated 3D-photometric Image: Once the 3D-photometric image is calibrated and projected onto the self-knowledge Nodal Map Module of the RRC, The RRC-Humanoid Robot is behaviorally programmed to control its body and limbs, and the verbal phoneme sound generator in relation to the 3D-photometric image. The 3D-photometric image is identified, recognized, visualized, and comprehended by generating different words by the verbal phoneme sound generator or undertaking distinguishing body or limb actions based on the 3D-photometric image. The following aspects of the image are processed: a) The rainbow of colors between the 4000 Angstrom- purple and the 8000 Angstrom-red. b) All objects in the field of view of the robotic system. And c) All shapes, forms and colors of the 3D-objects in the field of view of the robotic system.

The identification, recognition, visualization and comprehension of all objects, shapes, colors, and forms is

achieved behaviorally. The RRC-Robot behaviorally moves its body and limbs, and controls its verbal phoneme sound generator so as to distinguish the various identifications, recognitions, visualizations and comprehensions of all objects in the FOV of the visual system.

Note: Behavioral-programming of the Auditory RRC-Humanoid Robot generates an operational definition of the 'identification', 'recognition' and 'comprehension' levels of AI. Any human need merely verbally ask the RRC-Robot to identify, recognize, comprehend, or visualize any color or 3D-object in the FOV, in order to obtain a response that is indistinguishable from the response of another human.

For example: A person can hold an 'apple' in front of the Robot and ask the Robot to "Identify the object I am holding in front of you." A fully programmed Auditory RRC-Humanoid Robot will respond: "That is an apple." The robot could also demonstrate comprehension of the image by responding verbally with encyclopedic data relating to the object, etc.

4.0 Processing the Auditory Raw Data in the Interface Circuit and Behaviorally Programming the Processed Data.

4.0.1 The Interface Circuit Stage: The auditory raw data consists of the output of the sound receiving microphones that are sensitive to the auditory frequency range of zero to 20,000 cps (simulating the human ear). When a talking sound is applied to the ear-like receiving microphones, they generate a sequence of frequencies and amplitudes as a function of time. These frequencies and amplitudes may be illustrated in an amplitude-frequency-time (a-f-t)-diagram. This (a-f-t) data is inputted to the Interface Circuit.

The Interface Circuit stage performs the following functions: a) It processes the a-f-t data into collective modalities that have been selected to be characteristic of 120 phoneme-sound combinations present in the English language. b) With the aid of a spectrum analyzer, the collective modalities are tuned to the selected phoneme sound combinations in the English language, to musical sounds or to environmental noise. c) It performs behavioral speech processing on sequential sets of phoneme sounds and identifies and recognizes these sequential sets as " words" or "sentences" formed by the person speaking to the microphones of the auditory recording monitor. (Note: The behavioral speech processing performed in the Interface Circuit stage is described in the next section). d) The Interface Circuit may be programmed to recognize, identify and correct, incorrect grammatical structures. e) In addition the (a-f-t) data must be formatted within the Interface Circuit so that it is compatible with the RRC-Phoneme Sound Generator and the input to the RRC-Multi-dimensional Nodal Map Module [4] [5] [6] [7].

The final output of the Interface Circuit stage: The 'words and sentences,' music or environmental noise are inputted to their respective Multi-dimensional Nodal Map Modules of the RRC-Humanoid Robot, for behavioral programming.

4.0.2 Behavioral Programming of the Words and Sentences Inputted to the RRC-Multi-dimensional Nodal Map Module: Once the words and sentences are projected to the Multi-dimensional RRC-Nodal Map Module, the RRC-Humanoid Robot is behaviorally programmed to control body and limbs, and the verbal phoneme sound generator in relation to the words and sentences applied to the Multi-dimensional Nodal Map Module. Words and sentences are identified, recognized, and comprehended by behaviorally programming the RRC-Robot to generate different words by the verbal phoneme sound generator or undertaking distinguishing body or limb actions based on the words or sentences applied to the Multi-dimensional RRC-Nodal Map Module. The following behavioral programming procedures are undertaken: 1. Training to repeat, read, and write phoneme-sound based words and sentences by relating and correlating the acoustic content with the visual image represented by the signals. 2. Training the robot to comprehend the meaning of a 'heard' word. 3. Training to verbally describe experiential sensory data obtained from the visual, tactile, olfactory or gustatory sensors. 4. Training the robot to get into a 'conversational Mode,' 5. Training the robot to respond to commands, etc.

5.0 Speech Processing by Behavioral Programming the Auditory RRC-Humanoid Robot: Recognizing the Acoustic Sequential Set of Phoneme-signals as Phonetic Words and Sentences.

5.1 The Problem:

The problem of converting the perceived acoustic spectrographic (a-f-t) properties of language into an identifiable phonetic structure is an ill posed problem, similar to the inverse optics problem [8]. There is not a simple one to one mapping between the acoustic properties of speech and the phonetic structure of an utterance. Co-articulation (the segmentation problem) is generally identified as the major source of the problem. Co-articulation gives rise to difficulty in dividing the acoustic signal into discrete "chunks" that correspond to individual phonetic segments and a lack of invariance in the acoustic signal associated with any given phonetic segment.

Note: The usual methods for solving the problem includes lexical segmentation processing (co-articulation), word recognition processing, interactive-activation processing, context effect processing, syntactic effects on lexical access processing, lexical information and sentence processing, syntactic processing and intonation-structure processing.

5.2 The Behavioral Speech Processing Methodology for Solving the Inverse Auditory Problem.

Because of the complexity in the mapping between the acoustic signal and phonetic structure, an experiential, behavioral programming methodology was developed for 'unpacking' the highly encoded, context dependent speech signals. 'Unpacking' is performed in the Interface Circuit by programming the RRC to repeat and 'remember' (in the TSM-memory modules) the 'heard' words and sentences of multiple speakers.

Further 'unpacking' is performed by behavioral programming techniques that includes the following: First, by relating, correlating, associating and calibrating the heard verbal speech with the corresponding visual and tactile data obtained in the visual and tactile coordinate frames in which the robot is operating. Next, by training the RRC-Robot to be sensitive to such factors as acoustic phonetic context, speaker's 'body language,' speaking rates, loudness and 'emotion laden' intonations. The Auditory RRC-Humanoid Robot takes into account the acoustic consequences of such variations when mapping the acoustic signal into the phonetic structure. The problems of speaker's 'body language,' 'emotion laden' intonations, acoustic phonetic context, speaking rates, and loudness is solved in the Auditory RRC by coordinating the search engines of the visual and tactile systems with the search engine of the Auditory RRC-Humanoid Robot. speaking rates, loudness and 'emotion laden' intonations. The Auditory RRC-Humanoid Robot takes into account the acoustic consequences of such variations when mapping the acoustic signal into the phonetic structure. The problems of speaker's 'body language,' 'emotion laden' intonations, acoustic phonetic context, speaking rates, and loudness is solved in the Auditory RRC by coordinating the search engines of the visual and tactile systems with the search engine of the Auditory RRC-Humanoid Robot.

6.0 Innovative Features of RRC-Humanoid Robots: How the Invention Differs From, and is an Improvement Over, What Currently Exists.

Present day humanoid robots have never before been programmed with human-like AI that includes human-like identification, recognition, visualization and comprehension of a) the words and sentences 'heard' by an Auditory RRC-Humanoid Robot, b) the full array of sizes, distances, shapes, and colors of objects recorded in the FOV of the Visual RRC-Humanoid Robot, and c) a capability to respond verbally and intelligently to the queries or statements spoken by humans, or the visual signals observed by the robot. The RRC-Humanoid Robot does so because of innovative features that have been incorporated into the system. The following description of innovative features is divided into 3 parts; Part 1 describes innovative features that are common to the Visual

and Auditory RRC-Humanoid systems. Part 2 describes features that are unique to the Visual-RRC-Humanoid Robot. Part 3 describes features that are unique to the Auditory-RRC-Humanoid Robot.

6.1 Part 1-Innovative Features Common to the Visual and Auditory RRC-robots

6.1.1 Incorporation of the RRC ((Rosen A, & D.B. Rosen, 2003, 2006a,b,c): The RRC is an operating system that has been designed (by reverse engineering the functional characteristics of the human brain) to relate, correlate, prioritize and remember visual and auditory input data. Relating, correlating, prioritizing, and remembering visual and auditory input patterns is the essential analysis tool required to achieve human-like intelligence levels and to reduce the amount of programming required to a bounded number of programming steps.

    • Compared to other systems: Most other computer systems are artificially intelligent by calculating, computing and solving problems related to speech processing, or sizes, distances, shapes, and colors of objects recorded in the FOV of the visual system. They are not programmed to achieve human-like levels of AI.

6.1.2 Incorporation of the Relational Correlation Sequencer (RCS): An RRC consists of sets of RCS-modules and associated memory units called Task Selector Modules (TSMs) that operate by relating and correlating the input signals and prioritizing and remembering important correlations. The RCS is a proprietary module described by Rosen and Rosen (Rosen A, & D.B. Rosen, 2003, 2006a,b,c).

    • Compared to other systems: Most other robotic systems are not made up of modules specifically designed to relate and correlate the input signals and prioritize and remember important correlations.

6.1.3 Incorporation of a Behavioral Programming Methodology for Training the Humanoid Robot.

Behavioral programming is achieved by training the humanoid robot to control its body, limbs, and verbal-phoneme sound generator on the basis of input data from the 6 external sensors. It is an experiential supervised programming technique analogous to the autonomous learning of a human. The disadvantage of behavioral programming is that the RRC-Robot must be a fully built, mechanically operational system before behavioral programming techniques can be initiated. The advantages of behavioral programming is that it yields human-like levels of artificial intelligence never before programmed into a computer, as follows:

   a) Tactile behavioral programming yields a self-location and identification level of intelligence analogous to the proprioceptive intelligence and self-knowledge of a human.

b) Visual behavioral programming yields a behavioral form of intelligence for the identification, recognition, visualization, and comprehension of visual images analogous to humans. As a matter of fact, the operational definition of human-like identification, recognition, visualization, and comprehension is derived from the behavioral programming methodology.

c) Auditory behavioral programming yields a behavioral form of intelligence for the identification, recognition, and comprehension the "heard" sound input signals, and the capability to respond verbally (a behavioral control function), in the same manner as humans.

d) Auditory-visual-tactile behavioral programming yields a behavioral form of intelligence for the formation of human-like abstractions and human-like conceptualization of the input data as related to the data stored in the memory system. For example, behaviorally, a 'chair' is defined as an object that the robot can 'sit on.' Then the robot is trained to identify and recognize these objects by behaviorally 'sitting on them.' Thus a 'bean bag,' a bar stool, a recliner, or a folding chair would all be identified conceptually as chairs. And the robot has thereby achieved a level of abstraction for the concept of a chair. In a similar manner the robot may be trained to recognize and identify common nouns representing such everyday objects as 'doors' (a closeable entrance or egress opening of an enclosed space), table, tree, and verbs that are descriptive of 'freedom,' 'slavery,' 'democracy,' and 'totalitarianism.'

6.1.4 Incorporation of a Central Hub of Intelligence:
The RRC-Humanoid Robot is programmed to perform all tasks relative to a self-location and identification task, performed by a nodal map, known as the Self Nodal Map/coordinate frame, and associated with one of the RCSs that make up a RRC. It is important to stress the word "all", since no task may be performed by the system that is not related to the centralized Self-Nodal Map/coordinate frame. The centralized Self-Nodal Map coordinate frame is the central hub of intelligence for the system.

   • Compared to other systems: Most other intelligent computer systems do not relate all the programmed tasks to a single centralized coordinate frame or task. Therefore it is much more difficult to access the diverse "knowledge-data" stored in the computer system.

6.1.5 Design of a 'Self-knowledge' Capability:
A trained Self-Nodal Map-coordinate frame gives the robot a level of intelligence that may be called robotic proprioception knowledge or 'self location and identification' knowledge. The totality of the programmed 'self location and identification' data, stored in a TSM- memory module, is the basis for the 'self-knowledge' level of intelligence. A RRC Robot with a fully programmed "self-knowledge" capability 'knows,' behaviorally, the location of every surface point of the robotic body, the location of flailing limbs, and by extension, the location of every coordinate frame point in the near space defined by flailing limbs.

   • Compared to other systems: Most other computer systems don't have proprioceptive knowledge or a "self-knowledge" capability-coordinate frame to which all other data may be related. They do not internalize the data into a self-knowledge coordinate frame; that is, they do not relate all the programmed tasks to a single centralized coordinate frame. Examples of such machines, which do not have a self-knowledge capability, are the famous Turing machine and the chess playing computers that always win when playing against a human competitor.

6.1.6 Design of an "Awareness"-Monitoring Capability:
In order to be capable of achieving a level of programmed intelligence that can be termed human-like 'awareness' of the input data, the robotic system must constantly monitor the sensory data throughout the operational lifetime of the robot, and relate the monitored data to the 'self-knowledge' coordinate frame. When tactile sensors that form a protective covering of the robotic body, constantly monitor the environment around the robotic body for any possible tactile activation, then robotic 'self-knowledge' becomes another level of intelligence called 'robotic self-awareness' of the tactile environment around the robot. Robotic self-awareness coupled with 'self-knowledge' of the tactile sensory data may lead to a behavioral robotic reaction to the data that is analogous to the human-like modality, the sensation of 'feeling touch-pain,' associated with the pressure exerted on tactile mechano-receptors (pressure transducers).

   • Compared to other systems: Most other computer systems designed to perform monitoring or surveillance do not have a human-like 'awareness' capability unless the monitored data is constantly related to a 'self-knowledge' coordinate frame.

6.1.7 Internalization of the Data:
In the RRC-Humanoid Robot, the sensory data obtained by any recording monitor must be 'internalized' with respect to the 'self-knowledge' memory module. Internalization means that the data from each of the sensors must be related and correlated with the self-knowledge memory module in a manner such that the robot develops 'self-knowledge' of the visual data, the auditory data, the olfactory data, and the gustatory data. The 'self-knowledge' level of intelligence may therefore be gained for the auditory, olfactory, and gustatory sensors, in addition to the visual sensors.

   • Compared to other systems: Most other computer systems do not have a 'self-knowledge' level of intelligence capability. And they certainly cannot extend that self-knowledge capability to other sensors. That is, self-knowledge cannot be extended to the visual, auditory, olfactory, and gustatory sensors.

6.1.8 Quantifying the Amount of Programming Required to Achieve Human-like Levels of AI: With the internalization process in place, achieving human-like intelligence of the sensory data is dependent on the amount of behavioral training or programming performed on a RRC-Humanoid Robot. It is a software development involving relations and correlations between signals wherein 'robotic self-knowledge,' 'robotic awareness,' 'robotic comprehension,' 'robotic visualization,' and 'sensation' generation within the RRC, all refer to the level of training programming of the various modules of the RRC. The RRC-Robot may be programmed behaviorally to the level of intelligence of a human that learns all the relational and correlational data taught in the public educational system grades K-I through K-12. The number of relations, correlations, and the priority levels stored in the TSM-memory modules of such an RRC-Robot increases proportionally to the data learned in grades K-1 through K-12. And the total 'knowledge' gained by the system may be quantified by the number of relations and correlations programmed and stored into the system as the robot is trained from grade K-1 to grade K-12.

• Compared to other systems: The designers of most other intelligent-computer systems have never quantified the level of intelligence programmed into their system by the number of relations and correlations between the various sensory data inputs.

6.2 Part 2: Innovative Features of the Visual RRC-Humanoid Robot.

6.2.1 ***A Paradigm Shift in Methodology: Formation of a 3D-photometric Image.*** The analytical-programming employed by the Visual RRC-Humanoid Robot is a fundamental paradigm shift in the methodology generally employed by other computer vision systems. The shift in emphasis is from methodologies involving calculations of image size, distance, shape, form, and color, to an analytical methodology involving the formation of a 3D-photometric image within the controller, and relating, correlating, prioritizing and remembering various aspects of a 3D-photometric image.

In the Visual RRC-Humanoid Robot great emphasis is placed on the formation within the controller of a 3D-photometric image that is a high fidelity representation of objects/colors present in the FOV of the visual system. It is that image that is calibrated, applied to the 3D-coordinate frame defined by the 'self-knowledge' Nodal Map Module, and related and correlated with other input data. In the Visual RRC-Humanoid Robot, a human-like level of visualization intelligence is obtained by behavioral programming techniques applied to relating, correlating, prioritizing, remembering and acting on various aspects of the 3D-photometric image. The super-position of the visual 3D-photometric image onto the self-knowledge, itch-scratch, coordinate frame, facilitates the internalization of the 3D-photometric image and the development of the visualization level of AI.

• Compared to other systems: The formation of a high fidelity 3D-photometric image in the Interface Circuit of the Visual RRC-Robot is a unique, innovative and advantageous way to analyze the image, internalize the image, and reach the 'visualization' level of artificial intelligence.

6.2.2 Design of an Interface Circuit Compatible with Human-like Levels of AI: The visual Interface Circuit not only generates a 3-D photometric image, it also magnifies and displaces the image so that it is adjusted with, and completely calibrated with, the Euclidean itch-scratch, self location and identification coordinate frame, the so called self-knowledge coordinate frame of the system.

• Compared to other systems: The interface Circuit is highly dependent on the Nodal Map Module input system of the RRC, a proprietary module of the Visual RRC-Humanoid Robot.

6.3 Part 3: Innovative Features of the Auditory RRC-Humanoid Robot

The Auditory RRC-Humanoid Robot is a Visual RRC-Humanoid robot equipped with an auditory microphone monitor and verbal phoneme sound generator and programmed to reach human-like levels of declarative-verbal AI. The following innovative features are incorporated into the system.

6.3.1 A Paradigm Shift in the Analytical-Programming Methodology Employed in Speech Processing Systems: The behavioral programming methodology employed by the Auditory RRC-Humanoid Robot is a fundamental paradigm shift in the methodology generally employed by other computer speech processing systems. The shift in emphasis is from analytical programming methodologies involving calculations of acoustic signal pattern that are mapped onto phonetic structures, to an analytical methodology involving experiential or behavioral programming by relating, correlating, prioritizing, repeating and remembering various aspects of the of the acoustic signals.

6.3.2 Design of an Interface Circuit Compatible with Human-like Levels of AI: The innovativeness of the Auditory RRC-Humanoid Robot also lies in the design of an interface that facilitates a) the internalization and the human-like 'self-knowledge' level of intelligence of the auditory data; b) The design of a multi-dimensional p-phoneme vector space, input to the multi-dimensional Auditory Nodal Map Module; and c) The incorporation of a RCS that forms a babbling Sequence Stepper Module to facilitate repetitive programming.

References:
[1] Turing AM (1950), Computing Machinery and Intelligence, Mind 59: 433-460

[2] Turing AM (1953), Solvable and Unsolvable Problems, Science News 31: 7-23

[3] Rosen A. & D.B. Rosen (2007), The Turing Machine Revisited: ASSC-e archive (www.mcon.org)

[4] Rosen A. & D.B. Rosen (2003), The Design of a Volitional, Obstacle avoiding Robot, Machine Consciousness Journal 1: 41-56 (available for viewing at www.mcon.org)

[5] Rosen A. & D.B. Rosen (2006a), The Design of a Sensation Generating Mechanism, ASSC e-archive (available for viewing at www. mcon.org)

[6] Rosen A. & D.B.Rosen (2006b), A Neural Network Model of the Connectivity of the Biological Somatic Sensors, Published in IEEE –Xplore Proceedings of the IEEE-WCCI July 26-21 2006, Canada.

[7] Rosen A. & D.B. Rosen (2006c), An Electromechanical Neural Network Robotic Model of the Human Body and Brain, LNCS Part 1: 105-116; Proceedings of ICONIP-2006; Berlin: Springer-Verlag

[8] Marr D. (1962), 'Vision' Freeman Press

An artificial chemical reaction optimization algorithm for multiple-choice knapsack problem

Tung Khac Truong[1,2]**, Kenli Li**[1]**, Yuming Xu**[1]**, Aijia Ouyang**[1]**, and Xiaoyong Tang**[1]

[1]College of Information Science and Engineering, Hunan University,
National Supercomputing Center in Changsha, 410082, China.
[2]Faculty of Information Technology, Industrial University of Hochiminh city, Hochiminh, Vietnam.

Abstract— *Multiple-choice knapsack problem (MCKP) is a well-known NP-hard problem and it has a lot of applications in the real-world and theory. In this study, the Artificial chemical reaction optimization algorithm (ACROA) that uses integer string code is developed to solve MCKP. Four specific reaction operators are designed to implicate local and global search. A new penalty function that aims to force the algorithm search in both infeasible and feasible search space is suggested. The experiment on MCKP test set demonstrates that ACROA is superior to GA.*

Keywords: Artificial chemical reaction optimization; combinatorial; multiple-choice; knapsack problem.

1. Introduction

Given m classes $N_i = \{1, \ldots n_i\}$, $i = \{1, \ldots, m\}$ of items to pack in some knapsack of capacity W. Each item $j \in N_i$ has a cost c_{ij} and a size w_{ij}, and the problem is to choose one item from each class such that the total cost is minimized without having the total size to exceed W. The multiple-choice knapsack problem (MCKP) may thus be formulated as:

$$minimize \quad \sum_{i=1}^{m} \sum_{j=1}^{n_i} c_{ij} x_{ij} \qquad (1)$$

$$subject \ to \quad \sum_{i=1}^{m} \sum_{j=1}^{n_i} w_{ij} x_{ij} \leq W, \qquad (2)$$

$$\sum_{j=1}^{n_i} x_{ij} = 1, \quad \forall i \in \{1, 2, \ldots, m\}, \qquad (3)$$

$$x_{ij} \in \{0, 1\}, \forall i \in \{1, \ldots, m\}, j \in N_i. \qquad (4)$$

All coefficients c_{ij}, w_{ij}, and W are positive numbers, and the classes $N_1, \ldots, N_m$ are mutually disjoint.

MCKP is known as an NP-hard (Non-deterministic Polynomial-time hard) problem [1]. The problem has a large range of applications: Capital Budgeting [2], Menu Planning [3], transportation programming [4], nonlinear knapsack problems [2], sales resource allocation [3], design of information systems [5], etc. The MCKP also appear by Lagrange relaxation of several integer programming problems [6].

Since MCKP is an NP-hard problem. The exactly algorithms have complexity time in exponential functions. The heuristic algorithm has an advantage in finding approximate optimal in polynomial time. One of the well-known heuristic algorithm is GA [7]. Although GA is pioneer in solving MCKP, yet it has a drawback that it get stack in local optima.

Recently, Artificial chemical reaction optimization algorithm has been proposed in [8]. The ACROA is mimic from chemical reaction process. The ACROA is suggested with two encoding types: real code and binary. It is successful in multiple-sequence alignment, data mining, global numerical optimization, and others [8], [9].

In this study, the ACROA that uses integer string code is developed to solve MCKP. In the proposed algorithm, four specific reaction operators are designed to implicate local and global search. A new penalty function that aim to force the algorithm search in both infeasible and feasible search space is suggested. The experiment on MCKP test set demonstrates that ACROA superior to GA.

The rest of the paper is organized in sections: Section 3 briefly gives the original framework of ACROA, and GA. Section 4 explains the modification of the original ACROA to adapt it to the MCKP problem. We survey the behavior of ACROA and compare the simulated results of the ACROA with GA in Section 5. We conclude this paper and suggest potential future work in Section 6.

2. Genetic algorithm

In [7], a GA is proposed to solve MCKP. A chromosome is presented as m genes corresponding to m classes of items. The ith gene takes an integer number from a mutually exclusive set n_i, where $N_i = \{1, 2, ..., n_i\}$, for $i = \{1, 2, ..., m\}$. Thus the position of a gene is used to represent class that object belong to, and the value of the gene is used to represent the item selected from the class. For crossover and mutation, uniform crossover and random perturbation are used for genetic operations. For selection, roulette wheel selection is used, it also used elitist method. If the best chromosome of preceding population is not selected to new population, the elitist will replace a randomly chromosome in new population by the best one. For evaluation, A penalty method is proposed to a

fitness function to help the algorithm to search optimal solution from both the feasible and infeasible sides of the solution space. Let x_k be the kth chromosome in the current population and x_{ij}^k be the corresponding decision variables, the fitness function is given as follows.

$$f_k(x_k) = \sum_{i=1}^{m} \sum_{j=1}^{n_i} c_{ij} x_{ij}^k$$

The penalty coefficient p_k is calculated as:

$$p_k = \begin{cases} 0 & \text{if } (2) \text{ } is\ satisfied \\ \psi_0(\sum_{i=1}^{m} \sum_{i=1}^{n_i} w_{ij} x_{ij}^k - W) & \text{if } otherwise. \end{cases}$$

where ψ_0 is the large positive penalty value. And, the evaluation function is presented as follows:

$$eval(x_k) = \frac{1}{f_k(x_k) + p_k}, \quad k = \{1, 2, \ldots, popsize\}$$

3. Artificial chemical reaction optimization algorithm

The ACROA is a heuristic method proposed by Alatas in [8]. It inspired from the chemical reaction process. In the chemical reaction process, the system tends toward the highest entropy and the lowest enthalpy. The chemical reactions possess efficient objects, states, process, and events that can be designed as a computational method. Enthalpy or potential energy for minimization problem and entropy for maximization problem can be utilized as objective functions for the interested problem.

At the beginning, a $reactantNum$ number of reactant (solution) is generated. In the iteration phase, when the terminated criteria no met, then according to the condition (randomly variable), one chemical reaction operator is selected to execute. The reactants update work same as reversible reaction that help the objective function to grow forward to optima.

In ACROA, there are two chemical reaction types namely bimolecular reactions and monomolecular reactions. Bimolecular reactions are synthesis reaction, redox2 reaction, and displacement reaction; this type of reaction requires two reactants participating. Bimolecular reactions are redox1 reaction and decomposition reaction; one reactant is required in this reaction type. The ACROA flow chart of which is depicted in Fig. 1 and an outline of the algorithm is given in Algorithm 1. More details about ACROA can be found in [8], [9].

4. Design ACROA for MCKP

4.1 Solution Representation

An integer string is used to represent a reactant(solution). The y_i receives an integer in N_i, it represent $y_i \in N_i$ is chosen. The string length is m corresponding to a solution

---

Algorithm 1 ACROA algorithm

Input: Problem-specific information (the objective function f, constraints, and the dimensions of the problem)
Assign parameter values to $reactantNum$.
Setting the initial reactants and evaluation.
while stop criterion not met **do**
    Do chemical reactions.
    Reactants update.
end while
Output: The best solution.

---

y_1	y_2	y_3	$\cdots$		$y_{m\text{-}1}$	y_m

Fig. 2: Solution presentation

in MCKP. The solution presentation is depict in Fig. 2. By defining an indicator variable, y_i is as follows:

$$y_i = j \quad if \quad x_{ij} = 1, \quad j \in N_i, \quad i = 1, 2, \ldots, m$$

4.2 Objective and penalty functions

Let x be chromosome in the current population and x_{ij} be the corresponding decision variables. In the reaction, the *enthalpy* is not negative and the *enthalpy* is decreasing in the reaction process. To adopt to *enthalpy*, it is set as follows:

$$enthalpy(x) = \sum_{i=1}^{m} \sum_{i=1}^{n_i} c_{ij} x_{ij} + g(x) \qquad (5)$$

where $g(x)$ is penalty function as following:

$$g(x) = \begin{cases} 0 & \text{if } (2) \text{ } is\ hold \\ \Omega_0 + (\sum_{i=1}^{m} \sum_{i=1}^{n_i} w_{ij} x_{ij} - W) & \text{if } otherwise. \end{cases}$$

where Ω_0 is a given positive constant. The idea here is that, for violate solution will have a larger enthalpy. It forces the algorithm search both sides of search space that is feasible and infeasible domains.

4.3 Reaction operators

In this paper, we proposed integer string code for MCKP. Five specific problem reaction operators are explained as following.

4.3.1 Synthesis

In this operator one reactant will be created from two original reactants. It responds to the diversifications in the algorithm. The synthesis operator in [10] is redesigned for this problem. The pseudocode of the synthesis operator is described in Algorithm 2.
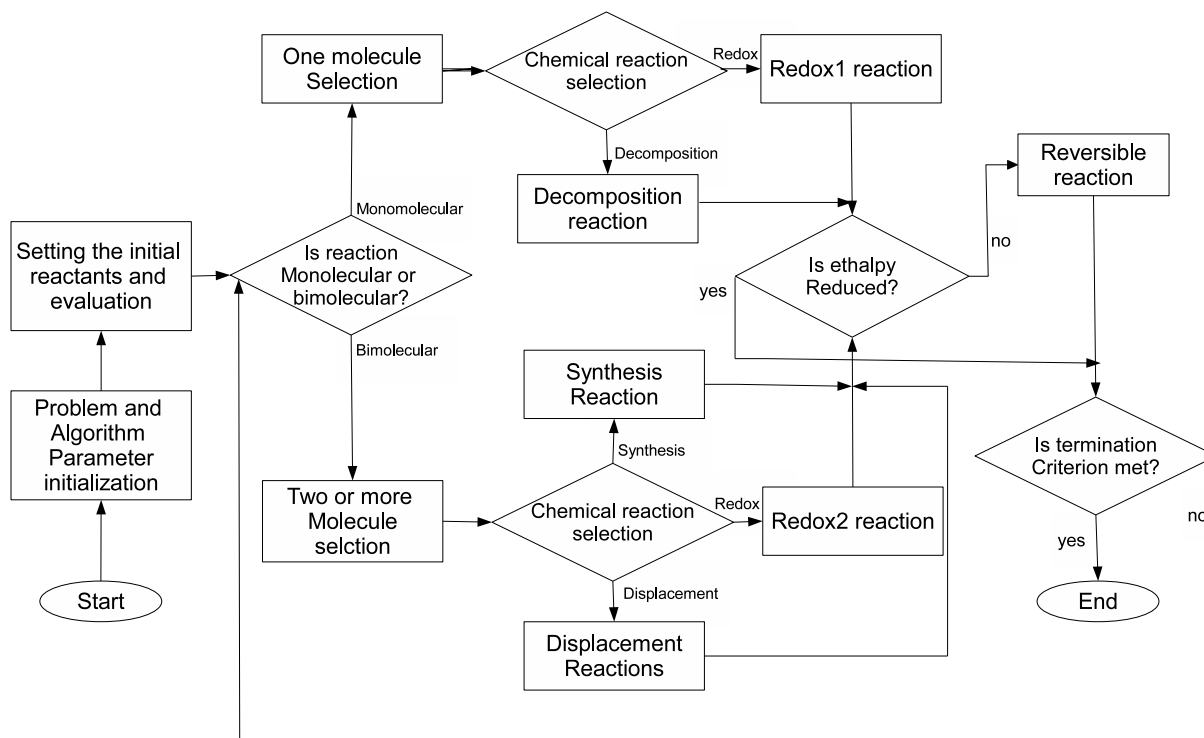
Fig. 1: ACROA flow chart [8]

Algorithm 2 $synthesis(x_1, x_2)$

Input: Reactants x_1 and x_2
for $i \leftarrow 1$ to n **do**
    Get t randomly in [0, 1]
    if $(t > 0.5)$ **then**
        $x'(i) \leftarrow x_1(i)$
    else
        $x'(i) \leftarrow x_2(i)$
    end if
end for
Output: x'

4.3.2 Displacement

This operator creates two new reactants from two original reactants. Each position of the two reactants strings are considered for information swapping based on a randomly generated mask similar to the mask used in uniform crossover used in genetic algorithms [8]. At the position where the mask value is 0, the reactants values are exchanged, otherwise the reactant values does not change.

4.3.3 Redox2

The two-points crossover is commonly used in genetic algorithm is used. This operator responds to intensifications.

4.3.4 Decomposition

Two random points in the reactant string are selected and the values between those points are reversed. This operator responds to diversifications of the algorithm.

4.3.5 Redox1

This operator is implemented for diversifications. One new reactant (solution) is generated from one original reactant. One position i^{th} is randomly selected from $\{1, \ldots, m\}$, and value of y_i is replaced by a random number in $\{1, \ldots, n_i\}$.

4.4 Reactants update

This step inspired reversible chemical reactions, chemical equilibrium test is performed. If the newly generated reactants give better function value, the new reactant set is included and the worse reactant is excluded similar to reversible chemical reactions. It helps the reactants move towards optima.

4.5 Termination criterion check

When the given termination criterion is met, the ACROA report the best solution. Otherwise, the chemical reaction process is repeated.

5. Experiment and analysis

5.1 Data test set

One type of randomly generated data instances are considered, each instance tested with data-range $R = 1000$ for different number of classes m and sizes n_i:

- Strongly correlated data instances (SC): In knapsack problem w_j is randomly generated in [1, R] and $c_j = w_j + 10$. For each class i generate n_i items (w'_j, c'_j) as for knapsack problem, and order these by increasing weight. The data instance for MCKP is then $w_{ij} = \sum_{h=1}^{j} w'_h$ and $c_{ij} = \sum_{h=1}^{j} c'_h$, $j = 1, 2, \ldots, n_i$. Such instances have no dominated items, and form an upper convex set.

  For each instances, the capacity W is calculated as follows.

$$W = \frac{1}{2} \sum_{i=1}^{m} (min_{j \in N_i}(w_{ij}) + max_{j \in N_i}(w_{ij}))$$

5.2 Parameter setting

GA's parameters are set as in [7]: $Popsize = 20$, $Pc = 0.8$, $Pm = 0.1$.

For ACROA, the $reactantNum$ is set to 20.

The terminated criteria is set the same for GA and ACROA that is function evaluations.

5.3 Experiment results

All the algorithms were implemented in Matlab R2011b. The test environment is set up on a personal computer with Pentium E6700 CPU at 3.2 GHz CPU, 2G RAM, running on Windows XP.

We have considered how the algorithm behaves for different problem sizes, test instances, and data-ranges.

We observe the convergence curves of three test instances in strongly correlated test set. The three instances with $(m = 10, n = 10)$, $(m = 100, n = 100)$, and $(m = 1000, n = 100)$ are used in this experiment. Figure 3 shows the evolution of the mean of the best total costs of CRO over 25 runs in the three instances. It indicates the global search ability and the convergence ability of CRO. There are several observations and they are given as follows:

In the case $(m = 10, n = 10)$, as depict in a) the convergence curve of GA is tied with CRO's, but CRO is still better. For instance $(m = 100, n = 100)$, the Fig. 3b) shows that CRO have a much more quick convergence compares with GA. For larger instance $(m = 1000, n = 100)$, the Fig. 3c show that CRO still have a good convergence, while GA
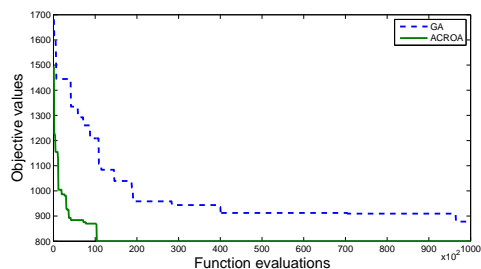
(a) $m = 10, n = 10$
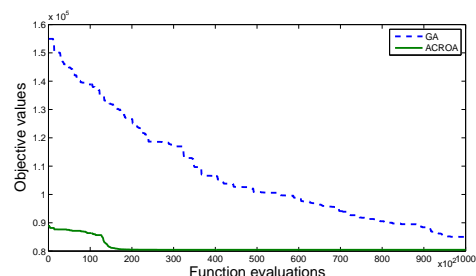
(b) $m = 100, n = 100$
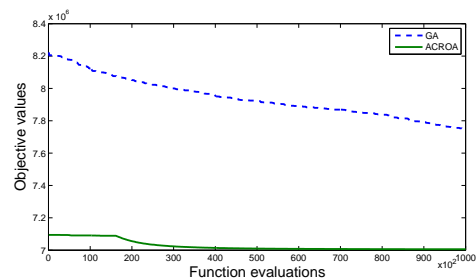
(c) $m = 1000, n = 100$

Fig. 3: Convergence curves of GA and ACROA on the MCKP. The objective function value were averaged over 25 runs.

shows a very slow convergence. From the Fig. 3 shows that CRO much more better GA in convergent rate and solution quality when solving large MCKP.

The simulation results are shown in table 1. m and n are number of class and number of items in a class, respectively. time and stdDev represent the elapsed time per run in second and the standard deviation, respectively. mean, worst and best are mean cost, worst cost and best cost, respectively. The maximum number of function evaluations 150 000, the number of runs 25.

Table 1 shows the experimental results of the strongly correlated instances. For all the proposed instances, CRO yields superior results compared with GA. The series of experimental results demonstrate the superiority and effectiveness of CRO. In comparison with GA; CRO can get better results in shorter time. The smaller standard deviation (StdDev) shows that the new algorithm is more robust than

Table 1: Simulation results for strongly correlated instances.

Instances		GA					CRO				
m	n	mean	best	worst	stdDev	time	mean	best	worst	stdDev	time
10	10	865	859	880	1.00	7.08	861	856	870	0.04	7.36
10	100	1621	1536	1764	2.00	13.90	1423	1421	1425	0.20	7.38
10	1000	1862	1617	1907	7.28	15.74	1247	1237	1265	0.28	7.66
100	10	1504	1344	1689	20.03	20.66	9192	9183	9198	0.03	8.97
100	100	16157	14348	18006	32.20	23.12	10012	9982	10075	22.20	8.47
100	1000	18122	15157	21757	61.04	23.94	14228	15190	16346	31.04	9.08
1000	10	124006	115035	142071	233.22	32.15	98003	97935	98493	133.22	17.89

GA.

6. Conclusion

The new approach artificial chemical reaction optimization algorithm using encoding integer string is proposed to solve multiple-choice knapsack problem. Five specific problem reaction operators are presented. A new penalty function is suggested to treat the infeasible solutions. The experiment on a large range of data set demonstrates that the proposed method has superior performance when compared with GA. It has shown a significant potential in solving MCKP. In the future, we will development a parallel version of this algorithm that improved the efficiency.

References

[1] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. America: W.H. Freeman, 1979.

[2] R. M. Nauss, "The 0Ű1 knapsack problem with multiple choice constraints," *European Journal of Operational Research*, vol. 2, no. 2, pp. 125 – 131, 1978.

[3] P. Sinha and A. A. Zoltners, "The multiple-choice knapsack problem," *Operational Research*, vol. 27, no. 3, p. 503, 1979.

[4] T. Zhong and R. Young, "Multiple choice knapsack problem: Example of planning choice in transportation," *Evaluation and Program Planning*, vol. 33, no. 2, pp. 128 – 137, 2010.

[5] P. C. Yue and C. K. Wong, "Storage cost considerations in secondary index selection," *International Journal of Parallel Programming*, vol. 4, pp. 307–327, 1975.

[6] M. L. Fisher, "The lagrangian relaxation method for solving integer programming problems," *Manage. Sci.*, vol. 50, no. 12 Supplement, pp. 1861–1871, Dec. 2004.

[7] M. R. Gen, R. Cheng, M. Sasaki, and Y. Jin, *Multiple-choice knapsack problem using genetic algorithms*. Maui, HI: Integrated Technology Systems, 1998.

[8] B. Alatas, "Acroa: Artificial chemical reaction optimization algorithm for global optimization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13 170 – 13 180, 2011.

[9] B. Alatas, "A novel chemistry based metaheuristic optimization method for mining of classification rules," *Expert Systems with Applications*, vol. 39, no. 12, pp. 11 080 – 11 088, 2012.

[10] T. K. Truong, K. Li, and Y. Xu, "Chemical reaction optimization with greedy strategy for the 0-1 knapsack problem," *Appl. Soft Comput.*, vol. 13, no. 4, pp. 1774–1780, Apr. 2013.

Generation Expansion Planning base on Modified Shuffled Frog Leaping in Restructured Environment

O. Abedinia
Electrical Engineering Department,
Semnan University, Semnan, Iran

N. Amjady
Electrical Engineering Department,
Semnan University, Semnan, Iran

H. A. Shayanfar [*]
Center of Excellence for Power
System Automation and Operation,
Elect. Eng. Dept., Iran University of
Science and Technology, Tehran, Iran

oveis.abedinia@gmail.com, n_amjady@yahoo.com, hashayanfar@yahoo.com

Abstract— **In this paper a generation expansion planning in restructured power systems is presented based on Modified Shuffled Frog Leaping (MSFL) for decoupling generation expansion planning from transmission expansion planning. Also for reducing complexity of the problem the benders decomposition is applied in this paper which divide the main problem in to two sub-problems as; maximize profits of each GENCO (PBGEP) and the security problem wants to satisfy security network constraints (SCGEP). Accordingly, calculate value of each GENCO's profit and total profit are considered as an optimization problem by proposed MSFL. The effectiveness of the proposed technique is applied over modified IEEE 30-bus system. The presented results demonstrate the efficiency the proposed technique.**

Keywords: MSFL, Generation Expansion Planning, Benders Decomposition, Security-Constrained Generation Expansion Planning.

I.   INTRODUCTION

The main objective of power system planning in regulated power systems is to meet the request of loads, while maintaining power system reliability. In this surrounding uncertainty is low. Transmission expansion planning is centralized and coordinated with generation expansion planning. Planners have availability to the required information for planning. Therefore, designer can design the least cost transmission plan based on the certain reliability criteria [1].

Composite power system expansion planning with open access to the transmission system has become a hot issue in the electricity energy industry in recent years [2-4]. Electric market access has moved the industry from conventional monopolistic electricity markets to competitive markets [4-6]. In a competitive market, the price of the delivered energy and the quality of energy supply including voltage quality and reliability of service are the main factors for business success. A key factor in today's competitive environment is the orientation toward customer's needs and willingness to pay for quality [4]. Composite system expansion planning addresses the

problem of broadening and strengthening an existing generation and transmission network to optimally serve a growing electricity market while satisfying a set of economic and technical constraints [5-6]. The problem is to minimize the cost subject to a reliability level constraint. Various techniques including branch and bound, sensitivity resolution, Bender decomposition, Simulated Annealing (SA), Genetic Algorithms (GA), Tabu Search (TS), and Greedy Randomized Adaptive Search Procedure (GRASP) have been used to study the problem [7-8].

In this paper Generation Expansion Planning (GEP) is discussed. GEP is one of the strategic planning for every country. In traditional environment, the purpose of GEP was to minimize system cost (investment & operation costs) while satisfy system load. Restructuring in power systems change many previous concept and redefine these. In restructured environment each GENCO wants to maximize its profit while ISO surveys system reliability and security [2].So, moreover size, place, type and construction time of new units, market price and profit of each GENCO have been considered. Therefore, GEP is a mixed-integer nonlinear problem with several constraints [9]. Solving this problem is very difficult. In addition, relation between GEP and Transmission Expansion Planning (TEP), is other difficulty for planners [10].

Also, several optimization algorithms such as decomposition method [6], genetic algorithm [7-8], artificial neural networks [9], etc. have been applied to solve the complicated problem; benders decomposition method will match GEP framework very well. In this paper the benders decomposition is used for decoupling main problem to two sub-problems. For this purpose, the Modified Shuffled Frog Leaping (MSFL) is proposed in this paper to search and find best location of wind power. The features and the advantages of MSFL technique, such as escaping from local optima snares, global optimization, appropriate robustness, simple mechanism and quick convergence ability, would make MSFL technique as a promising optimization approach.

---
* Corresponding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

The effectiveness of the proposed technique is applied over modified IEEE 30-bus system and analyzes output results. The presented results demonstrate the robustness of the proposed technique.

II. Model Description

The remaining assumptions are considered in this paper:

- The forecasted load duration curve for a planning year is divided into multiple load blocks. As depicted in Figure 1, we use three load blocks of peak, medium, and base.

- The marginal cost of the most expensive committed unit, which satisfies the network security, will be the pseudo market clearing price (MCP). A security constrained unit commitment (SCUC) algorithm is used here for the simulation of MCP.

Figure 2 shows decomposition scheme for solving GEP in restructured power systems.
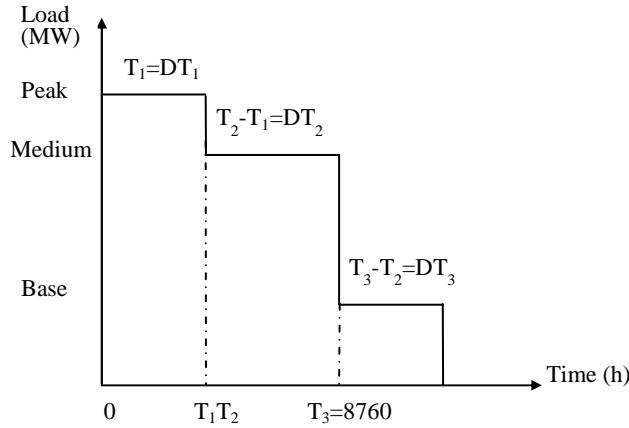
Figure 1.   Load duration curve for a planning year

A. Priced-Based Generation Expansion Planning

After determining expansion value of each GENCOs with considering future load, each GENCO should start t's planning process to maximize its profit. Profit in this paper defined as different between revenue from cost. Before GENCOs begin to plan. They should have generator data. this data that is for existing units and candidate units, included: unit type(gas, coil, etc), unit size, limit on the added unit per type, unit life, construction time, fixed cost of each unit (investment cost and fixed operation & maintenance costs), variable cost of each unit (operation costs), for of each unit, co2 emission of each GENCO. PBGEP is formulated as follows:

$$\min Y = -E\{\sum_{t=1}^{T}\sum_{b=1}^{B}DT_{bt} * [\sum_{i=1}^{NG}(R_{bt} * P_{G,ibt} - OC_{ibt} * P_{G,ibt}) -$$

$$\sum_{j=1}^{ND}CC_{jbt} * P_{C,jbt}]\} - \sum_{t}^{T}\sum_{i}^{CG}CI_{it} * (X_{it} - X_{i(t-1)})$$

$$-\left(\sum_{t}^{T}\sum_{i}^{CG}[FOM_{i} * X_{it}] - \sum_{t}^{T}\sum_{i}^{EG}[FOM_{i} * 1]\right) \tag{1}$$

$$\sum_{j=1}^{CG}CI_{it} * (X_{it} - X_{i(t-1)}) \le CI_{t}, t = (1,2,..., T) \tag{2}$$

$$\sum_{j=1}^{CG}cap_{it} * (X_{it} - X_{i(t-1)}) \le ULAC_{t}, t = (1,2,..., T) \tag{3}$$

$$\sum_{j=1}^{CG}(X_{it} - X_{i(t-1)}) \le ULAC_{t}, t = (1,2,..., T) \tag{4}$$

$$X_{it} = 0, if ; t < CT_{i}, (i = 1,2,..., PG)(t = 1,2,..., T) \tag{5}$$

$$\sum_{i=1}^{EG}cap * E_{it} + \sum_{i=1}^{CG}cap_{i} * (X_{it} - X_{i(t-1)}) \ge P_{D,bt} + P_{R,bt} \tag{6}$$

$$(b = peak\ load\ block)\quad(t=1,2,...,T)$$

$$X_{A1t} + X_{B1t} \le 1 \tag{7}$$

$$X_{A1t} = X_{A2t} = ... = X_{Amt}\ (t = T) \tag{8}$$

$$X_{A1t} = X_{A2t} = ... = X_{Ant}\ (t = T) \tag{9}$$

$$(A_1, A_2, ...,A_m \in A\ combination)$$
$$(B_1, B_2, ...,B_m \in B\ combination)$$

Outputs of this section are type, size and timing for the adding of new units that are as inputs of the next section.

B. Security-Constrained Generation Expansion Planning

This section includes four sub-problems itself as follows:

1. *MSFL sub-problem:*

In this sub-problem, ISO check planning bids of each GENCO to make sure that these are in acceptable level. If plans of each GENCOs are in the confine of *MSFL* output, ISO pass those plans and run next sub-problem. Otherwise, ISO should send a security signal to GENCOs for improving their plans. This process will be repeated until reach ISO desired plans.

2. *Feasibility sub-problem:*

Feasibility sub-problem formulated as follow,

$$DT_{bt} * \sum_{j=1}^{ND}P_{C,jbt} \le \kappa_{bt} \tag{10}$$

$$E\{DT_{bt} * \sum_{j=1}^{ND}P_{C,jbt}\} \le \varepsilon_{bt} \tag{11}$$

$$\min\ E\{DT_{bt} * \sum_{j=1}^{ND}P_{C,jbt}\} \tag{12}$$

$$\sum_{i=1}^{NG}P_{C,jbt}(\varphi) + \sum_{j=1}^{ND}P_{C,jbt}(\varphi) = P_{L,bt}(\varphi) \tag{13}$$
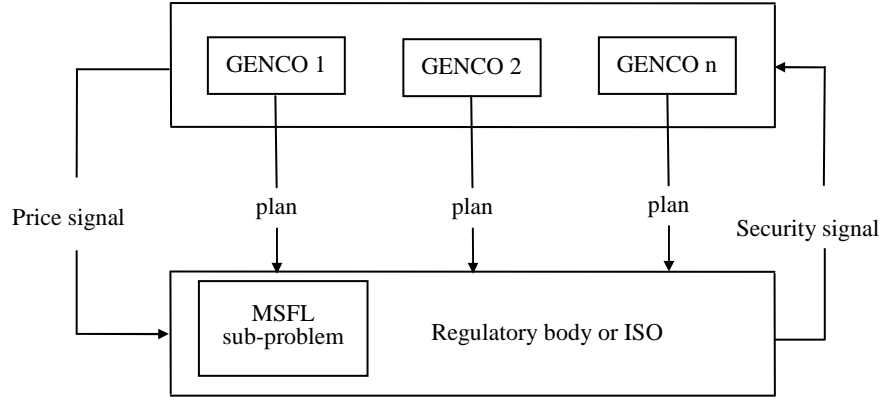
Figure 2.   Generation expansion planning framework

$$P_{L,kbt}(\varphi) = D_{km}(\kappa) * [P_{G,ibt}(\varphi) + P_{C,ibt}(\varphi) - P_{D,jbt}(\varphi)], i, j \in m \quad (14)$$

$$0 \leq P_{C,jbt}(\varphi) \leq P_{D,jbt}(\varphi) \quad (15)$$

For existing units,

$$P_{Gi,\min} * E_{it} \leq P_{G,jbt}(\varphi) \leq P_{Gi,\max} * (E_{it}) \quad (16)$$

For candidate units,

$$P_{Gi,\min} * X_{it} \leq P_{G,jbt}(\varphi) \leq P_{Gi,\max} * (X_{it})$$

$$\left| P_{L,kbt}(\varphi) \right| \leq PL_{k,\max} \quad (17)$$

If constraints (11) or (12) are not satisfied, corresponding benders cut will be generated as follow:

- If $DTbt * \Sigma^{ND}_{j=1} PC,jbt \leq \kappa bt$ is not satisfied, the Bender cut is:

$$E\{DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt}\} + \sum_{i=1}^{CG} \lambda_{ibt}^{n} P_{Gi,\max}(X_{it} - X_{it}^{n}) \leq \varepsilon_{bt}$$

$$(18)$$

These $n = 1, 2, 3, \ldots, N-1$ Bender cuts from the previous iterations are added to the master problem of resource planning to get the nth trial investment plan. The process will be repeated until a feasible plan is found for meeting the ISO's requirement on system reliability.

3. Security-constrained unit commitment (SCUC) sub-problem:

Before solving the operation sub-problem, we calculate the electricity MCP over the planning horizon. We assume the MCP is the marginal cost of the most expensive unit among committed units based on the network security [11]. SCUC is formulated as follows, the violations persist, the corresponding Bender cut will be generated as follows:

$$\min \ F = \sum_{j=1}^{NG} OC_{jbt} * P_{G,ibt} \quad (19)$$

$$\sum_{i=1}^{NG} P_{G,ibt}(\varphi) + \sum_{j=1}^{ND} P_{C,jbt}(\varphi) = P_{L,bt}(\varphi) \quad (20)$$

$$\sum_{i=1}^{NG} P_{Gi,\max} * I_{it} \geq P_{L,bt} + P_{R,bt} \quad (21)$$

$$P_{L,kbt}(\varphi) = D_{km}(\varphi) * [P_{C,jbt}(\varphi) + P_{C,jbt}(\varphi) - P_{D,jbt}(\varphi)], j \in m \quad (22)$$

$$\sum_{i=1}^{NG} P_{G,ibt} = P_{L,bt} \quad (23)$$

$$\sum_{t=1}^{T} \{\sum_{i=1}^{NG} P_{G,ibt} E_{ibt}\} \leq \sigma_{bt} \quad (24)$$

$$0 \leq P_{C,jbt}(\varphi) \leq P_{D,jbt}(\varphi) \quad (25)$$

For existing and candidate units,

$$P_{Gi,\min} * I_{it} \leq P_{G,ibt} \leq P_{Gi,\max} * I_{it} \quad (26)$$

$$E\{DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt}\} \leq \varepsilon_{bt} \quad (27)$$

$$DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt} \leq \kappa_{bt} \quad (28)$$

The violations persist, the corresponding Bender cut will be generated as follows:

- If $DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt} \leq \kappa_{bt}$ cannot be satisfied, the Bender cut is:

$$E\{DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt}\} + \sum_{i=1}^{CG} \mu_{ibt}^{n} P_{Gi,\max}(I_{it} - I_{it}^{n}) \leq \varepsilon_{bt}$$

$$(29)$$

- If $E\{DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt}\} \leq \varepsilon_{bt}$ cannot be satisfied, the Bender cut is:

$$E\{DT_{bt} * \sum_{j=1}^{ND} P_{C,jbt}\} + \sum_{i=1}^{CG} \mu_{ibt}^{n} P_{Gi,\max}(I_{it} - I_{it}^{n}) \leq \varepsilon_{bt}$$

$$(30)$$

Then the master problem of SCUC is solved iteratively to provide a least-cost generation schedule, while supplying the load demand.

4. Optimal operation sub-problem

After meeting the desired system reliability level and calculating the MCP, the optimal operation sub-problem for every planning scenario, year, and load block is formulated as follows:

$$\min W_{st} = -E\{\sum_{t=1}^{T}\sum_{b=1}^{B}DT_{bt} *[\sum_{i=1}^{NG}R_{bt}*P_{G,ibt} - OC_{ibt}*P_{G,ibt}) - \sum_{k=1}^{NL}TC_{k}*|P_{L,kbt}$$

$$-\sum_{j=1}^{ND}(CC_{jbt}*P_{C,jbt})]\} \tag{31}$$

For existing units which are committed,

$$P_{Gi,\min} \le P_{G,ibt}(\varphi) \le P_{Gi,\max} \tag{32}$$

For candidate units which are committed,

$$P_{Gi,\min} * X_{it} \le P_{G,ibt}(\varphi) \le P_{Gi,\max} * X_{it}$$

$$\left|P_{L,kbt}(\varphi)\right| \le PL_{k,\max} \tag{33}$$

The optimal operation cut associated with the nth trial solution is:

$$Z \le \sum\sum_{b}\{W_{bt}^{n} + [\sum_{i=1}^{CG}CI_{it}*(X_{it} - X_{i(t-1)}) + \Pi_{ibt}^{n}P_{Gi,\max}(X_{it} - X_{it}^{n})]\} \tag{34}$$

III. MSFL Technique

Shuffled Frog Leaping (SFL) algorithm, has been introduced by Eusuff and Lansey for water distribution system optimization, is a meta-heuristic for solving optimization problems [12]. SFLA is a decrease based stochastic search method that begins with an initial population of frogs whose characteristics, known such as memes, represent the decision variables. For this algorithm, the individual frogs are not the important parts; rather they are seen as hosts for memes and described as a memetic vector [13].The algorithm uses memetic evolution in the form of influencing of ideas from one individual to another in a local search. In SFL, the population consists of a set of frogs (solutions) partitioned into subsets, referred to as memeplexes. Actually the different memeplexes are considered as different cultures of frogs, each performing a local search [14].Hence, the positions of the frogs are presented as:

$$D_{i} = rand \times (X_{b} - X_{g}) \tag{35}$$

And for new position:

$$X_{i+1} = X_{i} + D_{i}, -D_{\max} \le D_{i} \le D_{\max} \tag{36}$$

Where,

$rand$ = random number between 0 and 1.
$D_{\max}$ = the maximum allowed change in a frog's position.

A. Modified Shuffled Frog Leaping Algorithm

It is difficult for SFLA algorithm to overcome local minima when handling some complicate functions [12]. MSFLA starts with an initial population of "X" frogs created randomly like other evolutionary algorithms. The whole population of frogs is then partitioned into subsets referred to as memeplexes. The various memeplexes are considered as different cultures of frogs. And they are located at different places in the solution space (i.e., global search). Each culture of frogs performs a deep local search. Within each memeplex, the solo frogs hold information that can be influenced by the information of their frogs within their memeplex, and evolve through a procedure of shift of information among frogs from

different memeplexes [14]. After a defined number of evolutionary steps, information is passed among memeplexes in a shuffling process. The local search and the proposed processes continue until a defined convergence criterion is satisfied.

It is necessary to note that the mutation vector dimension is equal to the memeplexes number. Therefore, a position changing formula turns to the following form.

$$D_{i} = rand \times C \times (f(X_{b}) - f(X_{w})) \times (X_{b} - X_{w}) \tag{37}$$

And for new position:

$$X_{i+1} = X_{i} + D_{i} \tag{38}$$

Where,
$C \in [0, C_{max}]$,
C_{max} = case dependant upper limit
$f(X_b)$= The best fitness functions that are found by the frogs in each memeplex.
$f(X_w)$= The worst fitness functions that are found by the frogs in each memeplex.
Similar to the original *SFL*, if the process produces a better solution, the worst frog is replaced by the better one.

IV. CASE STUDY

The IEEE 30-bus system depicted in Figure 3 has 44-lines, 21-demand sides and 5-GENCOs with seven existing units and 22 Candidate units. The data for generators, forecasted peak demand with a load growth rate of 5%, available investment over a 15 year planning horizon, as well as the possible sites and types of candidate units and other information about the system are given in [15]. The initial construction cost represents the cost of construction at year 1. We assume that the construction cost will increase by 3% per year based on inflation.
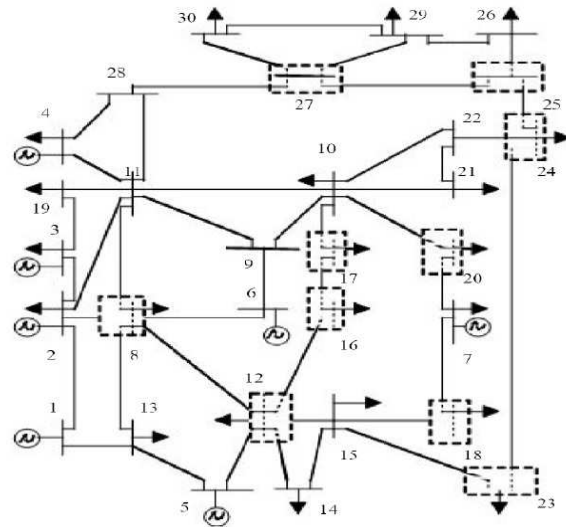
Figure 3.   IEEE 30-bus test system

In this study, the existing units are 4 years old each, have a useful life of 20 years, and will be out of service

after 16 years. The forecasted reserve for a planning year is 500 MW (the largest unit available). The ISO will use

this figure in maximizing the social welfare and minimizing the cost of supplying the load.

TABLE I. OUTPUTS OF MSFL ALGORITHM - CONTRIBUTION OF EACH GENERATOR TO LOAD

Load bus number	Load value (MW)	Contribution of generator to load (MW)						
		G_1	G_2	G_3	G_4	G_5	G_6	G_7
8	140	17.112	30.661	18.123	38.822	16.177	14.240	4.865
10	160	2.443	13.212	12.443	30.984	20.181	30.363	50.374
12	100	2.554	8.272	4.387	10.345	45.653	5.234	23.555
13	120	30.098	8.382	11.345	25.534	17.347	6.716	20.578
14	60	3.873	23.234	3.556	7.663	16.837	3.882	0.955
15	80	1.234	4.211	3.532	7.908	25.548	5.374	32.193
16	60	1.654	4.234	3.098	7.097	20.759	5.474	17.684
17	80	1.567	4.098	3.786	15.345	15.393	12.098	27.713
18	120	1.453	5.235	5.984	5.345	15.938	1.445	84.600
19	120	1.521	2.554	102.013	8.124	0.553	0.565	4.670
20	80	0.563	3.012	1.345	2.887	3.255	2.456	66.482
21	80	0.563	1.334	5.574	15.181	12.987	12.187	32.174
23	100	2.112	5.445	6.994	15.383	25.985	7.038	37.043
24	80	2.445	7.432	5.093	15.987	10.543	12.984	25.516
26	160	5.873	15.332	18.736	63.987	15.983	14.938	25.151
29	60	1.644	4.987	7.377	25.276	5.543	4.224	10.949
30	80	2.590	8.353	9.883	35.094	5.098	5.987	12.995
Total generation (MW)		79.299	149.988	223.269	330.962	273.78	145.205	477.497

TABLE II. OUTPUTS OF MSFL ALGORITHM FOR THE ALL OF BUSES

Bus Number	Load value (MW)	MSFL Output
8	140	335.828
9	0	221.993
10	100	125.374
11	0	290.287
12	60	402.277
13	80	130.338
14	60	95.223
15	80	95.019
16	120	519.223
17	120	534.388
18	80	520.763
19	80	80.134
20	100	501.244
21	80	120.918
22	0	118.982
23	60	355.837
24	80	298.298
25	0	240.198
26	160	265.982
27	0	226.287
28	0	198.287
29	60	215.127
30	80	210.765

TABLE III. OUTPUTS OF GEP ALGORITHM - NEW UNITS OVER THE 15-YEAR HORIZON

Planning Year	New unit added in each planning year
1	-
2	21,20
3	-
4	17
5	-
6	20
7	-
8	26
9	-
10	-
11	-
12	10
13	-
14	-
15	-

TABLE IV. OUTPUTS OF GEP ALGORITHM - VALUE OF EACH GENCO'S PROFIT AND TOTAL PROFIT

Genco1's profit	Genco2's profit	Genco3's profit	Genco4's profit	Genco5's profit	Total Profit
260.812	840.117	416.287	0	120.762	1637.978

At first, we calculate DLG matrix by MSFL. Where, the achieved results are presented in Table 1. The Table1 shows contribution of each generator to satisfy considered load. The values for all the newly added generations are computed and compared as shown in Table 2. Accordingly, GENCOs with considering these data and other necessary data for planning run PBGEP problem to

reach best plan that maximize its profit and submit it to ISO. Then, ISO with running SCGEP will accept or reject GENCO's proposed plans. Results of running this algorithm on supposed system are shown in Table 3. Table 4 shows value of each GENCO's profit and total profit. With comparing these results and outputs of MSFL, shows that selected units at planning horizon are in acceptable level.

IV. CONCLUSIONS

This paper presents a generation expansion planning in restructured power systems is presented based on Modified Shuffled Frog Leaping (MSFL) for decoupling generation expansion planning from transmission expansion planning. Accordingly, calculate value of each GENCO's profit and total profit are considered as an optimization problem by proposed MSFL. The effectiveness of the proposed technique is applied over modified IEEE 30-bus system. The presented results demonstrate the efficiency the proposed technique. With comparing these results and outputs of MSFL, shows that selected units at planning horizon are in acceptable level. In other word, with investing on this unit we ensure that can satisfy load without any congestion in transmission lines.

NOMENCLATURES

Indices:

I	Existing or candidate unit
j	Load point
k	Transmission line
m	Bus
n	Trial
b	Load block
t	Planning year
φ	Scenario

Parameters:

B	Number of load blocks
Cap_i	Capacity of unit i
CT_i	Required construction time for candidate unit i
C_{it}	Capital investment in year t
CI_{it}	Capital investment for candidate unit i in year t
CC_{jbt}	Curtailment cost coefficient for load j at load block b in year t
CG	Number of candidate units
CS	Set of candidate sites
D_{km}	Sensitivity of line k flow to generation at bus m
DT_{bt}	Duration of load block b in year t
EG	Number of existing units
FOM_i	Fixed O&M cost of unit i
TC_k	Transmission charge of line k
ND	Number of load points
NG	Number of committed units
NL	Number of transmission lines
OC_{ibt}	Operating cost unit i among committed units at load block b in year t
E_{ibt}	CO2 emission of unit i among committed units at load block b in year t
PL,bt	required system load at load block b in year t
PD,jbt	Forecasted load point j at load block b in year t
PD,bt	Forecasted system load at load block b in year t
PG_i,	min Lower limit of generation of unit i
PG_i,	max Upper limit of generation of unit i
PR,bt	Forecasted system reserve at load block b in year t
PL_k,	max Capacity of line k
T	Planning horizon
$ULAU_t$	Upper limit for the # of units added in year t
$ULAC_t$	Upper limit for generating capacity added in year t

δ_{bt}	Acceptable limit of co2 emission at load block b in year t
κ_{bt}	Acceptable level of curtailment at load block b in year t
ε_{bt}	Acceptable level of EENS
Δ	Convergence threshold

Variables:

E_{it}	State variable associated with existing unit i in year t; 1: on service, 0: out of service
I_{ibt}	Commitment of unit i at load block b in year t; 1:committed, 0: decommitted/out of service
R_{bt}	Electricity sale price at load block b in year t
PC, jbt	Curtailment of load j at load block b in year t
PG,ibt	Dispatched capacity of committed unit i at load block b in year t
PL,kbt	Line k flow at load block b in year t
X_{it}	State variable associated with candidate unit i in year t; 1: selected, 0: rejected. $(X_{i(t-1)} \_ X_{it})$

Dual variables:

λ^n_{ibt}	Marginal decrease in unserved energy with a 1MW increase in candidate unit i at load block b in year t associated with the nth trial plan.
μ^n_{ibt}	Marginal decrease in unserved energy with a 1MW increase in commitment unit i at load block b in year t associated with the nth trial plan.
π^n_{ibt}	Marginal increase in profit with a 1 MW increase in candidate unit i at load block b in year t associated with the nth trial plan.

REFERENCES

[1] R. Fang, D. J. Hill, "A New Strategy for Transmission Expansion in Competitive Electricity Markets," IEEE, Trans. on PS, vol.18, no.1, pp.374-380, Feb. 2003.

[2] D. Hongwei, Y. Yixine, H. Chunhua, X. Jian, Z. Yi, X.Rui, "Optimal Planning of Distribution Substation Location and Size-Model and Algorithm," Electrical Power and Energy System, Vol. 13, No. 6, pp. 353-357, Aug. 1996.

[3] Y. Fu, Z. Li, M. Shahidhpour, "Profit-Based Generation Resource Planning", IMA J. of Management Mathematics, Vol. 15, Issue 4, pp. 273-289, 2004.

[4] W.S Read, W.K. Newman, I.J. Perez-Arriaga, H.Rudnick, M.R. Gent & A.J. Roman, "Reliability in the New Market Structure(Part2)," IEEE Power Engineering Review, pp.10-16, January, 2000.

[5] H. Teixeira Firmo, L. Fernando Loureiro Legey, "Generation Expansion Planning: An Iterative Genetic Algorithm Approach", IEEE Transactions on Power Systems, Vol. 17, No. 3, pp. 901-906, August 2002.

[6] J. HyungRoh, M. Shahidehpour, Y. Fu, "Market-Based Coordination of Transmission and Generation Capacity Planning", IEEE Transactions on Power Systems, Vol. 22, No. 4, pp. 1406-1419, November 2007.

[7] T. Dhadbanjan, V. Chintamani, "Evaluation of Suitable Locations for Generation Expansion in Restructured Power Systems: A Novel Concept of T-Index", International Journal of Emerging Electric Power Systems, Vol. 10, Issue1, pp. 1-24, April 2009.

[8] J. Sirikum, A. Techanitisawad, V. Kachitvichyanukul, "A New Efficient GA-Benders' Decomposition Method: For Power Generation Expansion Planning With Emission Controls", IEEE Transactions on Power Systems, Vol. 22, No. 3, pp. 1092-1100, August 2007.

[9] J.B. Park, Y.M. Park, J.R. Won, K.Y. Lee, "An Improved Genetic Algorithm for Generation Expansion Planning", IEEE

Transactions on Power Systems, Vol. 15,No. 3, pp. 916-922, August 2000.

[10] H. Teixeira Firmo, L. Fernando LoureiroLegey,"Generation Expansion Planning: An Iterative GeneticAlgorithm Approach", IEEE Transactions on PowerSystems, Vol. 17, No. 3, pp. 901-906, August 2002.

[11] H. Sasaki, J. Kubokawa, M. Watanabe, R. Yokoyama, R.Tanabe, "A Solution of Generation Expansion Problem byMeans of Neural Network", First International Forum onApplications of Neural Networks to Power Systems, pp.219-224, July 1991.

[12] L. Noakes and R. Kozera,"A 2D Leap-Frog algorithm for optimal surface reconstruction", Proc. 44th Annual Meet. Opt. Eng. SPIE'99, III-3811, pp:317–328, 1999.

[13] L. Noakes and R. Kozera. "Denoising images: Non-linear leap-frog for shape and lightsource recovery", Chapter in Theoretical Foundations of Computer Vision Geometry, Morphology and Computational Images, pp: 419–436, Lecture notes in Computer Science 2616, 2003.

[14] L. Noakes and R. Kozera,"Nonlinearities and noise reduction in 3-ource photometric stereo", J. Math. Imag. and Vis., vol. 2, no. 18, pp:119-127, 2003.

[15] J. HyungRoh, M. Shahidehpour, Y. Fu, "Security-Constrained Resource Planning in Electricity Markets",IEEE Transactions on Power Systems, Vol. 22, No. 2, pp.812-820, May 2007.

BIOGRAPHIES

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 410 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Distribution Planning with Renewable Energy Units based on Modified Shuffled Frog Leaping Algorithm

H. A. Shayanfar [*]
Center of Excellence for Power
System Automation and Operation,
Elect. Eng. Dept., Iran University of
Science and Technology, Tehran, Iran

O. Abedinia
Electrical Engineering Department,
Semnan University, Semnan, Iran

N. Amjady
Electrical Engineering Department,
Semnan University, Semnan, Iran

hashayanfar@yahoo.com, oveis.abedinia@gmail.com, n_amjady@yahoo.com

Abstract— **The Distributed Generation (DG) has created a challenge and an opportunity for developing various novel technologies in power generation. DG prepares a multitude of services to utilities and consumers, containing standby generation, peaks chopping sufficiency, base load generation. In this paper a planning paradigm for network upgrade based on Modified Shuffled Frog Leaping (MSFL) algorithm is proposed. SFLA is a decrease based stochastic search method that begins with an initial population of frogs whose characteristics, known such as memes, represent the decision variables. The algorithm uses memetic evolution in the form of influencing of ideas from one individual to another in a local search. The paradigm is able to select amongst several choices equi-cost that one assuring the optimum in terms of voltage profile, considering various scenarios of DG penetration and load demand. The proposed algorithm is applied over the 30 lines, 28 buses power system. The achieved results demonstrate the good efficiency of the DG using the proposed technique in different scenarios.**

Keywords: Distribution Generation, Renewable Energy Units, MSFLA.

I. INTRODUCTION

Distributed generation, for the moment loosely defined as small-scale electricity generation, is a fairly new emphasis in the economics literature about electricity markets, but the concept behind it is not new at all. Recently the DG, electricity generation was the statute, not the exclusion. The first power plants only reserved electricity to consumers in the close neighborhood of the generation plant. The first grids were DC based, and hence, the reserve voltage was confined, as was the interval that could be used between generator and customers. Equilibrium demand and supply was partially done by local storage, i.e. batteries, which could be instantly coupled to the DC grid [1]. Along with DG, local reservoir is also returning to the arena. Later, technical evolutions, in literature as the egression of AC grids, allowed for electricity to be transported over longer distances, and economies of measurement in electricity generation lead to an increase in the power output of the generation units [2]. All this resulted in increased comfort and lower per unit costs and massive electricity systems were manufactured, including of huge transmission and distribution grids and big generation plants. Balancing requirement and supply was done by the averaging efficacy of the combination of large amounts of instantaneously various loads. Safety of supply was increased as the fracture of one power plant was compensated by the other power plants in the interconnected system. Indeed this interconnected high voltage system create the economy of scale in generation possible [3].

Also, sizing and location of the new distributed power plants are chosen by the producers taking into account economical, environmental and legal aspects that do not necessarily correspond to the system's control needs in local distribution areas and to the distribution network operator (DNO) wishes [4]. On the other hand, the power injected by the power producers operates according to the market price signals and on the availability of primary energy sources, especially in presence of renewable energies, that introduce high uncertainty in power availability. All that suggests to DNO to apply a wary planning policy that avoids adverse effects on the utility feeders, especially on the voltage profile reducing, or even negating, the benefits

Actually various DG elections are quick becoming economically viable [5]. Technologies that employ classical energy sources include gas turbines, micro turbines and IC engines. Recently, the ones that show promises for DG requisitions are Wind Electric Conversion Systems (WECS), geothermal systems, solar-thermal–electric systems, photovoltaic (PV) systems and fuel cells. The [6] have argued the benefits of DG by evaluating and quantifying in terms of capacity credit, energy quantity and energy cost saving. The impacts of betterment in voltage profile and loss reduction were not considered in the method. The authors have shown the benefits of DG with power electronic interface to supply ancillary services such as reactive power, voltage recline compensation and harmonic filtering in [7]. It has proved the

---

capability of DG to compensate voltage sag resulting from faults in the power system. But the materiel did not analyze the amount of power loss reduction due to DG installation. An eventual approach based on complication technique to quantify the benefit of voltage profile improvement involving wind turbine generation have evaluated in [8]. In [9] a Monte Carlo–based method for the adequacy assessment of distributed generation systems has shown. In [10] the authors show the solutions to prevent sensitive equations from disruptive operation by making use of DG in the presence of voltage dips. Actually the synthesis of clustering methods and the convex hull algorithm for corrosion of large sets in renewable distributed generation has demonstrated in [11]. The non-iterative analytical approaches to determine the optimal location for placing DG in both radial and networked systems to minimize power losses is presented in [12]. All the mentioned strategies are mathematically modeled and hence are found to be complex in its approach.

In this paper at first section the problem formulation and the mathematical model is presented; in the second and third section the proposed MSFLA technique and the multi-scenario analysis are defined respectively. SFLA is a decrease based stochastic search method that begins with an initial population of frogs whose characteristics, known such as memes, represent the decision variables. For this algorithm, the individual frogs are not the important parts; rather they are seen as hosts for memes and described as a memetic vector. The algorithm uses memetic evolution in the form of influencing of ideas from one individual to another in a local search. In SFL, the population consists of a set of frogs (solutions) partitioned into subsets, referred to as memeplexes. Actually the different memeplexes are considered as different cultures of frogs, each performing a local search. In the last section some case studies are pointed out. Section five is concluded the paper.

II.   PROBLEM FORMULATION

For each scenario of DG influence and of load growth, the problem analyzed consists in selecting some lines for network upgrade in a set of feasible new lines to improve the voltage profiles, assigned the site of the new DG [13].

This is a binary integer non-linear programming difficulty, where the decision variables are the new lines to build for optimal network upgrade.

To simplify the mathematical formulation and the performance phase, the following formulation has been introduced. Each distribution network is represented as a set:

$$U_x = \left\{ B_x, L_{B_x} \right\} \tag{1}$$

where B_x is the set of network nodes

$$B_x = \left\{ b_1 ... b_{N_x} \right\} \tag{2}$$

With N_x the number of buses of network U_x and L_{B_x} the set of network lines connecting the nodes b_h and b_k of B_x.

$$L_{B_x} = \left\{ l_i = (b_h, b_k) \big| (b_h, b_k) \in B_x, i = 1 ... M_x \right\} \tag{3}$$

With M_x the number of lines of network U_x.

Let U_{START} the starting network

$$U_{START} = \left\{ B_{START}, L_{B_{START}} \right\} \tag{4}$$

Each assigned scenario of new DG penetration and load connection defines an addition of N_{NEW} buses to the N_{START} ones of B_{START}, obtaining a new set B_{NEW}

$$B_{NEW} = \left\{ b_1 ... b_{N_{START}}, b_{N_{START}+1} ... b_{N_{START}+N_{NEW}} \right\} \tag{5}$$

Obviously:

$$B_{START} \subseteq B_{NEW} \tag{6}$$

The set of all feasible lines for the upgrade associated to B_{NEW} is:

$$L_{B_{NEW}} = \left\{ L_{B_{NEW}}^1, L_{B_{NEW}}^2 ... L_{B_{NEW}}^O \right\} \tag{7}$$

Where:

$$L_{B_{NEW}}^O = \{ l_i = (b_h, b_k) \big| (b_h, b_k) \in B_{NEW}$$

$$i = 1 ... M_{START}, M_{START} + 1 ... M_{NEW}^o + M_{START} \} \tag{8}$$

With, M_{START} the number of lines of $L_{B_{START}}$ and $M_{NEW}^o + M_{START}$ the number of new lines of the o-th solution.

Obviously:

$$L_{B_{START}} \subseteq L_{B_{NEW}}^O \text{ with }, o \in [0...O] \tag{9}$$

The set of the network alternatives is:

$$U_{NEW} = \left\{ B_{NEW}, L_{B_{NEW}} \right\} \tag{10}$$

In order to identify and to select L_{BNEW}^o, to each line $l_i \in L_{BNEW}^o$ connecting the nodes b_h and b_k of B_{NEW} it is associated a binary variable $S^{LBoNEW}_{hk} = S^L_{hk}$ that defines the state of the line l_i.

The set of variables S^L_{hk} defines the following square matrix;

$$\tilde{N} \times \tilde{N}, \text{ with }, \tilde{N} = N_{START} + N_{NEW}$$

$$\underline{\underline{s}}^L = \begin{bmatrix} s^L_{1,1} & \cdots & s^L_{1,k} & \cdots & s^L_{1,\tilde{N}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ s^L_{h,1} & \cdots & s^L_{h,k} & \cdots & s^L_{h,\tilde{N}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ s^L_{\tilde{N},1} & \cdots & s^L_{\tilde{N},k} & \cdots & s^L_{\tilde{N},\tilde{N}} \end{bmatrix} \tag{11}$$

Where

$$s^L_{hk} = \begin{cases} 1, \text{if the line } l_i \text{ joining } b_h \text{ and } b_k \text{ exists} \\ 0, \quad \text{else} \end{cases}$$

The number of no null elements of $\underline{\underline{s}}^L$ is 2. $(M^o_{NEW} + M_{START})$. Furthermore, $\underline{\underline{s}}^L$ is a symmetric matrix with the elements of the main diagonal equal to zero. In particular:

$$\begin{cases} S^L_{hh} = 0, \forall h \leq \tilde{N} \\ S^L_{hh} = S^L_{kh}, \forall h, k \leq \tilde{N} \end{cases} \tag{12}$$

For each combination of switches, it is defined the state of a matrix $\underline{\underline{s}}^L$ that individualizes univocally the network:

$$U_{NEW}^O = \left\{ B_{NEW}, L_{B_{NEW}}^O \right\} \subseteq U_{NEW} \tag{13}$$

With the formalism just introduced, the problem is to identify for each scenario σ the optimal network:

$$U_{NEW_\sigma}^{OPT} = \left\{ B_{NEW_\sigma} , L_{B_{NEW_\sigma}}^{OPT} \right\} \in U_{NEW} \quad (14)$$

That minimizes the following objective function:

$$\min_{U_{NEW}} \Phi_\sigma \quad (15)$$

Where Φ_σ represents the square mean error of the bus voltage:

$$\Phi_\sigma = \frac{1}{\tau.(N_{START} + N_{NEW})} \sum_{i=1}^{\tau} \sum_{j=1}^{N_{START} + N_{NEW}} (V_{j,i} - V_{j,i}^{ref})^2 \quad (16)$$

Calculated for the assigned scenario σ, with τ number of points characterizing the current scenario [14]. The optimization problem is subject to the following system constraints:

$$P_{h,i} = \sum_k P_{hk,i}(V_{h,i}, V_{k,i}, \theta_{h,i}, \theta_{k,i})$$

$$Q_{h,i} = \sum_k Q_{hk,i}(V_{h,i}, V_{k,i}, \theta_{h,i}, \theta_{k,i}) \quad (17)$$

$$V_{min} \leq V_{h,i} \leq V_{max}$$

$$I_{l_i} \leq I_{l_i}^{max} \quad (18)$$

$$with : \begin{cases} h = 1...N_{START} + N_{NEW} \\ i = 1...\tau \end{cases} \quad (19)$$

Where V_{min} and V_{max} are the voltage limits and I_{li}^{max} is the current limit on the $l_i \in L_{B_{NEW_\sigma}}^{OPT}$ line [15].

Obviously, all network constraints, power injected and absorbed at the network buses are function of the scenario σ considered.

III. MSFL TECHNIQUE

Shuffled Frog Leaping (SFL) algorithm, has been introduced by Eusuff and Lansey for water distribution system optimization, is a meta-heuristic for solving optimization problems [16]. SFLA is a decrease based stochastic search method that begins with an initial population of frogs whose characteristics, known such as memes, represent the decision variables. For this algorithm, the individual frogs are not the important parts; rather they are seen as hosts for memes and described as a memetic vector [17]. The algorithm uses memetic evolution in the form of influencing of ideas from one individual to another in a local search. In SFL, the population consists of a set of frogs (solutions) partitioned into subsets, referred to as memeplexes. Actually the different memeplexes are considered as different cultures of frogs, each performing a local search [18].

Hence, the positions of the frogs are presented as:

$$D_i = rand \times (X_b - X_g) \quad (20)$$

And for new position:

$$X_{i+1} = X_i + D_i, -D_{max} \leq D_i \leq D_{max} \quad (21)$$

Where,

rand = random number between 0 and 1.

D_{max} = the maximum allowed change in a frog's position.

A. Modified Shuffled Frog Leaping Algorithm

The original SFLA algorithm has good performance when dealing with some simple problems. However, it is difficult for SFLA algorithm to overcome local minima when handling some complicate functions [19]. MSFLA starts with an initial population of "X" frogs created randomly like other evolutionary algorithms. The whole population of frogs is then partitioned into subsets referred to as memeplexes. The various memeplexes are considered as different cultures of frogs. And they are located at different places in the solution space (i.e., global search). Each culture of frogs performs a deep local search. Within each memeplex, the individual frogs hold information that can be influenced by the information of their frogs within their memeplex, and evolve through a process of change of information among frogs from different memeplexes [19]. After a defined number of evolutionary steps, information is passed among memeplexes in a shuffling process. The local search and the proposed processes continue until a defined convergence criterion is satisfied.

It is necessary to note that the mutation vector dimension is equal to the memeplexes number. Therefore, a position changing formula turns to the following form.

$$D_i = rand \times C \times (f(X_b) - f(X_w)) \times (X_b - X_w) \quad (22)$$

And for new position:

$$X_{i+1} = X_i + D_i \quad (23)$$

Where,

$C \in [0, C_{max}]$,

C_{max} = case dependant upper limit

$f(X_b)$= The best fitness functions that are found by the frogs in each memeplexs.

$f(X_w)$= The worst fitness functions that are found by the frogs in each memeplexs.

- Similar to the original *SFL*, if the process produces a better solution, the worst frog is replaced by the better one.

IV. MULTI-SCENARIO ANALYSIS

The problem formulation before seen allows determining the optimal network upgrade for an assigned profile of power produced by DG and of power absorbed by the loads. Nevertheless, in the planning problem the uncertainty of data related to the growth of power injected and absorbed requires an analysis that takes into account several scenarios. One solution is given by the usage of a probabilistic multi-scenario approach that furnish to the planners data on the which is possible to base the decision and to see the *"worst case"* taking into account the expectation of power demand and of

DG penetration.

In this way the analysis of uncertainties is combined a priori and their expectation can be used in a subsequent evaluation [20]. Each different scenario of DG penetration and load demand growth is characterized by a probability $\wp^{DG}{}_\sigma$ and $\wp^{LOAD}{}_\sigma$ respectively. All the scenarios are combined and the related cost function Φ_σ are optimized by the algorithm before drawn [21].

The probability $\wp_\sigma = \wp_\sigma{}^{DG} \cdot \wp_\sigma{}^{LOAD}$ is, then, associated to the selected optimal network $U^{OPT}{}_{NEW\sigma}$.

To each new line feasible l_i, connecting the bus h to the bus k of B_{NEW}, and for each scenario s, we associate the probability:

$$\wp_\sigma(l_i) = \wp_\sigma \cdot s_{h,k}^{L^{OPT}_{BNEW}\ \sigma} \tag{24}$$

After to have estimated all the scenarios, for each new feasible line l_i we define the figure of merit $\Pi(l_i)$ as sum of the probability $\wp_\sigma(l_i)$ [22]:

$$\Pi(l_i) = \sum_\sigma \wp_\sigma(l_i) \tag{25}$$

V.   NUMERICAL RESULTS

The proposed algorithm has been tested on the 30 lines, 28 buses test network depicted in figure 2.

Some different scenarios have been analyzed: in particular, we consider 4 scenarios of DG penetration and 3 scenarios of load demand growth. Each scenario, characterized by a probability $\wp_\sigma$, has been obtained as variation of a reference scenario. In particular, table I shows the DG scenarios hypothesized: starting from the reference scenario σ_1^{DG}, characterized by 3 new DG connected to the buses 5, 10 and 13 and that occurs with a probability $\wp_1^{DG}=0.5$, we consider the scenario σ_2^{DG} with probability $\wp_2^{DG}=0.25$ coming from the reference scenario without DG in node number 13, the scenario σ_3^{DG} with probability $\wp_3^{DG}=0.15$ coming from the reference scenario without DG in node number 5, and the scenario σ_4^{DG} whit probability $\wp_4^{DG}=0.1$ coming from scenario σ_1^{DG} with cut in power produced by the DG of 20%.

TABLE I – DG SCENARIOS ASSUMED

	Scenario			
	σ_1^{DG}	σ_2^{DG}	σ_3^{DG}	σ_4^{DG}
Probability $\wp_\sigma^{DG}$	50%	25%	15%	10%

The three load scenarios depicted in table II are: the reference scenario σ_1^{LOAD} characterized by a probability $\wp_1^{LOAD}=0.55$, the scenario σ_2^{LOAD} with probability $\wp_2^{LOAD}=0.30$ and with a power load that rises of 10% compared with the reference scenario σ_1^{LOAD}, and the scenario σ_3^{LOAD} with probability $\wp_2^{LOAD}=0.15$ and with a power load that fall of 10% compared with those of the reference scenario σ_1^{LOAD}.

TABLE II – LOAD SCENARIOS ASSUMED

	Scenario		
	σ_1^{LOAD}	σ_2^{LOAD}	σ_3^{LOAD}
Probability $\wp_\sigma^{LOAD}$	55%	30%	15%

In this way we define twelve network scenarios shown in table III.

In order to draw attention to the procedure pointed out, the solution of the scenario $\sigma_{1,1}$ is shown.
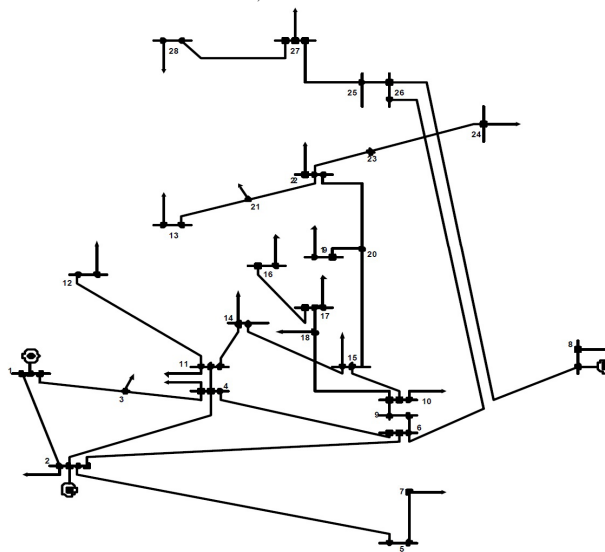
Figure 1.   Test network: initial configuration

TABLE III – NETWORK SCENARIOS

Scenario			
Network	DG	LOAD	Probability
$\sigma_{1,1}$	σ_1^{DG}	σ_1^{LOAD}	28%
$\sigma_{1,2}$	σ_1^{DG}	σ_2^{LOAD}	15.5%
$\sigma_{1,3}$	σ_1^{DG}	σ_3^{LOAD}	6.5%
$\sigma_{2,1}$	σ_2^{DG}	σ_1^{LOAD}	13.2%
$\sigma_{2,2}$	σ_2^{DG}	σ_2^{LOAD}	8%
$\sigma_{2,3}$	σ_2^{DG}	σ_3^{LOAD}	3.8%
$\sigma_{3,1}$	σ_3^{DG}	σ_1^{LOAD}	8.2%
$\sigma_{3,2}$	σ_3^{DG}	σ_2^{LOAD}	4.6%
$\sigma_{3,3}$	σ_3^{DG}	σ_3^{LOAD}	2.2%
$\sigma_{4,1}$	σ_4^{DG}	σ_1^{LOAD}	5.5%
$\sigma_{4,2}$	σ_4^{DG}	σ_2^{LOAD}	2.2%
$\sigma_{4,3}$	σ_4^{DG}	σ_3^{LOAD}	2.3%

The scenario is characterized by the probability:
$\wp_{1,1}= \wp_1^{DG} \cdot \wp_1^{LOAD}=0.5 \times 0.55=0.275$

In figure 3 the result of the MSFLA procedure in the $\sigma_{1,1}$ scenario is shown, highlighting the new DG power plant.

In figure 4 are shown the nodal voltages of the scenario $\sigma_{1,1}$ in presence of DG before and after grid upgrade pointed out by MSFLA.
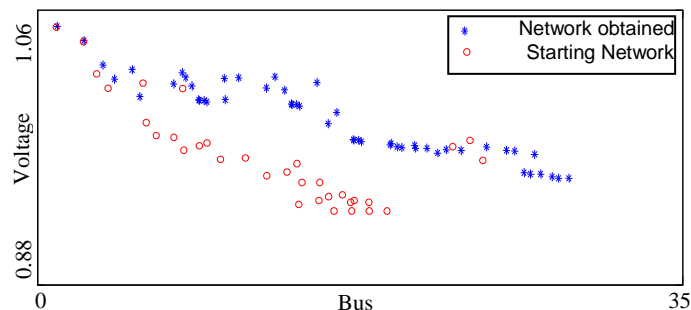
Figure 2.   Nodal Voltage in presence of DG before and after optimization in scenario $\sigma 1,1$.
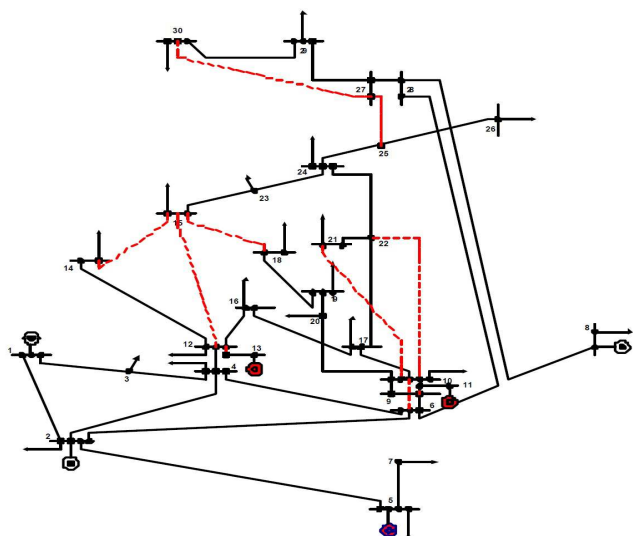
Figure 3.   Network obtained considering in scenario σ1,1.

In the case depicted by the scenario $\sigma_{1,1}$, the most probable, amongst all the feasible and equi-cost lines to build, the proposed procedure has proposed to build the lines connecting the buses 12 and 14, 12 and 15, 15 and 18, 6 and 10, 10 and 22, 25 and 27, 27 and 30.

The proposed procedure has been run on a Pentium IV 2.8 GHz computer, 512 MB ram 800 MHz FSB; all the twelve scenarios have been considered. The simulation results are shown in table IV.

In particular, the procedure proposes 4 different network configurations for the twelve scenarios hypothesized.

The results of the simulations show that the optimal network upgrades mainly depend on DG penetration.

In table V are shown the new lines selected by the procedure with the related probability.

The last row in the table V shows, instead, the figure of merit $\Pi(li)$ associated to the all feasible network lines. It is possible to note that the procedure does no select the feasible line $l_{6,8}$.

TABLE IV – SIMULATION RESULTS

Network Scenario	Number of new lines	Number of total lines	Computer time (s)
$\sigma_{1,1}$	9	39	55
$\sigma_{1,2}$	9	39	44
$\sigma_{1,3}$	9	39	46
$\sigma_{2,1}$	8	38	54
$\sigma_{2,2}$	8	38	51
$\sigma_{2,3}$	8	38	42
$\sigma_{3,1}$	10	40	62
$\sigma_{3,2}$	10	40	46
$\sigma_{3,3}$	10	40	46
$\sigma_{4,1}$	9	39	49
$\sigma_{4,2}$	9	39	55
$\sigma_{4,3}$	9	39	40

TABLE V – SIMULATION RESULTS

	Probability $\wp_{\sigma}(l_i)$																					
	from	to	from	to	from	to	from	to	from	to	from	to	from	to	from	to	from	to	from	to	from	to
	6	7	6	8	6	10	9	11	12	13	12	15	14	15	15	18	10	21	25	27	27	30
$\sigma_{1,1}$	0		0		27%		27%		27%		27%		27%		27%		27%		27%		27%	
$\sigma_{1,2}$	16.5%		0		16.5%		16.5%		16.5%		16.5%		0		16.5%		16.5%		16.5%		16.5%	
$\sigma_{1,3}$	6.6%		0		6.6%		6.6%		6.6%		6.6%		0		6.6%		6.6%		6.6%		6.6%	
$\sigma_{2,1}$	13.2%		0		13.2%		13.2%		0		13.2%		0		13.2%		13.2%		13.2%		13.2%	
$\sigma_{2,2}$	8%		0		8%		8%		0		8%		0		8%		8%		8%		8%	
$\sigma_{2,3}$	3.8%		0		3.8%		3.8%		0		3.8%		0		3.8%		3.8%		3.8%		3.8%	
$\sigma_{3,1}$	8.2%		0		8.2%		8.2%		8.2%		8.2%		8.2%		8.2%		8.2%		8.2%		8.2%	
$\sigma_{3,2}$	4.6%		0		4.6%		4.6%		4.6%		4.6%		4.6%		4.6%		4.6%		4.6%		4.6%	
$\sigma_{3,3}$	2.1%		0		2.1%		2.1%		2.1%		2.1%		2.1%		2.1%		2.1%		2.1%		2.1%	
$\sigma_{4,1}$	5.2%		0		5.2%		5.2%		5.2%		5.2%		0		5.2%		5.2%		5.2%		5.2%	
$\sigma_{4,2}$	0		0		2.2%		2.2%		2.2%		2.2%		2.2%		2.2%		2.2%		2.2%		2.2%	
$\sigma_{4,3}$	2.6%		0		2.6%		2.6%		2.6%		2.6%		0		2.6%		2.6%		2.6%		2.6%	
$\Pi(li)$	70.8%		0		100%		100%		75%		100		43.2%		100%		100%		100%		100%	

VI.   Conclusion

To support distribution network operator in planning phase in presence of distributed generation the Modified Shuffled Frog Leaping (MSFL) algorithm is proposed in this paper. Among the feasible network upgrades, the method selects the new lines to build that minimize the mean square error of the buses voltages. The proposed MSFL is applied to the 30 lines, 28 bus power system with multi-scenario analysis in order to evaluate different levels of DG penetration and load demand. The simulation results demonstrate the superiority of MSFL to improve significantly the bus voltages driving opportunely the planner in the distribution network upgrading phase. Further develops of the work are in progress in order to consider no equi-cost upgrades and formalizing a multi-objective optimization problem.

References

[1]   T. Ackermann, G. Andersson, L. Söder, Distribited generation: a definition, Electric Power System Research - Elsevier, n 57, 2001, pp-195-204.

[2]   H. Lee Willis, W.G. Scott, Distributed power generation: planning and evaluation, Marcel Dekker press.

[3]   M. Esmalifalak, Z. Han, and L. Song "Effect of Stealthy Bad Data Injection on Network Congestion In Market Based Power System" IEEE Wireless Communications and Networking Conference (WCNC 2012), Paris, France, Apr. 2012.

[4] M. Esmalifalak, G. Shi, Z. Han, and L. Song "Attack Against Electricity Market–Attacker and Defender Gaming" IEEE Global Communications Conference Exhibition Industry Forum (Globecom 2012), Anaheim, USA, Dec. 2012.

[5] A.Yaghooti, G. Ahmad khanbeigi, and M. Esmali Falak, "Generation Expansion Planning in IEEE Power System Using Probabilistic Production Simulation" IEEE Energy Conference & Exhibition, Dec. 18-22, 2010, Manama, Bahrain.

[6] T. Hoff and D. S. Shugar, "The value of gridsupport photovoltaics in reducing distribution system losses," IEEE Transactions Energy Conversion, vol. 10, pp. 569–576, Sept. 1995.

[7] G. Joos, B.T. Ooi, D. McGillis, F.D. Galiana and R. Marceau, "The Potential of Distributed Generation to Provide Ancillary Services", IEEE Power Engineering Society Summer Meeting, pp 1762- 1767, 2000.

[8] P. Chiradeja, R. Ramakumar, "Voltage Profile Improvement with Distributed Wind Turbine Generation - A Case Study", IEEE Power Engineering Society General Meeting, Vol. 4, pp 2331-2336, 2003.

[9] Y. G. Hegazy, M. M. A. Salama, and A. Y. Chikhan" Adequacy Assessment of Distributed Generation Systems Using Monte Carlo Simulation", IEEE Transactions on Power Systems, Vol. 18, No. 1, pp.48-52, February 2003.

[10] Koen J. P. Macken, Math H. J. Bollen, and Ronnie J. M. Belmans," Mitigation of Voltage Dips Through Distributed Generation Systems", IEEE Transactions on Industry Applications, Vol. 40, No. 6, pp. 1686- 1693, November/December 2004.

[11] Aleksandar Pregelj, Miroslav Begovic and Ajeet Rohatgi, "Quantitative Techniques for Analysis of Large Data Sets in Renewable Distributed Generation", IEEE Transactions on Power Systems, Vol. 19, No. 3, 1277- 1285, August 2004.

[12] Caisheng Wang, and M. Hashem NehrirV, "Analytical Approaches for Optimal Placement of Distributed Generation Sources in Power Systems", IEEE Transactions on Power Systems, Vol. 19, No. 4, pp. 2068 – 2076, November 2004.

[13] G. Celli, F. Pilo, Optimal distributed generation allocation in MV distribution networks, Proc. of Innovative Computing for Power - Electric Energy Meets the Market International Conference, May 2001, pages 81-86.

[14] E. Lakervi, E.J. Holmes, Electricity distribution network design, IEE Power Engineering Series 2, 2003. Bulk Power System Dynamics and Control - VI, August 22-27, 2004, Cortina d'Ampezzo, Italy 379

[15] I. Mac Gill, R.J. Kaye, Decentralized coordination of power system operation using dual evolutionary programming, IEEE Transaction on Power System, vol. 14, n. 1, February 1999, pages 112-119.

[16] L. Noakes and R. Kozera, "A 2D Leap-Frog algorithm for optimal surface reconstruction", Proc. 44th Annual Meet. Opt. Eng. SPIE'99, III-3811, pp: 317–328, 1999.

[17] L. Noakes and R. Kozera. "Denoising images: Non-linear leap-frog for shape and lightsource recovery", Chapter in Theoretical Foundations of Computer Vision Geometry, Morphology and Computational Images, pp: 419–436, Lecture notes in Computer Science 2616, 2003.

[18] L. Noakes and R. Kozera, "Nonlinearities and noise reduction in 3- ource photometric stereo", J. Math. Imag. and Vis., vol. 2, no. 18, pp: 119-127, 2003.

[19] M. Eusuff, and E. Lansey, "Optimization of water distribution network design using the shuffled frog leaping algorithm", Journal of Water Resources Planning and Management, ASCE, vol. 129, no. 3, pp: 210-225, May 2003.

[20] J. Kennedy, and R. Eberhart, "Particle swarm optimization", Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, pp: 1942-1948, 1995.

[21] H. Kirkham, D. Nightingale, T. Koerner, Energy management system design with dispersed storage and generation, IEEE Transaction on Power Apparatus and Systems, n. 7, 1981, pages 3432-3441.

[22] V. Calderaro, V. Galdi, S. Raiti, A. Vaccaro, ICT in distribution networks with wind power generation to improve voltage profiles, Proc. of 39th International Universities Power Engineering Conference UPEC 2004, Bristol, September, 2004.

[23] M.I. Marei, E.F. El-Saadany, M.M.A. Salama, Flexible distributed generation: (FDG), IEEE Power Engineering Society Summer Meeting, 2002, vol. 1, July 2002, pages 49 – 53.

BIOGRAPHIES

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 410 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in

1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

OPTIMZATION OF HEAT TREATMENT PROCESS PARAMETERS USING NEURAL NETWORKS AND NELDER-MEAD ALGORITHM

Tarık ÇAKAR[1], Fatih KESKİNKILIÇ[2]

[1]*Department of Industrial Engineering,Sakarya University,Esentepe Campus,Serdivan,Turkey*
2Mucur Technical Vocational Schools of Higher Education, Ahi Evran University,Mucur, Kırşehir,Turkey

Abstract: Metallurgical processes consist of different and complex production operations. One of them is heat treatment. Hardness value is an important response variable for heat treatment process. Heat treatment parameters and interactions between each other are not known clearly. Hence it is hard to define convenient parameters for requested hardness value. In this study, effects of heat treatment parameters on hardness are modelled using back propagation artificial neural network (BPANN) model. BPANN is used to formulate a fitness function for predicting the value of the response based on the parameter settings and then Nelder-Mead algorithm takes the fitness function from the trained network to search for the optimal heat treatment parameters (furnace heat and heat treatment time) combination.

Keywords: Heat Treatment, Neural Network, Hardness, Modelling, Nelder-Mead Algorithm

1. INTRODUCTION

High quality and cheap parts are always preferred by costumer. This situation is result of the increasing competition. There are some uncontrollable parameters in heat treatment process because of using different raw material in casting process. Trial-and-error technic has been used for investigate to optimization of materials processing for a long time (Song and Zhang, 2001). Thus increases cost and makes products less qualified. According to Lahoucine-Abaih et al. the most important control factors for the mechanical properties are the chemical composition and the tempering treatment after the quenching (Lahoucine-Abaih et al. 2007)

In this study, Back Propagation Artificial Neural Network (BPANN) model is used for determining the effects of heat treatment parameters on hardness. Modelling and also prediction specialty of BPANN is used for increasing casting parts quality and decreasing manufacturing costs. BPANN trained with heat treatment parameters that determined as input and hardness value that determined output. BPANN is used to formulate a fitness function for predicting the value of the response based on the parameter settings and then Nelder-Mead algorithm takes the fitness function from the trained network to search for the optimal heat treatment parameters (furnace heat and heat treatment time) combination. This study is applied in a casting factory that produces casting grinding media for the cement and mining industries.

The flexibility and simplicity of neural networks have made them a popular modelling and forecasting tool across different research areas in recent years. A variety of neural network models have been developed, among which the back-propagation (BP) network is the most widely adopted in the present study. According to Song and Zhang an artificial neural network can be applied very well to model the effects of the heat treatment technique on mechanical properties (Song and Zhang, 2001).

2. BUILDING THE NEURAL NETWORK MODEL AND OPTIMIZATION

The use of artificial neural networks has become popular. Material properties such as hardness, tensile strength, fatigue, and yield strength are a complex function of many parameters such as alloying elements and heat treatment conditions and developing theoretical models that can quantitatively predict these

parameters is not a straightforward task (Sidhu et al. 2012). The artificial neural networks are one of the most powerful modelling techniques with very quick return for the practice (Malinov and Sha, 2004).

In this study the reasons of using neural network are;
- Complex problems can be easily modelled
- Does not need any prior knowledge. Appropriate samples are enough for the model
- Neural networks can be applied to problems that do not have algorithmic solutions or algorithmic solutions that are too complex to be found (Khalaj et al. 2011)
- Applications of artificial neural networks are the cheapest in terms of costs and they are the most efficient in terms of time (Keskinkilic, 2010)

Complicated and non-linear complex relationships between outputs and inputs can be modelled and provided sufficient reliable data is available for training by BPANN.

2.1. Dataset and Input/output Parameters

The neural relationships would be very useful to industries for designing their experiments, an eventually their alloy (Reddy et al. 2002). To develop a neural network with good performance, there needs to be an adequate quantity of experimental data available (Sha and Edwards, 2007). In this study 130 parameters used for BPANN model. In order to modelling heat treatment process 110 operation parameters have taken for learning 20 parameters picked randomly for testing. Experts opinion literal review were choosing heat treatment process parameters (Chang et al, 2007), (Khalaj et al. 2011). All data sets were normalized to a -1:1 range for computation.

Table 1. Analyse of training and test sets

	C%	Si%	Mn%	P%	S%	Cr%	T	C°	HRC
Min	2,20	0,50	0,70	0,01	0,01	16,00	338,00	956,00	55,00
Max	2,60	1,00	1,00	0,05	0,05	17,99	358,00	990,00	65,00
Mean	2,42	0,68	0,88	0,03	0,03	16,75	348,33	973,57	59,14
Std. D.	0,12	0,15	0,09	0,01	0,01	0,65	5,24	11,16	3,47

Chemical properties of balls percentages of Carbon (C %), percentages of Silisium (Si %), percentages of Manganese (Mn %), percentages of Phosphors (P %), percentages of Sulphur (S %), percentages of Chrome (Cr %) are determined as input parameter for BPANN. Heat treatment process time (T) and furnace heat (C°) are determined as input parameter for BPANN as well. Measured hardness value in Rockwell (HRC), after heat treatment determined as output for the model(Keskinkilic and Cakar, 2012).
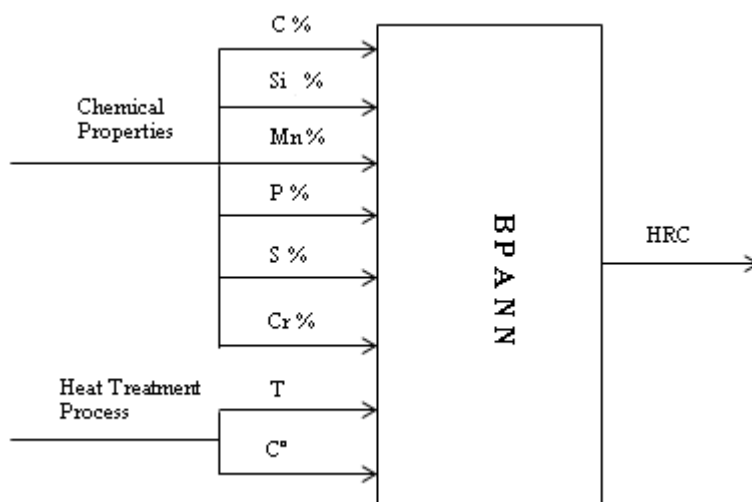
Figure 1. Schematic model of Back Propagation Artificial Neural Network

2.2. Neural Network Training

Before the determination of this architecture of BPANN model many different of experiments has performed. Figure 2 shows the determined back propagation neural network model architecture. BPANN has 8 neurons on input layer and two hidden layers one of them has 6 neurons and the other has 4 neurons. There is one neuron on output layer. The architecture was chosen as a result of these experiments gives the closest results.
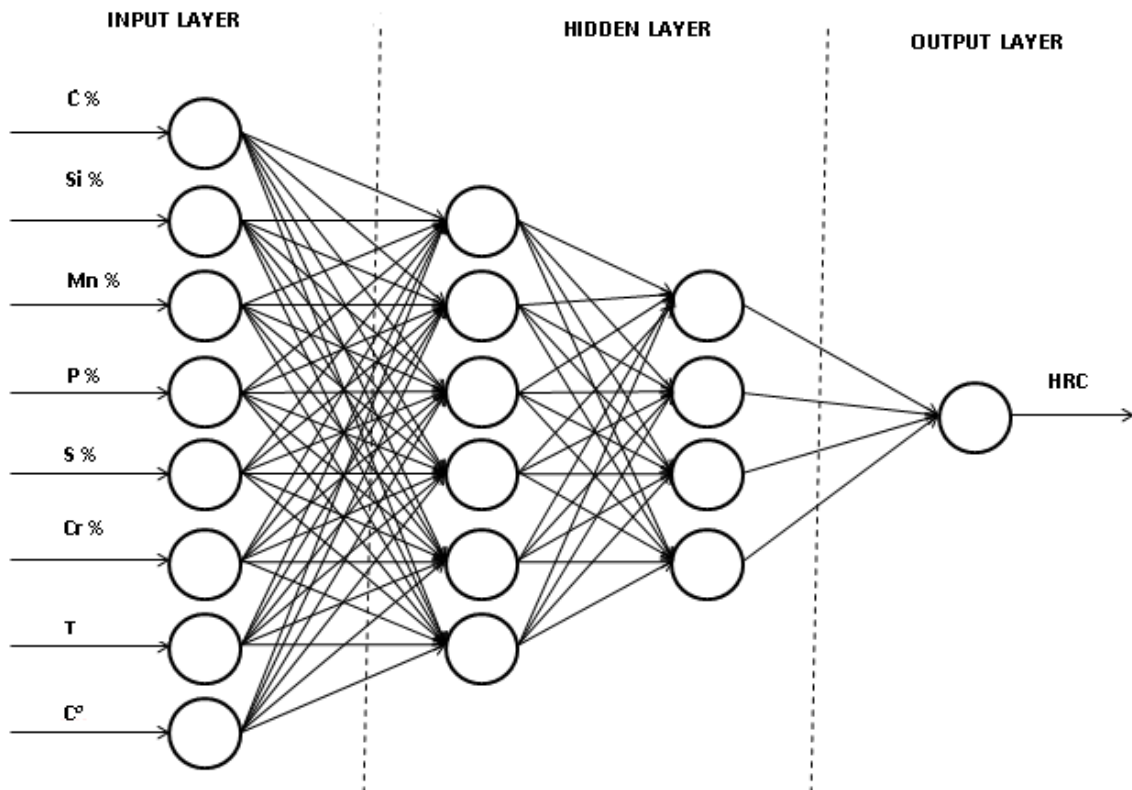
Figure 2. The architecture of Back Propagation Artificial Neural Network.

Artificial neural network model generated with Matlab R2010a. Neural network has reached desired performance value (Mean Square Error) in 525th iteration as shown figure 3. Artificial neural network training data is given below. Momentum rate (γ) and learning rate (α) obtained by simulation.

MSE	10^{-4}		
α	0,05		
γ	0,8		
Transfer Functions	First Layer	Second Layer	Third Layer
	Hyperbolic Tangent Sigmoid	Hyperbolic Tangent Sigmoid	Hyperbolic Tangent Sigmoid
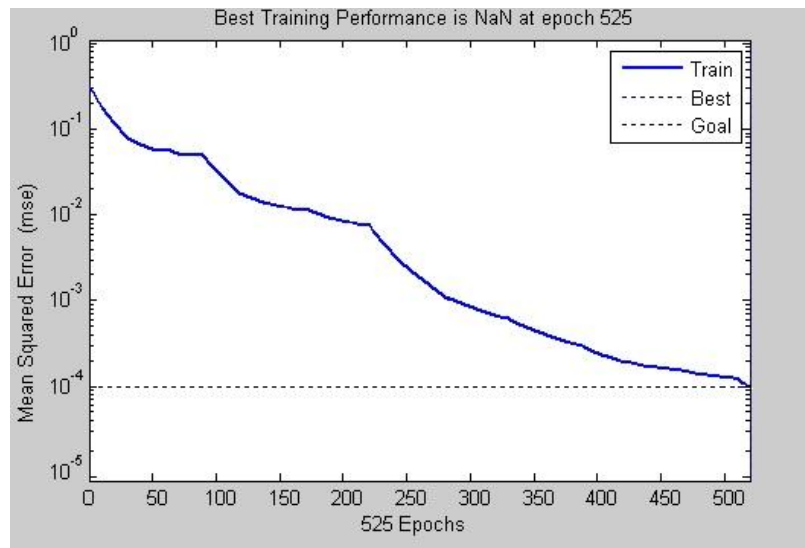
Figure 3. Learning Curve

2.3. Nelder –Mead Algorithm

In this study Nelder–Mead Algorithm is used for optimization. This method first was introduced by Spendley et. al. in 1962 and later in year 1965 Nelder and Mead expanded their method (Malek and Shekari Beidokhti ,2006). Nelder–Mead Algorithm is a direct search method that does not use numerical or analytic gradients. The MATLAB r2010 program built-in routine ''fminsearch()'' uses the Nelder–Mead algorithm to minimize a multivariable objective function. .

Nelder-Mead algorithm takes the fitness function from the trained network to search for the optimal heat treatment parameters (furnace heat and heat treatment time) combination.

According to factory experiments and customer's request 90 mm. diameters grinding ball hardness value determined as 64 HRC and Nelder-Mead Algorithm used for searching optimum parameters. Table 2 shows the parameters were used for Nelder –Mead Algorithm.

	C%	Si%	Mn%	P%	S%	Cr%	T	C°	HRC (required)
	0.8002	-0.3243	0.4876	0.4543	-0.9123	0.3042	?	?	0.8
formulation	x1	x2	x3	x4	x5	x6	x7	x8	y_ysa

C%, Si%, Mn%, P%, S%, Cr% values comes from casting. For this parameter furnace heat (C°) and heat treatment time (T) optimized by Nelder-Mead Algorithm and figure 4 shows the result.
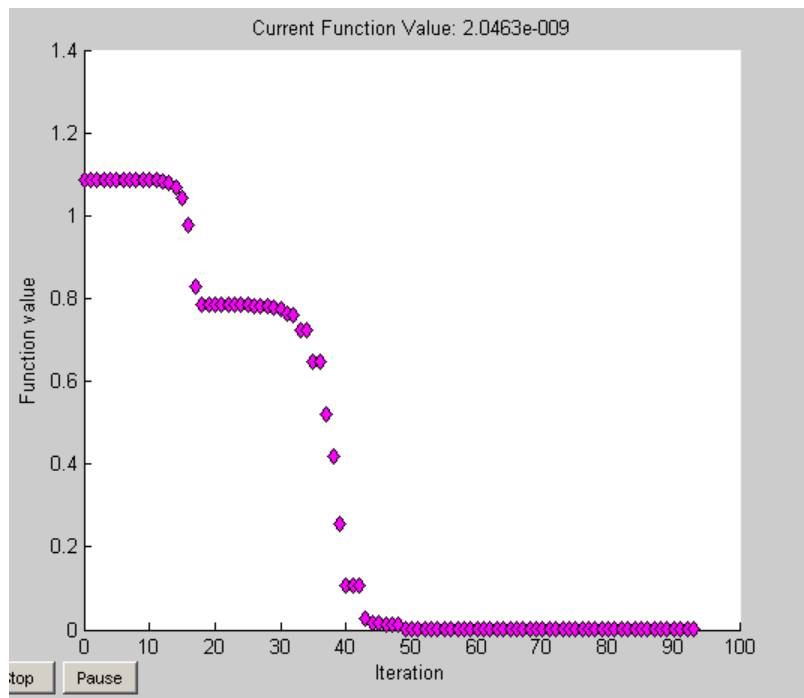
Figure 4. Optimization Function Value Plot

Fitness function of trained BPANN model is designed as M File For Matlab R2010 is below.

```
function fitness1= neldermead1(v)

x1= 0.8002;

x2= -0.3243;

x3= 0.4876;

x4= 0.4543;

x5= -0.9123;

x6= 0.3042;

y_ysa = 0.8;

x7 = v(1)< 1 > -1 ;

x8 = v(2)< 1 > -1 ;

 a1=-0.58485*x1+0.035466*x2+0.20587*x3-0.89896*x4+0.59904*x5+0.41987*x6+1.088*x7
0.21724*x8+1.878;

 B1 = 2 / (1 + 2.71828182846 ^ (-2 * a1)) - 1;

a2=0.31238*x1+0.83279*x2-0.73064*x3+0.30797*x4-0.59127*x5+0.26436*x6-0.74227*x7-0.63316*x8-
1.1742;

 B2 = 2 / (1 + 2.71828182846 ^ (-2 * a2)) - 1;

a3=-0.070326*x1-0.49116*x2-1.0824*x3-0.12381*x4+0.12421*x5+0.63875*x6-
0.68887*x7+0.84183*x8+0.53773;

B3 = 2 / (1 + 2.71828182846 ^ (-2 * a3)) - 1;
```

a4=-0.26509*x1+0.79434*x2-0.090284*x3-0.86625*x4-0.60704*x5-0.030157*x6-0.041022*x7-1.2133*x8-0.21271;

B4 = 2 / (1 + 2.71828182846 ^ (-2 * a4)) - 1;

a5=0.66158*x1+0.43736*x2+0.59978*x3-0.33718*x4+0.14022*x5-0.8339*x6-0.96417*x7+0.59696*x8+1.0625;

 B5 = 2 / (1 + 2.71828182846 ^ (-2 * a5)) - 1;

a6=0.010804*x1-0.33588*x2+0.93053*x3-0.64992*x4-0.60707*x5-0.75099*x6-0.89232*x7+0.73777*x8+1.6353;

B6 = 2 / (1 + 2.71828182846 ^ (-2 * a6)) - 1;

c1 = 0.93723*B1+0.7665*B2-0.61156*B3+0.21256*B4+0.33913*B5-1.0644*B6-1.7561;

D1 = 2 / (1 + 2.71828182846 ^ (-2 * c1)) - 1;

c2 = -1.3156*B1-0.021737*B2-0.94637*B3+0.1057*B4-0.49716*B5-0.84176*B6+0.4679;

D2 = 2 / (1 + 2.71828182846 ^ (-2 * c2)) - 1;

c3 = -0.17678*B1+0.85063*B2-0.76109*B3-0.8328*B4+0.066926*B5-0.90953*B6-0.68646;

D3 = 2 / (1 + 2.71828182846 ^ (-2 * c3)) - 1;

c4 = -0.38197*B1-0.72463*B2+1.0987*B3+0.9584*B4-0.1934*B5-1.0225*B6-1.5855;

D4 = 2 / (1 + 2.71828182846 ^ (-2 * c4)) - 1;

e1 = (-0.26368*D1+0.49131*D2-0.38967*D3+1.1713*D4+0.69527);

 Y1 = 2 / (1 + 2.71828182846 ^ (-2 * e1)) - 1;

 fitness1 = abs(Y1 - (y_ysa));

3. CONCLUSIONS

Back propagation artificial neural network model has shown good agreement with tested heat treatment model. Nelder-Mead Algorithm is used for optimization the parameters. The algorithm proposed here finds the approximate solution in a closed analytical form. Results are very useful for optimizing heat treatment parameters to fulfill a highly efficient output state.

In this study BPANN use for just 90 mm diameters grinding ball heat treatment process. Results can be extended for other heat treatment parts.

REFERENCES

Chang, L., Liu, L., Liu, J.H., 2007. Properties of Cross-Rolled Low Alloy White Cast Iron Grinding Ball. *Journal Of Iron And Steel Research, International. 2007, 14(5): 47-51*

Keskinkilic, F., 2010. Analysis of parameters which effect casting product hardness with artificial neural networks and a case *Study, M.Sc. Thesis. Kırıkkale University Graduate School of Natural and Applied Sciences Department of Industrial Engineering*

Keskinkilic, F., Cakar, T. 2012. The Modellıng Of Heat Treatment Process Parameters Usıng Neural Networks. Proceedings of 8th International Symposium on Intelligent and Manufacturing Systems. 27-28, 2012: 714-720

Khalaj, G., Yoozbashizadeh, H., Khodabandeh, A., Nazari, A., 2001. Artificial neural network to predict the effect of heat treatments on Vickers microhardness of low-carbon Nb microalloyed steels. *Journal of Neural Comput & Applic*. DOI 10.1007/s00521-011-0779-z

Lahoucine-Abaih, L., Van Bennekom, A., Fathi, M.,2007. Development of an ANN for the Prediction of Heat Treatment Temperatures for Martensitic Stainless Steels.*IEEE International Symposium on Industrial Electronics.* 1-4244-0755 IEEE

Malek, A., Shekari Beidokhti, R.,2006. Numerical solution for high order differential equations using a hybrid neural network—Optimization method. *Applied Mathematics and Computation* 183 (2006) 260–271

Malinov, S., Sha, W., 2004.   Application of artificial neural networks for modelling correlations in titanium alloys, *Materials Science and Engineering* A365 (2004) 202–211

Reddy, N. S., Krishnaiah, J., Kiran Kuma, Y., Acharya, N.N., 2002. Modeling of low nickel-chromium steels by using back propagation neural networks. *Proceedings of the 9th International Conference on Neural Information Processing* (ICONIP'O2) , Vol.. 2.

Sha, W., Edwards, K.L., 2007. The use of artificial neural networks in materials science based research, *Materials and Design 28 (2007) 1747–1752*

Sidhu, G., Bhole, S. D., Chen D.L., Essadiqi, E., 2012., Development and experimental validation of a neural network model for prediction and analysis of the strength of bainitic steels. Materials and Design 41 (2012) 99–107

Song, R.G., Zhang, Q.Z., 2001.Heat treatment technique optimization for 7175 aluminum alloy by an artificial neural network and genetic algorithm. *Journal of Material Processing Technology* 117 (2001) 84-88

Adaptive Sliding Mode & Nussbaum Gain Hybrid Control of Supersonic Missiles with Unknown Control Direction

Junwei Lei, Jinyong Yu, Lingling Wang, Hewei Zhao, Jianhong Shi and Xiaojun Guo

Department of Control Engineering, Naval Aeronautical and Astronautical University, Yantai, Shandong, China

Abstract - *A new nonlinear sliding mode, which can be expended and used in a large family nonlinear systems , is skillfully constructed for supersonic missile control with unknown control direction. Later, by using of the Nussbaum gain method, the unknown control direction problem is solved and also the appearance of control singularity was avoided. So the nonlinear sliding mode method and Nussbaum gain method are integrated perfectly in this paper and neither one is dispensable. Finally, this method is applied in supersonic missile system and the simulation result shows the rightness of the proposed method.*

Keywords: Unknown control direction; Nussbaum Gain; Sliding Mode; Supersonic Missile; Adaptive; Control

1 Introduction

The control system of missiles described by the differential equations have strong nonlinearities and time varying characteristics. A traditional way to design the autopilots is to linearize these equations under different flight conditions, then to apply linear design techniques to the linearized models and finally to schedule the control parameters according to different flight conditions and states.

However, this so-called gain scheduling method is a tedious procedure. All the models described under different flight conditions should be designed one by one. Furthermore, it does not give an optimal solution nor guarantee stability robustness of the whole trajectory. Also, the assumption of the uncertainties and model errors are not considered sufficiently.

Adaptive backstepping controllers were designed for nonlinear models of missile motion in many papers. The nonlinearities in the class of systems that we considered is due to both parametric uncertainty and unknown nonlinear functions. The uncertainties in missile pitch plane's model considered in[10] are consisted of uncertain parameters and unknown nonlinear functions, where the unknown functions represents the model error or the time varying of the system. The main assumption is that these unknown functions satisfy socalled triangular bounds conditions. In particular, the

unknown nonlinaerities satisfy some growth conditions characterized by bounding functions composed of known functions multiplied by unknown parameters. Polycarpou & Ioannou [10]designed an adaptive backstepping controller which guaranteed the uncertain system's uniform ultimate boundedness. Based on that, Seung-Hwan Kim & Yon-Sik Kim [12]further estimated a virtural control coefficient.

All the above research are based on the assumption that the control direction is known. To make it sample, some paper even assume that the sign of control coefficient, which also appeared in [12], is negative or positive.

But the control direction is unknown under some complex flight condition of supersonic or high-supersonic missiles. Or the sign of control coefficients will be changed unexpectedly in some special flight area. So the traditional control law based on the known control direction will face the unstable threaten.

The Nussbaum gain method can deal with the unknown control direction problem, which is discussed in many papers recently[3-21].Adaptive and backstepping technologies are used in many papers to construct controllers for supersonic missiles. But there are two disadvantages by using the backstepping technology. First, the control law will be more and more complex as the order of the system increase. Second, the differential bomb problem will be caused. Although some relative methods are proposed to solve the differential bomb problem, they are still very tedious.

For missile control systems, the control direction is assumed to be known in many papers. In fact, as the increase of the velocity of missiles, it is necessary to design a controller that can cope with the unknown control direction situation because the sign of control coefficient will be changed unexpectedly in some special flight conditions. A robust sliding mode controller was designed for airplane flight control system in [13]. The linear model of F-14 was considered and error and high-order derivative of error were used to construct the sliding mode surface . Finally, a robust sliding mode control law was designed by the way commonly used in many papers. It is wroth pointing out that there are one advantage by adopting this method. It is very difficult to get the high-order derivative of error for many actual systems,

especially for flight control systems. Some times it is impossible to measure the derivative of error because the signal does not have a physical meaning. Sometimes it is possible to measure the signal but it is too expense to measure the signal accurately. So in this paper the error and the integral of error and else are considered to construct a new kind of nonlinear sliding mode surface. Also, compared with the backstepping method, it is able to make the design sampler by using the sliding mode method . Over all, considering the unknown control direction problem, a new kind of nonlinear sliding mode adaptive controller is designed for supersonic missile systems. Meanwhile, the Nussbaum gain method is integrated with sliding mode surface to solve the unknown control direction problem in this paper.

2   Problem description

The nonlinear model of supersonic missile's pitch plane flight motion mainly discusses in this paper can be written as follows:

$$\dot{x}_1 = x_2 + \theta_{11}\varsigma_{11}(x_1) + \theta_{12}\varsigma_{12}(x_1) + g_1 x_3 + \Delta_1(x_1, t) \quad (1)$$

$$\dot{x}_2 = b_2 x_3 + \theta_{21}\varsigma_{21}(x_1) + \theta_{22}\varsigma_{22}(x_1) + \Delta_2(x_1, t) \quad (2)$$

$$\dot{x}_3 = au - ax_3 \quad (3)$$

Where the definition of all parameters are the same as it in paper[10]. The control objective is a design u such that the state x_1 is able to track the desired x_1^d.

Lemma 1: For the above model, the state x_1 can be converged to the desired state x_1^d, if the states of the system can be converged to the following sliding mode, in other word, it satisfies the following equation:

$$S = (x_1 - x_1^d) + a_s^2 \int_0^t (x_1 - x_1^d)dt$$
$$+ (x_2^2 + x_3^2)sign\{\int_0^t (x_1 - x_1^d)dt\} = 0 \quad (4)$$

Proof: Define a new variable as

$$w_1 = \int_0^t (x_1 - x_1^d)dt \quad (5)$$

If the states of system satisfies

$$S = \dot{w}_1 + a_s^2 w_1 + (x_2^2 + x_3^2)sign\{w_1\} = 0 \quad (6)$$

Then we have

$$\dot{w}_1 = -a_s^2 w_1 - (x_2^2 + x_3^2)sign(w_1) \quad (7)$$

Choose the following Lyapunov function

$$V = \frac{1}{2}w_1^2 \quad (8)$$

We solve the derivative of the Lyapunov function along the trajectory of solution, it is easy to get

$$\dot{V} = -a_s^2 w_1^2 - (x_2^2 + x_3^2)|w_1| \le 0 \quad (9)$$

So we get $w_1 \to 0$ and $x_1 \to x_1^d$.

Considering the appearance of the squared items which will make the control law to be tedious, we can improve the Lemma 1 by using a nonlinear function as follows:

Lemma 2: For the above model, the state x_1 can be converged to the desired state x_1^d, if the states of the system can be converged to the following sliding mode, in other word, it satisfies the following equation:

$$S = (x_1 - x_1^d) + a_s^2 \int_0^t (x_1 - x_1^d)dt$$
$$+ (x_2 + x_3)sign\{(x_2 + x_3)\int_0^t (x_1 - x_1^d)dt\} = 0 \quad (10)$$

Considering the using of sign function in above equation, it is not convenient to get the derivative. So the sliding mode surface can be improved as follows.

Lemma 3: For the above model, the state x_1 can be converged to the desired state x_1^d, if the states of the system can be converged to the following sliding mode, in other word, it satisfies the following equation:

$$S = (x_1 - x_1^d) + a_s^2 \int_0^t (x_1 - x_1^d)dt$$
$$+ (x_2 + x_3)ac\tan Q = 0, \quad (11)$$
$$Q = (x_2 + x_3)\int_0^t (x_1 - x_1^d)dt$$

Considering that the equation (11) is still complex, so it can be improved as follows.

Lemma 4: For the above model, the state x_1 can be converged to the desired state x_1^d, if the states of the system can be converged to the following sliding mode, in other word, it satisfies the following equation:

$$S = (x_1 - x_1^d) + (a_s^2 + x_2^2 + x_3^2)\int_0^t (x_1 - x_1^d)dt \quad (12)$$

In some situation, the sliding mode surface can also be defined as Lemma 5.

Lemma 5: For the above model, the state x_1 can be converged to the desired state x_1^d, if the states of the system can be converged to the following sliding mode, in other word, it satisfies the following equation:

$$S = (a_s^2 + x_2^2 + x_3^2)(x_1 - x_1^d) + \int_0^t (x_1 - x_1^d)dt \quad (13)$$

3 Nussbaum gain & Sliding mode design

In this section, the lemma 5 is used to design the nonlinear sliding mode controller, and other kinds of nonlinear sliding mode controller can follow the same design procedure. First, define the sliding mode surface as

$$S = (1 + x_2^2 + x_3^2)(x_1 - x_1^d) + c_1 \int (x_1 - x_1^d) dt \quad (14)$$

Then we solve the derivative of sliding mode surface

$$\dot{S} = (1 + x_2^2 + x_3^2)\dot{x}_1 + (1 + 2x_2\dot{x}_2 + 2x_3\dot{x}_3) \\ (x_1 - x_1^d) + c_1(x_1 - x_1^d) \quad (15)$$

To make the following illustration brief, we define $e = x_1 - x_1^d$ then we have

$$f_1 = (1 + x_2^2 + x_3^2) \\ f_2 = 2x_2 = \partial f_1 / \partial x_2 \quad (16) \\ f_3 = 2x_3 = \partial f_1 / \partial x_3$$

Also the derivative of the sliding mode surface can be described as

$$\dot{S} = (1 + c_1)e + f_1\dot{x}_1 + f_2 e\dot{x}_2 + f_3 e\dot{x}_3 \quad (17)$$

Considering the system model (1), we have

$$\dot{S} = (1 + c_1)e + f_1 x_2 + f_1\theta_{11}\varsigma_{11}(x_1) + f_1\theta_{12}\varsigma_{12}(x_1) \\ + f_1 g_1 x_3 + f_1\Delta_1(x_1,t) \\ + f_2 e b_2 x_3 + f_2 e\theta_{21}\varsigma_{21}(x_1) + f_2 e\theta_{22}\varsigma_{22}(x_1) \\ + f_2 e\Delta_2(x_1,t) + auf_3 e - f_3 eax_3 \quad (18)$$

So the ideal control can be designed as

$$u^* = \frac{1}{aef_3}\{-(1 + c_1)e + f_3 eax_3 - f_1 x_2 - f(S) \\ - f_1\hat{\theta}_{11}\varsigma_{11}(x_1) - f_1\hat{\theta}_{12}\varsigma_{12}(x_1) - f_1\hat{g}_1 x_3 \quad (19) \\ - f_1\Delta_1(x_1,t) - f_2 e\hat{b}_2 x_3 - f_2 e\theta_{21}\varsigma_{21}(x_1) \\ - f_2 e\theta_{22}\varsigma_{22}(x_1) - f_2 e\Delta_2(x_1,t)\}$$

Because of the uncertainties of the model, it is impossible to get the ideal control, so we consider designing the adaptive control law as follows

$$u_{dd} = \frac{1}{aef_3}\{-(1 + c_1)e + f_3 eax_3 - f_1 x_2 \\ - f(S) - f_1\hat{\theta}_{11}\varsigma_{11}(x_1) - f_1\hat{\theta}_{12}\varsigma_{12}(x_1) \quad (20) \\ - f_1\hat{g}_1 x_3 - f_2 e\hat{b}_2 x_3 - f_2 e\hat{\theta}_{21}\varsigma_{21}(x_1) \\ - f_2 e\hat{\theta}_{22}\varsigma_{22}(x_1) - \eta_1(x_1,t) - \eta_2(x_1,t)\}$$

Where

$$f(S) = k_1 S \quad (21)$$

Define

$$u_{dN} = -(1 + c_1)e + f_3 eax_3 - f_1 x_2 - f(S) \\ - f_1\hat{\theta}_{11}\varsigma_{11}(x_1) - f_1\hat{\theta}_{12}\varsigma_{12}(x_1) - f_1\hat{g}_1 x_3 \\ - f_2 e\hat{b}_2 x_3 - f_2 e\hat{\theta}_{21}\varsigma_{21}(x_1) \quad (22) \\ - f_2 e\hat{\theta}_{22}\varsigma_{22}(x_1) - \eta_1(x_1,t) - \eta_2(x_1,t)$$

Design the Nussbaum gain adaptive control as

$$u = -N(k)u_{dN} \quad (23)$$

where $N(k)$ is a Nussbaum gain function, and design the adaptive regulation law as:

$$\dot{k} = -Su_{dN} \quad (24)$$

The derivative of the sliding mode surface can be rewritten as

$$\dot{S} = (1 + c_1)e + f_1 x_2 + f_1\theta_{11}\varsigma_{11}(x_1) \\ + f_1\theta_{12}\varsigma_{12}(x_1) + f_1 g_1 x_3 + f_1\Delta_1(x_1,t) \\ + f_2 e b_2 x_3 + f_2 e\theta_{21}\varsigma_{21}(x_1) \quad (25) \\ + f_2 e\theta_{22}\varsigma_{22}(x_1) + f_2 e\Delta_2(x_1,t) \\ - f_3 eax_3 + au_{dd}f_3 e + a(u - u_{dd})f_3 e$$

It can also be further changed as

$$\dot{S} = -f(S) + f_1\tilde{\theta}_{11}\varsigma_{11}(x_1) + f_1\tilde{\theta}_{12}\varsigma_{12}(x_1) \\ + f_1\tilde{g}_1 x_3 + f_1\Delta_1(x_1,t) + f_2 e\tilde{b}_2 x_3 \\ + f_2 e\tilde{\theta}_{21}\varsigma_{21}(x_1) + f_2 e\tilde{\theta}_{22}\varsigma_{22}(x_1) \quad (26) \\ + f_2 e\Delta_2(x_1,t) - \eta_1(x_1,t) - \eta_2(x_1,t) \\ + a(u - u_{dd})f_3 e$$

Then it means that

$$S\dot{S} = -Sf(S) + Saf_3 e(u - u_{dd}) + SE_\Delta \quad (27)$$

where

$$E_\Delta = f_1\tilde{\theta}_{11}\varsigma_{11}(x_1) + f_1\tilde{\theta}_{12}\varsigma_{12}(x_1) \\ + f_1\tilde{g}_1 x_3 + f_1\Delta_1(x_1,t) + f_2 e\tilde{b}_2 x_3 \\ + f_2 e\tilde{\theta}_{21}\varsigma_{21}(x_1) + f_2 e\tilde{\theta}_{22}\varsigma_{22}(x_1) \quad (28) \\ + f_2 e\Delta_2(x_1,t) - \eta_1(x_1,t) - \eta_2(x_1,t)$$

Design the regulation law as follows

$$\dot{\hat{\theta}}_{11} = Sf_1\varsigma_{11}(x_1), \dot{\hat{\theta}}_{12} = Sf_1\varsigma_{12}(x_1) \quad (29)$$

$$\dot{\hat{\theta}}_{21} = Sf_2 e\varsigma_{21}(x_1), \dot{\hat{\theta}}_{22} = Sf_2 e\varsigma_{22}(x_1) \qquad (30)$$

$$\dot{\hat{g}}_1 = Sf_1 x_3, \dot{\hat{b}}_2 = Sf_2 e x_3 \qquad (31)$$

Considering that

$$\left| f_1 \Delta_1(x,t) \right| \le \psi_1 p_1(x)\left| f_1 \right| \qquad (32)$$

$$\left| f_2 e \Delta_2(x,t) \right| \le \psi_2 p_2(x)\left| f_2 e \right| \qquad (33)$$

We design

$$\eta_1(x_1,t) = \beta_{11}(S,x_1,\hat{\Omega}_{11})$$

$$\eta_2(x_1,t) = \beta_{12}(S,e,x_1,\hat{\Omega}_{12}) \qquad (34)$$

$$\beta_{11}(S,x_1,\hat{\Omega}_{11}) = \hat{\Omega}_{11}\omega_{11}$$

$$\omega_{11} = q_{11}(x_1)\tanh(\frac{Sq_{11}(x)}{\varepsilon_{11}}) \qquad (35)$$

$$\beta_{12}(S,x_1,\hat{\Omega}_{12}) = \hat{\Omega}_{12}\omega_{12}, \omega_{12} = q_{12}(x_1)\tanh(\frac{Sq_{12}(x)}{\varepsilon_{12}}) \quad (36)$$

$$q_{11} = p_1(x)\left|f_1\right|, q_{12} = p_2(x)\left|f_2 e\right| \qquad (37)$$

where

$$
\begin{aligned}
& Sf_1\Delta_1(x_1,t) - S\eta_1(x_1,t) \\
& + Sf_2 e\Delta_2(x_1,t) - S\eta_2(x_1,t) \\
& = Sf_1\Delta_1(x_1,t) + Sf_2 e\Delta_2(x_1,t) \\
& - S\beta_{11}(S,x_1,\hat{\Omega}_{11}) - S\beta_{12}(S,e,x_1,\hat{\Omega}_{12}) \\
& = Sf_1\Delta_1(x_1,t) + Sf_2 e\Delta_2(x_1,t) \\
& - S\hat{\Omega}_{11}q_{11}(x)\tanh(\frac{Sq_{11}(x)}{\varepsilon_{11}}) \\
& - S\hat{\Omega}_{12}q_{12}(x)\tanh(\frac{Sq_{12}(x)}{\varepsilon_{12}}) \\
& \le \psi_1\left|Sq_{11}(x)\right| + \psi_2\left|Sq_{12}(x)\right| \\
& - S\hat{\Omega}_{11}q_{11}(x)\tanh(\frac{Sq_{11}(x)}{\varepsilon_{11}}) \\
& - S\hat{\Omega}_{12}q_{12}(x)\tanh(\frac{Sq_{12}(x)}{\varepsilon_{12}}) \\
& < (\psi_1 - \hat{\Omega}_{11})S\omega_{11} + (\psi_2 - \hat{\Omega}_{12})S\omega_{12} \\
& < \tilde{\Omega}_{11}S\omega_{11} + \tilde{\Omega}_{12}S\omega_{12}
\end{aligned}
\qquad (38)
$$

Define $\tilde{\Omega}_{11} = \psi_1 - \hat{\Omega}_{11}$, $\tilde{\Omega}_{12} = \psi_2 - \hat{\Omega}_{12}$, and design the adaptive turning law as

$$\dot{\hat{\Omega}}_{11} = S\omega_{11}, \dot{\hat{\Omega}}_{12} = S\omega_{12} \qquad (39)$$

Then choose the Lyapunov function as

$$V = \frac{1}{2}S^2 + \frac{1}{2}\sum_{i,j=1,2}\theta_{ij}^2 + \frac{1}{2}\sum_{i=1,2}\tilde{\Omega}_{1i}^2 \\ + \frac{1}{2}\tilde{g}_1^{\,2} + \frac{1}{2}\tilde{b}_2^{\,2} \qquad (40)$$

It is easy to prove that

$$
\begin{aligned}
\dot{V} & \le Saf_3 e(u - u_{dd}) \\
& \le -Saf_3 eN(k)u_{dN} - Su_{dN} \\
& \le -[1 + af_3 eN(k)]u_{dN}S
\end{aligned}
\qquad (41)
$$

Using the integrate operation on both side of the inequality, we get

$$
\begin{aligned}
V(t) - V(0) & \le -\int_{k(0)}^{k(t)}[(1 + af_3 eN(k)]dk \\
& = -k(t) + k(0) - \int_{k(0)}^{k(t)} af_3 eN(k)dk
\end{aligned}
\qquad (42)
$$

Considering the characteristic of Nussbaum gain function as follows

$$\limsup_{s\to\infty}\frac{1}{s}\int_0^s N(x)dx = +\infty \qquad (43)$$

$$\liminf_{s\to\infty}\frac{1}{s}\int_0^s N(x)dx = -\infty \qquad (44)$$

Using the apagoge method, it is easy to prove that $S \to 0$ .According to the lemma 5, we can make the conclusion that the system is stable and the state x_1 can be converged to the desired value x_1^d .

4   Example and simulation

The supersonic model described in[2] is used to do the simulation as follow

$$\dot{\alpha} = \left(\frac{QS}{mV}\right)[C_z(\alpha,M_m) + B_z\delta] + q$$

$$\dot{q} = \left(\frac{QSd}{I_{yy}}\right)[C_m(\alpha,M_m) + B_m\delta]$$

$$\dot{\delta} = au - a\delta$$

The nonlinear sliding mode controller is designed as

$$S = (x_1 - x_1^d) + c_1 \int (x_1 - x_1^d) dt + f(x_2, x_3, z)$$

$$f(x_2, x_3, z) = x_2^2 sign(z), \quad z = \int (x_1 - x_1^d) dt,$$

$$z = \int (x_1 - x_1^d) dt$$

$$u_{dN} = [-x_2 - c_1(x_1 - x_1^d) - f(S) - \hat{\theta}_{11}\varsigma_{11}(x_1) - \hat{\theta}_{12}\varsigma_{12}(x_1)$$
$$- \eta_1(x_1, t) - \hat{\theta}_{21} f_2 \varsigma_{21}(x_1) - \hat{\theta}_{22} f_2 \varsigma_{22}(x_1) - \eta_2(x_1, t)]$$
$$x_{3d} = -N(k) u_{dN}, \quad z_3 = x_3 - x_3^d,$$

$$u = \frac{1}{a}(ax_3 - f(z_3) + \hat{l}_1 S),$$

$$\eta_1(x_1, t) = \beta_{11}(S, x_1, \hat{\Omega}_{11}), \quad \eta_2(x_1, t) = \beta_{12}(S, x_1, \hat{\Omega}_{12})$$

$$\beta_{11}(S, x_1, \hat{\Omega}_{11}) = \hat{\Omega}_{11}\omega_{11}, \quad \omega_{11} = q_{11}(x_1) \tanh(\frac{Sq_{11}(x)}{\varepsilon_{11}})$$

$$\beta_{12}(S, x_1, \hat{\Omega}_{12}) = \hat{\Omega}_{12}\omega_{12}, \quad \omega_{12} = q_{12}(x_1) \tanh(\frac{Sq_{12}(x)}{\varepsilon_{12}}),$$

$$q_{11} = p_1(x), \quad q_{12} = p_2(x)|f_2|,$$

$$\dot{\hat{\Omega}}_{11} = S\omega_{11}, \quad \dot{\hat{\Omega}}_{12} = S\omega_{12},$$

$$\dot{\hat{\theta}}_{11} = S\varsigma_{11}(x_1), \quad \dot{\hat{\theta}}_{12} = S\varsigma_{12}(x_1),$$

$$\dot{\hat{\theta}}_{21} = Sf_2\varsigma_{21}(x_1), \quad \dot{\hat{\theta}}_{22} = Sf_2\varsigma_{22}(x_1),$$

$$\dot{\hat{g}}_1 = Sx_3, \quad \dot{\hat{b}}_2 = Sf_2 x_3,$$

The definition of parameters is as follows:

$$[\varsigma_{11} \quad \varsigma_{12} \quad \varsigma_{21} \quad \varsigma_{22}]$$
$$= [\phi_{z1b}(x_1) \quad \phi_{z2b}(x_1) \quad \phi_{m1b}(x_1) \quad \phi_{m2b}(x_1)]$$

$$p_1(x_1) = [|\phi_{z1}(x_1)| + |\phi_{z2}(x_1)M_m|],$$
$$p_2(x_1) = [|\phi_{m1}(x_1)| + |\phi_{m2}(x_1)M_m|]$$

Choose the desired value of attack angle as $\alpha^d = x_1^d = 2^0$, and assume the uncertainties of aeronautical parameters as $p_1 = p_2 = p_3 = p_4 = p_5 = 0.5$, finally the simulation result is as Fig.1, Fig.2, Fig.3 and Fig.4.
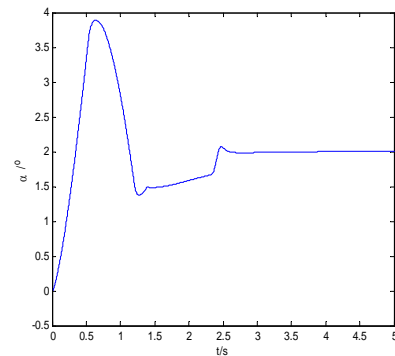
Fig. 1 Angle of attack response
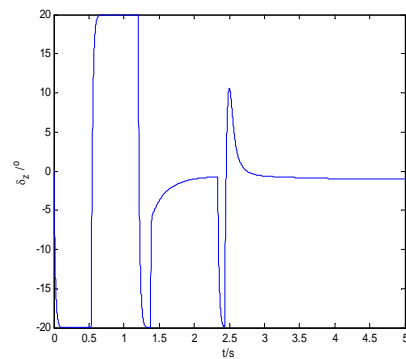
Fig. 2 Actuator response
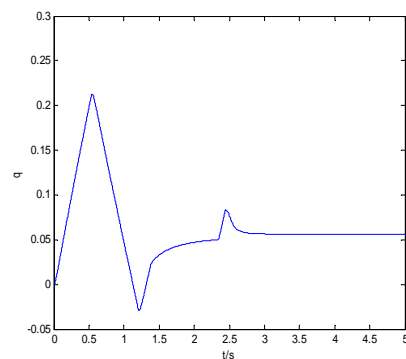
Fig. 3 Rotation response
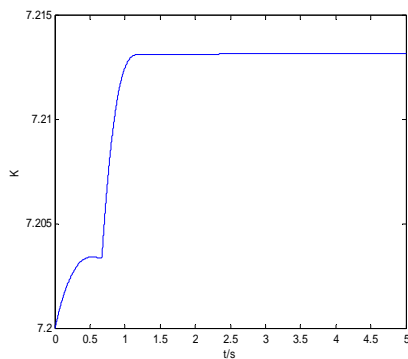
Fig. 3 Nussbaum gain curve
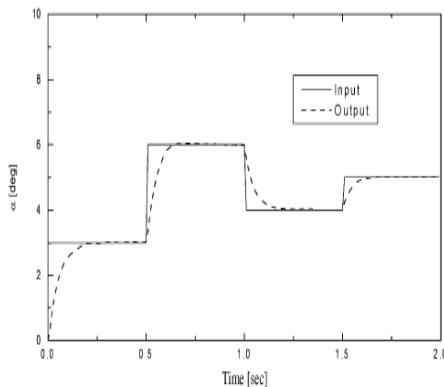
Fig. 5 Angle of Attack Response

By using the method in [12], The response curve of attack angle is shown as fig. 5. So we can find that the curves are not very smooth with our method. Also it has oscillation because of the searching characteristic when the Nussbaum gain method is used. It should be pointed out that the uncertainties considered in [12] is weaker than it in this paper, also the unknown control direction problem is not considered in [12].

5   Conclusions

By using the sliding mode method, a three order model was changed to a one order sliding mode, so the differential bomb problem ,which is often caused by using backstepping technology, was avoided . Also, it does not need to measure the high order derivative of error because of the constructing of nonlinear sliding mode.

It is worth pointing out that both the unknown control direction problem and the control singularity problem were solved by introducing the Nussbaum gain method and nonlinear sliding mode. The construction of nonlinear mode, which can be extended and used in a large family of nonlinear systems, is very meaningful and skillful. So the both methods were integrated perfectly in this paper. Without the nonlinear sliding mode method, the order of the system can not be reduced; without the using of Nussbaum gain

method, the unknown control direction can not be solved and control singularity problem will appear. So neither of the two methods is dispensable.

Acknowledgment

6   References

[1]   Charalampos P. Bechlioulis, George A. Rovithakis, adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems , Automatica 45(2009)532-538

[2]   C. C. Fuh and P. C. Tung, "Controlling Chaos Using Differential Geometric Approach," Physical Review Letters, vol. 75, no. 16, pp. 2952-2955, 1995.

[3]   Ge, S. S., & Wang, J. (2003). Robust adaptive tracking for time-varying uncertain nonlinear systems with unknown control coefficients. IEEE Transactions on Automatic Control, 48(8), 1463_1469.

[4]   Ge, S. S., Hong, F., & Lee, T. H. (2004). Adaptive neural control of nonlinear time-delay system with unknown virtual control coefficients.IEEE Transactions on Systems, Man, and Cybernetics-PartB: Cybernetics, 34(1), 499– 516.

[5]   Jagannathan S, He P. Neural-network-based state-feedback control of a nonlinear discrete-time system in nonstrict feedback form. IEEE Transactions on Neural Networks 2008;19(12):2073_87.

[6]   KE Hai-sen, YE Xu-dong, Robust adaptive controller design for a class of nonlinear systems with unknown high frequency gains, Univ SCIENCE A 2006 7(3):315-320

[7]   Liu Yunguang, Output-feedback Adaptive Control for a Class of Nonlinear Systems with Unknown Control Directions, ACTA AUTOMATICA SINICA, Vol. 33, No. 12, 1306-1312

[8]   M. Krstic, J. Sun, and P. V. Kokotovic, "Control of Feedback Linearizable Systems with Input Unmodeled Dynamics," Proc. of the 33rd Conference on Decision and Control, Lake Buena Vista, FL, pp. 1633-1638, 1994

[9] Nussbaum, R. D. (1983). Some remarks on the conjecture in parameter adaptive control. Systems and Control Letters, 3(3), 243– 246

[10] Polycarpou, M. M., Ioannou, P. A. A robust adaptive nonlinear control design. Automatica, Vol. 32. No. 3 (1996) 423-427

[11] Qian, C., & Lin, W. (2002). Output feedback control of a class of nonlinear systems: A nonseparation principle paradigm. IEEE Transactions on Automatic Control, 47, 1710– 1715

[12] Seung-Hwan Kim, Yoon-Sik Kim, Chanho Song, A robust adaptive nonlinear control approach to missile autopilot design, Control engineering practice, 12(2004) pp. 149-154

[13] Sahjendra N. Singh, Marc Steinberg, Robert D. Digirolamo, Variable Structure Robust Flight Control System for the F-I4, IEEE tansactions on aerospace and electronic systems vol. 33, No. 1 , 1997

[14] T.P.Zhang , S.S.Ge Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs, Automatica 43(2007) 1021-1033

[15] Tsinias, J. (1991). A theorem on global stabilization of nonlinear systems by linear feedback. System and Control Letters, 17, 357-362

[16] Weisheng Chen, Adaptive NN control for discrete-time pure-feedback systems with unknown control direction under amplitude and rate actuator constraints, ISA Transactions 48 (2009) 304_311

[17] Xu H J and Mirmirani M.Robust adaptive sliding control for a class of MIMO nonlinear systems [A].In: AIAA Guidance,Navigation,and Control Conference and Exhibit[C].Montreal, Canada, AIAA 2001-4168.

[18] Xuedong Ye, Global adaptive control of nonlinear systems with unknown control directions—non – overparameterization design, Journal of China Jiliang University,Jan, 2005 Vol.14,No,1

[19] Ye, X. D., & Jiang, J. P. (1998). Adaptive nonlinear design without a prioriknowledge of control directions. IEEE Transactions on Automatic Control, 43(11), 1617– 1621.

[20] Yan Li , YangQuan Chen,When is a Mittag_Leffler function a Nussbaum function? Automatica 45 (2009) 1957_1959

[21] Yang C, Ge SS, Xiang C, Chai TY, Lee TH. Output feedback NN control for two classes of discrete-time systems with unknown control directions in a unifiedapproach. IEEE Transactions on Neural Networks 2008;19(11):1873_86.

SESSION

EVOLUTIONARY SYSTEMS, GENETIC ALGORITHMS, GENETIC PROGRAMMING AND APPLICATIONS

Chair(s)

TBA

Artificial Intelligence Approaches to Music Composition

Richard Fox and Adil Khan
Department of Computer Science
Northern Kentucky University, Highland Heights, KY 41099

Abstract – *Artificial Intelligence (AI) techniques have been successfully applied to a wide range of problems that perform problem solving such as diagnosis, decision making and optimization problems. However, any AI algorithm applied to a creative problem requires some mechanism to substitute for the creative spark found in the human, as the computer has no creative capacity. Randomness, as supplied by a random number generator, cannot be the sole mechanism to bring about a creative composition. This paper examines three AI approaches applied to music composition. Specifically, the paper introduces MAGMA, a knowledge-based system that uses three different AI algorithms to generate music. Sample songs generated by MAGMA are compared.*

Keywords: music composition, Markov models, routine planning, genetic algorithms, artificial intelligence

1 Introduction

Artificial Intelligence (AI) research primarily produces problem solving systems that are aids in human decision making and identification. AI is often applied to problems such as diagnosis, design, and optimization. These problems often have specific solutions so that the AI system results can be tested against similar solutions as generated by human experts. AI research has also been applied to problems involving creativity such as composition of poetry or music. In such cases, output must be judged through subjective standards. However, when AI is used to solve creative problems, the AI algorithm must substitute something in place of the creative spark that humans seem to have. Randomness by itself will not be sufficient.

Similar to other areas of AI research, AI applied to music composition varies greatly by both the AI method(s) employed and the specific problem being solved. At one end of the music composition spectrum is the use of AI to generate a musical accompaniment in real time to musical performance [1, 2, 3]. At the other end are AI systems that compose music from scratch [4, 5, 6]. Also in the mix are AI systems which are trained based on previously composed pieces of music (e.g., classical music compositions of Bach and Beethoven), which then generate similar sounding compositions [7, 8]. AI-based music composition should not be confused with other forms of computer-generated music such as the use of MIDI technology or digital samplers and synthesizers.

Music composition in humans is performed in several different ways. Some people "hear" music in their mind and transcribe it onto sheet music. Others plan out the music through a deliberative process. Still others create music spontaneously through "jamming". Or, musicians might combine these approaches. We might view these three different approaches as the application of music theory, the application of planning, and the application of randomness with pruning of music that "doesn't work" as determined by some fitness evaluation. In AI, we might implement these three approaches using knowledge-based reasoning coupled with a stochastic approach, routine planning and genetic algorithms respectively. In the most recent AI research into music composition, the primary method utilized is the genetic algorithm (or some variation). Other approaches include neural networks, fractal geometry, and stochastic approaches using Markov chains.

There are many works that have applied genetic algorithms to music generation, including for instance GenDash [4] which models each measure of a song as a different population to evolve, or the work of Donnelly and Sheppard [5] which evolves four-part harmonies and their rhythm. CONGA [6], unlike the previous two mentioned systems, uses humans to perform the fitness analysis of each piece of music, thus drawing upon more than music theory to evaluate how listenable a piece of music is. AMUSE is used to generate and evolve improvised melodies given a harmonic context [9] and BlueJam employs a combination of heuristics and genetic algorithms [10].

There are numerous examples of stochastic approaches to music generation. For instance, the Stochos system uses eight different types of stochastic functions [11]. Bell applied Markov chains to control pitch, duration and chords of music that was generated by genetic algorithm [12].

The research presented here consists of three different AI techniques applied to the music composition problem. Specifically, this paper describes a system called MAGMA, the Multi-AlGorithmic Music Arranger, which utilizes three AI algorithms to generate music: stochastic generation via Markov chains, routine planning, and genetic algorithms.

The paper examines these AI-based algorithms in section 2. Next, the paper introduces MAGMA and describes how these algorithms are used to implement music creation. In section 4, two different sets of song excerpts are examined to compare the capabilities of the three different algorithms. Section 5 offers some conclusions and future work. The research reported here is a work in progress. The goal of this research is not to indicate that any single approach is better than another but to demonstrate how these approaches can be utilized to compose music.

2   Background

This section introduces the three algorithms that are used in the MAGMA system to compose music. These algorithms are described independently of music composition, saving those details for section 3.

A Markov chain is a state transition diagram whose links (edges) are annotated with probabilities [13]. The entire Markov chain represents a statistical model. The Markov chain can then be used to generate a possible sequence of events and the probability that the sequence will arise. Figure 1 illustrates a simple Markov chain of daily weather patterns. Given the Markov chain, one could generate the probability of a sequence of daily weather patterns. For instance, the probability given that today is sunny of the next three days being sunny, sunny, snowy, would be .4 * .4 * .2 whereas the probability of the next three days being rainy, rainy, sunny would be .4 * .6 * .3.
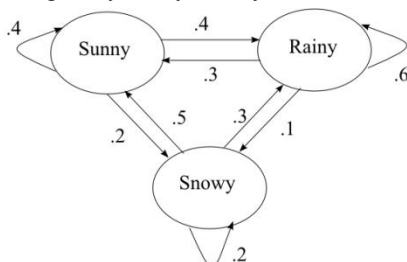
Figure 1: A Markov Chain

The statistical model can be easily generated from a database of events. For instance, the data from figure 1 could come from several months of weather data. Although the Markov chain is not specific to AI, it has found a number of uses within AI. Most commonly, a variation called a hidden Markov model (HMM) adds prior probabilities for each state and evidential probabilities of the likelihood that a given node is reached given the data. The HMM can be used to generate the most likely explanation for the appearance of some data. HMMs have found tremendous success in speech recognition while Markov chains have a common application to generating music.

Routine planning (or routine design) [14] captures the prototypical sequence of problem solving activities that a domain expert might undertake in planning or designing an artifact. It is a knowledge-based approach that describes the solution to a problem using the routine knowledge that the domain expert will compile through years of experience in solving variations of the problem. Routine design/planning has been used to solve a number of routine problems from air cylinder design to air force mission planning to nutritional meal planning.

The actual knowledge of the knowledge base breaks into several categories. First is a hierarchy of the components involved in the routine design or planning problem. This represents plan decomposition. The idea is that to construct the given artifact (whether a physical object or an abstract object) requires designing each of the components and subcomponents in turn. Second, to design any given (sub)component, the expert utilizes plan steps specific to that component. There may be many different plan steps for each component. The selection of the appropriate plan step is based on pattern-matching knowledge that identifies the plan step with the greatest chance of providing success in the designed artifact meeting user specifications, along with decisions already made on the design of other components. As component-level interactions may nullify a partial design, redesign steps are available, for instance decreasing a component's size because it is too large to fit within another component. Lastly, failure handling knowledge can be applied in the case that the designed artifact does not completely fulfill its intended function or meet all user specifications.

Figure 2 illustrates plan decomposition of routine design/planning along with plan steps available to design each (sub)component. In this figure, the overall artifact to be designed consists of three components, two of which have subcomponents.
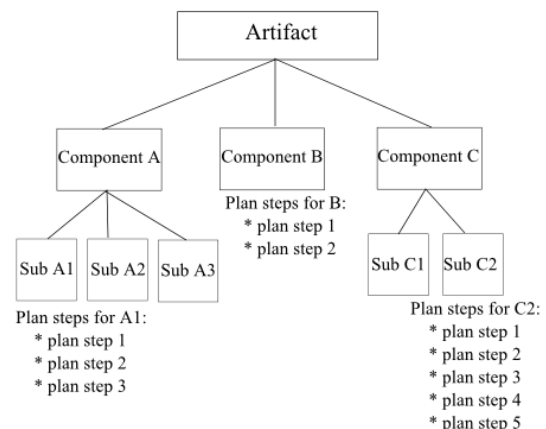
Figure 2: Routine Design Plan Decomposition

The genetic algorithm (GA) uses a form of natural selection to evolve a better solution [15]. The GA models the domain using a string of symbols, often called a

chromosome, along with a fitness function. The GA starts with a base set of chromosomes. This population is treated as parents. The parents generate children through a series of genetic operations (mutation, inversion and crossover) that manipulate select parent chromosomes. In mutation, one or more values in a chromosome are randomly changed. For inversion, a sequence of chromosome values are reversed or otherwise rearranged. Crossover swaps portions of the chromosomes found in two parents so that two variations are generated as children. The result of these operations is a new population of children making up the next generation. These child chromosomes are evaluated using a fitness function. The fitness function models aspects of the domain so that the resulting value indicates how good the particular child is. A selection mechanism is then used to select the children who should survive into the next generation. Those children become the parents and the cycle continues. Selection algorithms include selecting the highest evaluated children, randomly selecting children, using rankings as probabilities for selection, and looking for diversity among the children being selected. The GA strategy is applied over and over for some number of iterations (generations) or until a child has been found whose fitness exceeds some desired threshold.

There are many open questions when solving problems with GAs. These include how one models the domain with a chromosome (which features are modeled? what values are permitted?), which genetic operations are applied and how often/to what extent, what fitness function should be used, what selection mechanism(s) is applied, how many children are generated, and how many children are selected to become parents. GAs have been applied to a wide variety of problems including optimization problems, planning/design problems, and a variations of GAs are often applied to automatic generation of program code. Section 3 elaborates on how these three algorithms can be applied to music composition.

3   MAGMA

MAGMA (Multi-AlGorithmic Music Arranger) is an experimental AI system currently under construction to compose music, specifically pop songs. The system uses a stochastic algorithm based on Markov chains, a routine planning algorithm, and a genetic algorithm to generate songs. The system takes user specifications as input and generates a song in the form of a MIDI file.

The user specifications determine what type of song the user is interested in. There are five preferences, each rated on a 5-point scale. These preferences are as follows.

- Transition determines the size of a transition from chord to chord or note to note. The transition value can be thought of as the maximum "step size" between any two chords or notes. A higher transition makes the song sound more "volatile" and might cause a song to have a more dissonant sound while a small transition might create a boring song.

- Repetition dictates how likely chords and notes might repeat before transitioning to other chords or notes. The lower the repetition, the more "creative" a song might sound. Repetition is also applied to song structure so that a lower repetition would lead to song components that are not repeated as often.

- Variety applies to chords, notes and song structure, impacting how many chords/notes/song components are generated. A higher variety leads to a song that has more parts and more diversity within those parts. The lower the variety, the simpler the song.

- Range applies to the chords and notes, similar to repetition and variety, but in this case it influences the chords and notes over the entire song. Range also controls the number of octaves that might be applied as well as the instruments selected for the MIDI file.

- Mood impacts the key and the tempo of the song. A more somber mood may cause a minor key and a slower tempo. A more upbeat mood would often result in a major key and a faster tempo. An intermediate mood may cause minor and major keys for different song components such as a minor key for the verse and a major key for the chorus. The mood also impacts the instruments selected.

The user also specifies which of the three algorithms to utilize. MAGMA will generate a single song using the selected algorithm (stochastic approach, planning approach, genetic algorithm approach). The output of MAGMA is a MIDI file of a song that might vary in duration from 2 minutes to 5 minutes. See figure 3 for an overview of the system's architecture.
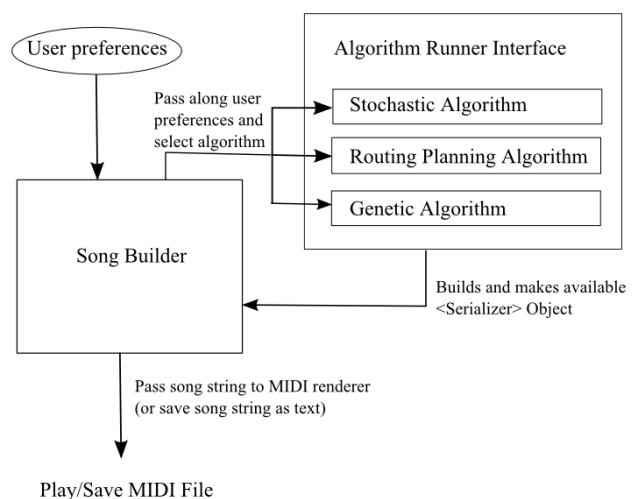
Figure 3: MAGMA's Architecture

Each of the three algorithms generates a song through planning decomposition, as shown in figure 4. First, the

song's structure is generated. Song structures are based on a few components: introductions (I), verses (V), choruses (C), bridges (B), solo sections (S) and outros (O). A simple song may have a structure of I-V-C-V-C-O and a more complex song might have a more elaborate structure of I-V-V-C-V-B-C-S-C-O. In some cases, an intro and/or outro may match a verse and in other cases, the intro, outro and verse may all differ. Additionally, a song may contain a modulation either between or within components, for instance by having two repeated choruses shift from the key of D to the key of E.
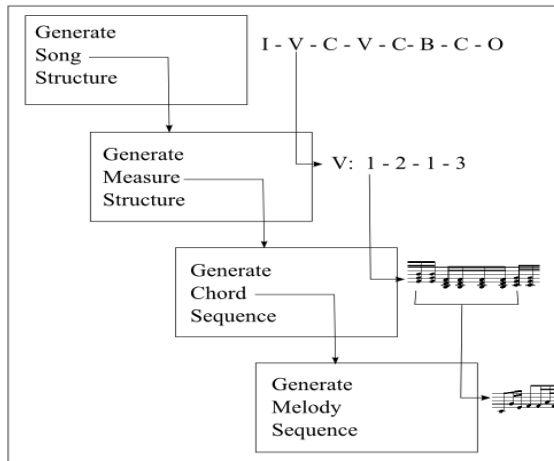
Figure 4: Four Steps to Generate a Song

Given the song structure, or song components, the next step is to generate the structure of each of those components. This structure is based on the number of and types of measures. For instance, a verse might consist of 4 or 8 measures. These measures may all be identical, or they may alternate between different chord sequences. A 4-measure component may follow a pattern like 1-1-1-2 or 1-2-1-3 or 1-1-1-1 or even 1-2-3-4. Obviously, repetition and variety impact the generation of the component's structure.

Now MAGMA must generate the actual chords for each of the measures. If a verse has a pattern of 1-1-1-2, then MAGMA must generate two chord sequences for the verse. The measure will comprise some number of "beats". In 4/4 timing for instance, a measure might consist of 4 quarter notes or 8 eighth notes or 1 whole note. The variety will help dictate whether all chords have the same duration or whether some chords will be longer and/or shorter than others. A more diverse measure might consist of a half note followed by a quarter note followed by two eighth notes.

Given the chords, MAGMA now generates a melody sequence over those chords. The melody is generated independently of the chord sequence but must match the chord sequence's duration. The notes generated are based on the key that makes up the component (e.g., if the verse is in the key of Am, the notes generated must match Am). In

this way, the notes are not necessarily based on the chord that the notes are played over.

All chord and melody data are generated using Nashville notation so that they are key agnostic (using numbers in place of notes). This permits easy translation of a note or chord sequence based on the generated key. The key is generated randomly, but is impacted by the user's preference for mood. Selections for tempo and musical instruments (for the MIDI file) also combine mood and randomness.

Each of the three algorithms (stochastic, planning, genetic algorithm) performs these four plan decomposition steps in different ways. However, all three of the algorithms perform these steps in the order given in figure 4.

For the stochastic algorithm, each of the four steps is handled through separate Markov chains, each of which was automatically generated using a simple parsing algorithm from a collection of pop songs. For instance, the chord transition probabilities were generated by parsing the chords of numerous songs. The parser provides the Markov chains as transition probabilities matrixes, as shown in figure 5. MAGMA utilizes one matrix for song structure, one for measure structure, one for chord generation and one for melody generation. There are additional matrices for phrase transition (e.g., chorus to verse chord transitions), and chord duration. For each generated song, the probabilities of the matrices are altered based on the user specifications to better match the user's interests.

	-7	-6	-5	.	.	.	12	13	14
-7	.11	.21	.31	.	.	.	.33	.18	.17
-6	.1	.32	.39	·	·	·	.43	.28	.67
-5	.7	.2	.9	·	·	·	.25	.46	.72
.	·	·	·				·	·	·
.	·	·	·				·	·	·
.	·	·	·				·	·	·
12	.11	.13	.2	.	.	.	.6	.23	.17
13	.4	.14	.82	.	.	.	.1	.2	.11
14	.19	.11	.5	.	.	.	.9	.21	.16

Figure 5: Partial Transition Matrix for Notes/Chords

The planning approach has numerous plan steps that define specific sequences of song components (the song structure), measures, chord sequences and note sequences. The song structure is generated by selecting the sequence that most closely matches the user specification. Next, for each song component, its structure is generated in terms of the pattern of measures. Again, the planning system has a number of enumerated lists such as 1-2-1-2 or 1-1-2-1-1-2-1-3, and sequence selected is the one that most closely matches the user's specifications. Given a single measure, now the planning approach selects a chord sequence that

again most closely matches the user specifications. Similarly, notes are then generated.

Below are three examples of song structure and the user preferences that each would best match. For instance, the first component would proscribe a song that consists of the pattern Introduction, Verse, Chorus, Verse, Chorus and Outro. It has a repetition value of 2 and a variety value of 3. The third example has a repetition value of 1 and a variety value of 5.

- `{R=2,V=3}=I|V|C|V|C|O`
- `{R=2,V=4}=I|V|C|V|C|B|C|O`
- `{R=1,V=5}=I|V|C|B|O`

Chord sequence and melody sequence plans include the pitch (the chord or the note) and the rhythm. What follows are examples of chord sequences. The numbers indicate the chord to play where 1 equals the key (e.g., a key of D would mean that a 2 is an E chord and 5 is an A chord). A 0 indicates a rest, or no chord/note played. The letter following the number represents the duration of the chord or note with w meaning a whole note, h being a half note, q being a quarter note, i being an eighth note, s being a sixteenth note, etc. The two lists below are examples of chord sequences and note/melody sequences respectively.

- `{R=1,T=1,V=2,H=3}=4q|4q|5q|5q`
- `{R=1,T=1,V=2,H=2}=5h|6h`
- `{R=1,T=1,V=1,H=1}=5w`

- `{R=2,V=2,T=2,H=4}=0q|0i|3i|3i|5i|5q`
- `{R=1,V=2,T=1,H=2}=2qqq|-3q`
- `{R=3,V=3,T=1,H=2}=3q|3q|2q|1q`
- `{R=4,V=1,T=1,H=3}=-1q|1qi|0i|-1q`

To further illustrate, a song with the key of A might generate a measure whose chord sequence is A-A-A-G-G-G-G-E and consist solely of eighth notes expressed as 1e|1e|1e|7e|7e|7e|7e|5e. Another example might be a song with a key of Cmaj with a measure whose chord sequence is D (half note), E (3 quarter notes) F (half note). In the latter case, this would be expressed as 2w|3qqq|4h.

In order to ensure that the melody generated matches to some extent the chords generated, three additional mechanisms are applied. First, the duration of a measure, as generated during the chord sequence phase, is used to "chop off" a melody. That is, if a selected melody plan step is too lengthy, its duration is simply truncated upon reaching the end of the chord sequence. Second, the melody must match the key of the song component (e.g., the key of the verse). This may or may not precisely match the chords and so additional work may be required to perform "failure handling" to ensure the notes and chords work together with

respect to music theory. Finally, transition rules are applied to alter the sequence of chords as they move from one song component to another (e.g., verse to chorus transition).

The Genetic Algorithm approach employs four types of chromosomes, one each for the song structure, component structure, chord sequences and note sequences. Five fitness functions are utilized, one per type of user preference. The fitness of a chromosome is based on how closely it matches the given user preference. The five fitness function values are combined using a weighted average.

The song structure is generated with an initial chromosome size based on the user preference. If the preference is for low variety and low repetition for example, the size will be smaller than a preference of high repetition and high variety.

For each song component (verse, chorus, etc.) a measure sequence is evolved. The chromosome's size will depend on user preferences. A longer chromosome will be generated for high repetition and variety, while a shorter chromosome for low repetition and variety. At this level, each member of the chromosome represents a measure. So if the final chromosome is evolved to 1,1,2,1,3,2, then three distinct measures make up that song component.

Given a single measure, the next step is to generate its chord sequence. The chord sequence is influenced by user preference of repetition, variety and transition. Repetition and variety will also influence the size of the chromosome (chord sequence).

Once the chord sequence has been generated, a melody is evolved for each chord in that sequence. The chromosome comprising the melody is influenced by repetition, variety and transition. If the duration of the melody is larger than the underlying chord then it is trimmed to match the chord duration. If it is shorter, then the last note is padded to match the duration of the chord. The process of chord and then melody generation is repeated for every measure of the song.

During each phase of the genetic algorithm, every chromosome is scored by the fitness function, generating a real number between 0 and 1. This value is the combined weighting of the fitness functions that encode the user specifications applied for that portion of the song composition. For instance, music theory, diversity, repetition and transition are all applied for chord and note sequences while range, mood, diversity, transition and repetition are used to generate the song's structure.

A chromosome for a chord or melody sequence is denoted using Nashville notation which facilitates easy computation of transitions, distinct notes, and so forth. As

an example, a chromosome storing the sequence 1, -3, 1, 5, 7, 9, 12 has almost no repetition but contains a high degree of transition. Negative numbers in the sequence represent lower octaves while numbers greater than 7 represent higher octaves. The use of the Nashville notation also simplifies music theory computations irrelevant of the key.

For each portion of song generation by the genetic algorithm (refer back to figure 4), the initial population size is 10 chromosomes. The initial population is created randomly. For each generation, four parents are selected from the base population, two of which have the highest fitness and two of which are the most diverse from the two parents. The four parents produce six children by crossover, and four additional children through mutation. For song structure and song component generation, 10-15 iterations are performed. For chord and melody sequence generation, 100 iterations are performed. The rationale being that structure is more easily evolved than the chord and note sequences.

4   Examples

This section examines and compares two sets of songs generated by MAGMA. For the first set of songs, the user specified high transition, high variety, highly rhythmic, low repetition and high range. Excerpts from the stochastic, planning and genetic algorithm portions of MAGMA are shown in figures 6, 7 and 8 respectively. The excerpts shown in the figures are typical of the whole song from which they were excerpted.

Figure 6: Excerpt from Stochastic Approach

Figure 7: Excerpt from Planning Approach

Figure 8: Excerpt from Genetic Algorithm Approach

Notice in figure 6 that the melody of the stochastic approach consists of rapidly changing notes where the transition of change (step size) between notes is being quite large. This leads to a chaotic, uneven sound. The melody generated by the genetic algorithm (figure 8) is sparser than the other two songs leading perhaps to a more listenable melody. However, the chord sequences from the genetic algorithm have a rhythmic pattern which makes the song less listenable (see how the chords are played continuously in the measure). The song generated from the planning approach (figure 7) offers a compromise between these two extremes. The melody is neither chaotic nor with large step sizes, and the chord structure, while not as diverse as the genetic algorithm approach also does not suffer from a lack of rhythm. In the opinion of the authors, the song generated from the planning algorithm is far more listenable.

The second set of songs was generated using specifications of low transition, low variety, less rhythmic, high repetition and low range. The low range had a large impact on all three generated songs in that they are at a much slower tempo. In the case of the genetic algorithm output, the song was placed in a minor key giving it a somber sound. Excerpts from the three songs as generated by the stochastic approach, planning approach and genetic algorithm approach are shown in figures 9, 10 and 11 respectively.

Figure 9: Excerpt from Stochastic Approach

Figure 10: Excerpt from Planning Approach

Figure 11: Excerpt from Genetic Algorithm Approach

The song generated by the stochastic approach is quite simple to the point of being boring. As can be seen in figure 9, the melody is highly repetitive. In this excerpt, the chords remain the same throughout this passage of music. The song generated from the planning approach is slightly more interesting while remaining basic. Only the genetic algorithm produced a piece of music that has some

variability. And yet, the large step sizes between notes cause this song to lack coherence.

This brief look at the results of the three algorithms cannot lead to any definitive conclusions. However, it should be apparent that the stochastic algorithm is at a disadvantage because it does not apply any explicit strategy to either follow music theory or compositional strategies that make a song listenable. The main detractor of the songs generated by the genetic algorithm is the overly random nature of the notes. The planning approach does not suffer from either of these problems but may lack in originality because it is impacted the least by randomness.

5   Conclusions

MAGMA, the Multi-AlGorithmic Music Arranger, applies three artificial intelligence algorithms to compose pop songs: a stochastic approach using Markov chains, a routine planning approach and a genetic algorithm approach. MAGMA receives user specifications and applies the algorithms to generate songs. Song generation starts with the development of the song's structure. This structure reflects the component parts that will make up the song, such as verses and choruses. Each component is itself decomposed into measures. A component might be made up of several distinct measures such as a pattern of 1-2-1-2. Each measure is then designed by generating both chord sequences and notes (a melody). The generation of song structure, measures, chords and notes is all handled by one of the AI algorithms.

MAGMA is a proof-of-concept system being constructed as part of a master's thesis in computer science. The goal is to demonstrate how these approaches can be applied for music composition, and to compare and contrast the music generated from these algorithms. MAGMA generates songs as MIDI files so that people can both listen to the music and view the music in staff notation. While the analysis provided in this paper can point out flaws with generated music, a survey of listeners can provide a subjective comparison as well. This is intended as future work.

While the point of this paper is not to compare the three approaches, the examples shown do demonstrate a drawback of a strictly stochastic approach to music composition in that there is little to no ability to apply music theory or principles that lead to listenable music. The genetic algorithm approach has an advantage over the planning approach in that it can provide music with a large variability and randomness. The planning approach is perhaps a good middle ground. Ultimately, it is hoped that MAGMA can combine the strengths of the three algorithms to produce the most listenable music.

References

[1] Simon, I., Morris, D., and Basu, S (2008). "MySong: automatic accompaniment generation for vocal melodies." Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems. p. 727-734, ACM.

[2] Assayag, G., Bloch, G, Chemillier, M., Cont, A., and Dubnov, S. (2006). "Omax brothers: a dynamic yopology of agents for improvization learning." Proceedings of the 1st ACM workshop on Audio and music computing multimedia, p. 125-132, ACM.

[3] Blackwell, T (2007). "Swarming and music." Evolutionary Computer Music, p. 194-217, Springer.

[4] Waschka, R (2007). "Composing with Genetic Algorithms: GenDash." Evolutionary Computer Music, p. 117-136, Springer.

[5] Donnelly, P., and Sheppard, J (2011). "Evolving four-part harmony using genetic algorithms." Applications of Evolutionary Computation, p. 273-282, Springer.

[6] Tokui, Nao, and Hitoshi Iba. "Music composition with interactive evolutionary computation." Proceedings of the 3rd International Conference on Generative Art. Vol. 17. No. 2. 2000, Generative Design Lab.

[7] P. Hamel and D. Eck. Learning features from music audio with deep belief networks. In 11th International Society for Music Information Retrieval Conference (ISMIR 2010), 2010.

[8] Zhang, Q., and Miranda, E (2006). "Evolving musical performance profiles using genetic algorithms with structural fitness." Proceedings of the 8th annual conference on Genetic and evolutionary computation, p. 1833-1840, ACM.

[9] Özcan, E., and Erçal, T (2008). "A genetic algorithm for generating improvised music." Artificial Evolution, p. 266-277, Springer.

[10] Rowe, C (2008). "BlueJam: A heuristic-based approach to evolutionary music generation." Technical report, parallaxed.net/bluejam.

[11] Bokesoy, S., and Pape G (2003). "Stochos: Software for Real-Time Synthesis of Stochastic Music." Computer Music Journal 27.3, p. 33-43, MIT Press.

[12] Bell, C (2011). "Algorithmic music composition using dynamic Markov chains and genetic algorithms." Journal of Computing Sciences in Colleges 27.2, 99-107, CCSC.

[13] Hazewinkel, M. ed. (2001), "Markov chain", Encyclopedia of Mathematics, Springer.

[14] Brinkop, A., Laudwein, N., and Maasen R (1995). "Routine Design for Mechanical Engineering." AI Magazine, p 74-85, AAAI.

[15] Holland, J (1992). Adaptation in Natural and Artificial Systems. Cambridge, MIT Press.

Artificial Creativity and Self-Evolution: Abductive Reasoning in Artificial Life Forms

James A. Crowder
Raytheon Intelligence, Information, and Services
16800 E. Centretech Parkway, Aurora, Colorado 80011

Shelli Friess MA, NCC
Relevant Counseling
P.O. Box 4193, Englewood, CO 80155

Abstract - *What contributions can Cognitive Science offer to the understanding of nature of providing creativity to artificial life forms? For this discussion, it is necessary to investigate creative processes from a mechanistic perspective as well as involve subjective elements which cannot, in principle, be described from this perspective. These two basic approaches will be investigated here, focusing the artificial creative process on the nature of artificial abductive reasoning. As an initial hypothesis we will characterize creativity as a self-organizing process in which abductive reasoning occurs through the use of self-organizing, semantic topical maps in conjunction with an abductive neural network, allowing the creation and expansion of well a structured set of beliefs within the artificial system. This process is considered here as part of the establishment of order parameters in the flow of information available to allow artificial life forms to self-organize and infer on sensory information. In this sense, we will argue that a deeper understanding of how self-organizing processes involving abductive reasoning may take place in artificial dynamic systems, and how this can assist in the creation of an artificial creative process within an artificially intelligent artificial life form we refer to as a Synthetic, Evolving Life Form (**SELF**)*

Here we present a self-evolving, abductive, hypothesis-based reasoning framework called the Advanced Learning Abductive Network (ALAN) that provides the ability to mimic human experience-based reasoning.

Keywords: Artificial Life Form, Artificial Creativity, Intuition

1. Introduction

Turing and others have hypothesized that computers cannot be creative, due to the absence of novelty in its flow of information processing. We believe the use of stochastic, possibilistic abductive networks provides a very novel approach to information processing, allowing the artificially intelligent system to vary its information processing flow, depending on the generated hypotheses and continuously recombinant neural fiber network creation process.

The hypothesis we would like to consider here is that creativity is a directly related problem solving activity in which explorations of problem spaces lead to the expansion of belief domains. We believe successful expansion of beliefs in an artificial cognitive system is initiated by algorithms that provide updates of the artificial cognitive system's Conceptual Ontology [Taylor and Raskin 2010]. Here we discuss the general heuristics within the genetic hypothesis generation process that will be used to guide the support and rebuttal informational search processes and problem solving activities; which includes strategies for examining, comparing, altering and combining concepts, strings of symbols, and the heuristics themselves. But what kind of creativity is possible for the Artificial Intelligence (AI) system in this context? We believe the answer is that it is similar to the one which humans experience in our everyday life: the experience of new and original ideas that have value, based on the overall goals, constraints, and mission directives of the environment the AI system is within. Within this context, we put forth the design and implementation of algorithms required for an Advanced Learning, Abductive Network (ALAN) as a candidate to facilitate artificial creativity (i.e., advanced hypothesis generation and testing), and therefore autonomous, real-time decision support, from an objective perspective; the abductive dialectic argument structure providing the inference engine upon which artificial creative reasoning is based.

2. Human vs. Artificial Reasoning

2.1 Human Reasoning Concepts

Human reasoning is dynamic in that there are many processes involved. There are different types of reasoning necessary to allow humans to navigate their world effectively and efficiently. Here we will provide a brief overview; as the topic of human reasoning is vast. So much information comes into the brain at one time that it is impossible to consciously be aware of all of it. Just imagine for a minute how many things the human brain is handling in one instant. We have memories, associations, and habitual ways of thinking. We have beliefs, assumptions, and predictions. We have experiences, past, present, and planned. We have senses and perceptions. We have defense mechanisms and feelings. Our brains are active! It is hard to imagine what all is happening in an instant of experience for a human, but it is essential to explore the possibilities in order to understand how the concepts will translate into artificial reasoning.

2.2 Modular Reasoning

Cognitive modularity seems to have flourished with Fodor [Fodor 1983]. He thought that humans use domain-

specific modules that together form part of the reasoning system within the human brain. According to Fodor, there are conditions for modular cognition; one is that other parts of the brain have limited access to each reasoning module. This type of reasoning is mandatory, innate, shallow and very fast. He also stipulated that each module was fixed to a neural architecture and that information was encapsulated; since other modules have limited access to each other. More modern psychology believes that cognitive modularity as actually massive modularity. This school of thought suggests that the mind is even more modular with specific functions and specialization [Garica 2007]. This type of Modular Reasoning is used within the SELF Sensory Processing, before Sensory Integration. There is differing views on massive modularity. According to Raymond Gibbs and Van Orden [Gibbs and Orden 2010], massive modularity theory has its problems empirically. They state that the studies fail to be able to separate modules. They also argue that massive modularity theory fails to discover input criteria and state that it may be impossible given the nature of context embedded human nature. Lastly they argue that massive modularity does not acknowledge the interaction of brain, body, and world in human thinking.

2.3 Distributed Reasoning

The distributive theory suggests that there is more to the brain than separate modules. Beyond some very specific areas such and motor control, distributed reasoning theory suggest there are many fuzzy connections between systems of the brain. The distributive theory challenges boundaries of the mind, skull, and even body. taking into account the environment, artifacts, people. This theory is reflected in the Fuzzy, Possibilistic Abductive Network utilized within the SELF cognitive framework.

The distributive reasoning theory by Hutchins [Hutchins and Lintem 1995] provides some insights into human reasoning. Hutchins provides five different models that affect human reasoning. First, he postulates that there are modules within the brain that are specialized in function and structure and are united in a complex way. Second, he argues that cognition at a macro level is distributed outside the individual; such as the media. Media can be internal and external. Third, there is human culture which influences the individual. Fourth, there is society which cognitive activity is distributed in tools, rules, and contexts. Finally, he argues that cognition is distributive in time, both vertical and lateral time dimensions of the subject [Rogers 1997].

Yvonne Rogers [Rogers 1997] provides a detailed analysis of the distributive cognitive model. Rogers cites Hutchins as creating a computational model of two modules of the brain that can together recover depth that neither module alone could do. One general assumption of the distributive

human cognitive system is that it is made of more than one module and that each module in the cognitive system has different cognitive properties than the individual, and is different than the cognitive brain as a whole. Another general assumption made by Rogers is that members (modules) of the system have knowledge that is both variable and redundant and that members of the system can pool resources. Another is distribution of access to information. This enables the coordination of expectations and coordination of action within the human biological reasoning framework [Rogers 1997]. These concepts are utilized throughout the SELF, which utilizes localized processing modules (processing "experts") as well as distributed Cognitive Perceptron Intelligent Software Agents (called Cognitrons) experts that communicate and collaborate throughout the SELF cognitive system.

2.4 Types of Reasoning

Humans can reason in different ways. The three major human reasoning strategies are inductive, deductive, and abductive.

Inductive Reasoning: Inductive reasoning involves coming to a conclusion after evaluating facts; reasoning from specific facts to a general conclusion. This allows for inferences. It also requires human experience to validate any conclusion. An example might be: Zebras that are at the zoo have stripes; therefore all zebras have stripes [DeRaedt 1992].

Deductive Reasoning: Deductive reasoning is just the opposite. Deductive reasoning moves from a general principle to specific cases. This type of reasoning is based on accepted truths. An example of deductive reasoning might be: All zebras have stripes therefore when I go to the zoo the zebra will have stripes.

Abductive Reasoning: Abductive reasoning allows for explanatory hypothesis generation or generating ideas outside of the given facts to explain something that has no immediate satisfactory explanation.

There are many ways in which people reason, but often human reasoning follows either inductive or deductive reasoning. Consider a few ways in which humans think about things. Take cause and effect reasoning where causes and effects are considered. Analogical reasoning is a way of relating things to other novel situations. Comparative reasoning as it implies is comparing things, in which humans often engage. Still another reasoning method is conditional reasoning, or if/then reasoning. Many of us have used the pros and cons methods of reasoning also. Then there is Systemic reasoning where the whole is greater than the sum of its parts. There is also reasoning using examples. As you can see there are

numerous ways in which humans can reason about things and situations. These are all logical ways of reasoning.

2.5 Artificial "SELF" Reasoning

As discussed above, reasoning takes on many forms, but two important ones within the SELF is both induction and abduction:

- Induction: Extrapolates from information and experiences to make accurate predictions about future situations.

- Abduction: Genetic algorithms generate populations of hypotheses and a Dialectic Argument (Tolemin) Structure is used to reason about and learn about a given set of information, experiences, or situations, also called "*Concept Learning*."

Earlier we discussed the use of hypothesis-based reasoning. Here we provide more detail of its architecture and design within an artificial neural structure. Hypothesis-based reasoning structures seek answers to questions that require interplay between doubt and belief, where knowledge is understood to be fallible. This 'playfulness' is the key to searching and exploring information. Utilizing this framework for reasoning about information, hypotheses, and problems provides a robust, adaptive information processing system capable of handling new situations. Here we utilize abductive logic, sometimes called critical thinking, in order to distinguish it from more formal logic methods like deduction and induction. Whereas data mining utilizes induction to develop assertions that are probably true, the dialectic search uses abductive logic methods and processes to develop hypotheses that are possibly true. We do not use Bayesian methods because they cannot measure possibilistics, but measure probabilistic metrics. Instead we utilize a fuzzy implementation of Renyi's entropy and mutual information theory to provide possibilistic measure of mutual information and topical separation [Roberts and Tarassenko 1994].

3. Artificial, Possibilistic Abductive Reasoning

The original McCulloch-Pitts model of a neuron contributed greatly to our understanding of neuron-based systems. However, their model failed to take into account that even the simplest type of human nerve cell exhibits non-deterministic behavior [Newell 2003 & Nishimori, Nakamura, and Shiino 1990]. Some have attempted to take this into account through modeling this as randomness, creating a stochastic neural network, but much of the behavior is not random, but carries a type of imprecision which is associated with the lack of a sharp

transition from the occurrence of an event to the non-occurrence of the event. This leads us to the definition of a network not steeped in Bayesian statistics (a Bayseian Belief Neural Network - BBNN), but one utilizing possibilistics, based on fuzzy characteristics, combined with an abductive, hypothesis-based decision network; and thus creating a Possibilistic, Abductive Neural Network (PANN) [Cooper and Herskovits 1992]. Here we discuss the theory and architecture for a Possibilistic, Abductive Neural Network capable of complex hypothesis generation and testing, leading to artificial creativity and discovery within a SELF [Dimopoulas and Kakas 1996].

3.1 Artificial Creativity in a SELF

Neuroscience research into the human perceptron [Jones 1999] determined that the noise and imprecision in the human nervous system was not, in fact, inconvenient, but was actually essential to the types of computations the brain performed [Crowder and Friess 2012]. The brain learns to make spatio-temporal associations in the presence of noisy, imprecise information, and any artificially intelligent system that tries to emulate human processing must be able to make similar noisy, imprecise associations within its artificial neural systems even when they are not completely specified, have incomplete, imprecise, or conflicting information, as well as taking into account the behavior of the entity, i.e., accounting for its own internal state [Bonarini 1997].

This leads to a Possibilistic, Abductive Neural Network (PANN) [Crowder 2012] that is capable of complex hypothesis generation and testing in the presence of multiple, noise, imprecise, and possibly incomplete information; the types of environments an autonomous SELF is likely to be found. These conditions are typical in real-time processing situations and will be essential for **complex decision support to system operators**, and will be crucial as we move toward autonomous systems that must learn, reason, analyze, and make critical decisions in real-world environments. This work will form the basis for an Advanced Learning, Abductive Network (ALAN) that will mimic human reasoning to provide autonomous, real-time, complex decision support.

4. The Advanced Learning Abductive Network (ALAN)

4.1 Artificial Creativity through Problem Solving

Touring and others have hypothesized that computers cannot be creative, due to the absence of novelty in its flow of information processing. The use of stochastic, possibilistic abductive networks provides a very novel approach to information processing, allowing the artificially intelligent system to vary its information processing flow, depending on the generated hypotheses

and continuously recombinant neural fiber network creation process [Crowder 2010].

One hypothesis we would like to consider here is that creativity is directly related problem solving activity in which explorations of problem spaces lead to the expansion of belief domains. A successful expansion of beliefs is initiated by an update of the cognitive system's Conceptual Ontology [Raskin, Taylor and Hemplemann 2010]. General heuristics within the genetic hypothesis generation process guide the support and rebuttal informational search processes and problem solving activities; which include strategies for examining, comparing, altering and combining concepts, strings of symbols, and the heuristics themselves.

But what kind of creativity is possible for a SELF in this context? We believe the answer is that it is similar to the one which humans experience in our everyday life: the experience of new and original ideas that have value, based on the overall goals, constraints, and mission directives of the environment the SELF is within.

4.2 ALAN Abductive Reasoning Framework

As discussed above, hypothesis-based reasoning is a reasoning framework seeks answers to questions that require interplay between doubt and belief, where knowledge is understood to be fallible. Cognitrons which are capable of learning and reasoning about information, hypotheses, and problems provide a robust, adaptive information processing system capable of handling new situations. The key value of the Cognitrons within the abductive, hypothesis-based reasoning framework is that they provide the ability to learn from sensory data and from each other [Crowder and Carbone 2011]. Using unsupervised learning methods, the Cognitrons have to potential to provide the operations and analytical structures to extract knowledge and context from various sources of information. Cognitrons can be cloned to support as many operators as required and as the system resources allow.

In the abductive, hypothesis-based processes, information is utilized to generate and assess hypotheses from thought processes created by Cognitrons. This is achieved by utilizing the Cognitrons to learn and reason about the hypotheses and information utilizing a Dialectic Argument Structure (DAS) framework shown in Figure 1. Cognitrons, as discussed, are autonomous software agents that create, in essence, an information agent ecosystem, comprehending its external and internal environment and acting on it over time, in pursuit of its own agenda and goals, so as to affect what it comprehends in the future.

Alerts based on the measure of possibility (certainty) of information inform an Interface Cognitron there is information for review. Constructs within the ALAN

framework rank information and flag those that are considered to be the most certain. The review is facilitated by presenting the operator or user with the DAS warrant, backing and links to support and rebuttal sources, traced back through the self-organizing topical maps. This autonomous information search process includes a review process which engages the ALAN cognitive processes, which include critical learning and reasoning objectives which include:

1. Specialization of a DAS to search and track using the signature of a particular Topic of Interest (TOI).
2. Investigate semantic anomalies found in the computation of possibilities that may be caused by the use of information obfuscation techniques.
3. Review of DAS adaptations undertaken by the evolving hypothesis answer structure. When invalid, the DAS learns additional support/rebuttal arguments to prevent the adaptation from re-occurring.
4.
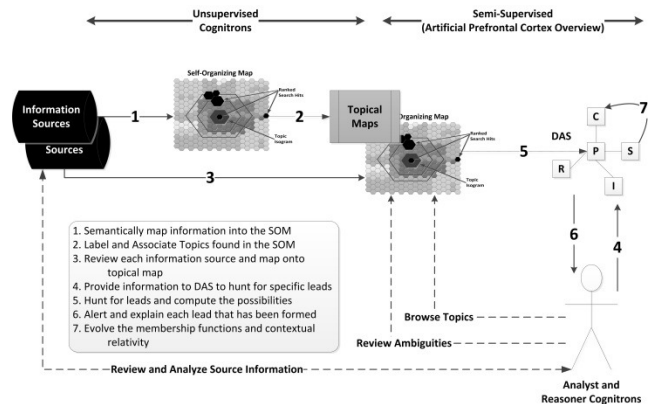
Figure 1 – The ALAN DAS Information Search Process

The ALAN DAS lattice is used to explain the information, compute the overall possibilistics, based on the fuzziness of the support, and rebuttal information, and compute the sensitivity of the claim to the fuzziness of the input data. Being able to review the lattice and assess its sensitivity to the fuzziness of the input data enables the user to effectively assess the quality of the lead. Figure 2 illustrates the DAS fuzzy possibilistic lattice connections.
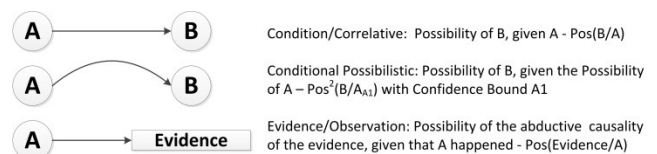
Figure 2 – The Fuzzy Possibilistic Lattice Connections

Utilizing the ALAN cognitive processing environment, the DAS and the Cognitrons mimic human reasoning to process information and develop intelligence. Figure 3 illustrates a high-level view of the architecture for the

ALAN cognitive processing framework. This process includes Search Information Cognitrons that mine through multiple sources to provide data/information to other Cognitrons throughout the ALAN framework. This is called the Federated Search, and is shown in Figure 4.
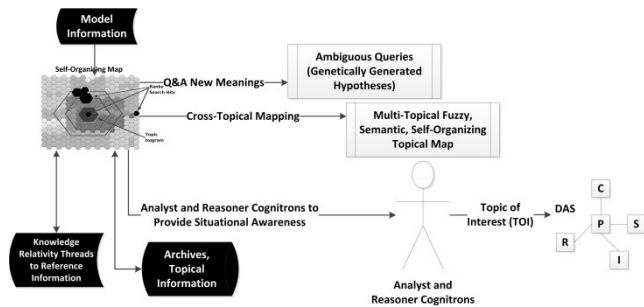
Figure 3 – The ALAN Processing Architecture
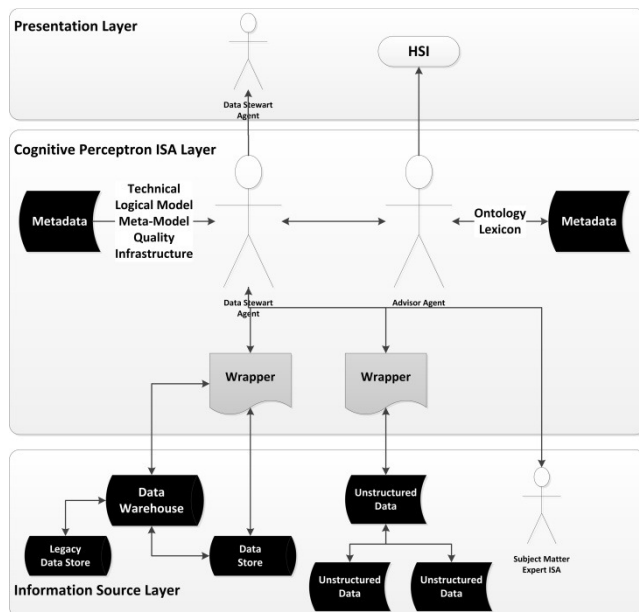
Figure 4 – Federated Search Process within ALAN

5. Conclusions

The ALAN processing environment allows data to be processes into relevant, actionable knowledge. Based on the technologies described above, situational management is one of the most innovative components of ALAN. Utilizing the cognitive framework within ALAN, it can provide real-time processing and display of dynamic, situational awareness information. Information gathering, processing, and analyzing must be done continually to keep track of current trends in the context of current situations, both local and overall, and provide timely and accurate knowledge within a changing environment to allow systems to anticipate and respond to new situations. To achieve the combination of awareness, flexibility, and ability means supporting dynamic and flexible processes that adapt as situations and environments change. This is

possible with the learning and self-evolving ALAN processing and reasoning framework.

6. References

1. Bonarini, A. 1997. Anytime Learning and Adaptation of Structured Fuzzy Behaviors. Adaptive Behavior Volume 5. Cambridge MA: The MIT Press.
2. Cooper, G. and Herskovits, E. 1992. A Bayesian Method for the Induction of Probabilistic Networks from Data'. Machine Learning 9, 309{347.
3. Crowder, J. 2010. Flexible Object Architectures for Hybrid Neural Processing Systems. Proceedings of the 12th annual International Conference on Artificial Intelligence, Las Vegas, NV.
4. Crowder, J. 2012. Possibilistic, Abductive Neural Networks (PANNs) for Decision Support in Autonomous Systems: The Advanced Learning, Abductive Network (ALAN). Proceedings of the 1st International Conference on Robotic Intelligence and Applications, Gwanju, Korea.
5. Crowder, J. and Carbone, J. 2011. Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. Proceedings of the 13th annual International Conference on Artificial Intelligence, Las Vegas, NV.
6. Crowder, J. and Friess, S. 2012. Artificial Psychology: The Psychology of AI. Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics. Orlando, FL.
7. DeRaedt, L. 1992. Interactive Theory Revision: An Inductive Logic Programming Approach. Academic Press.
8. Dimopoulos, Y. and A. Kakas: 1996. Abduction and Inductive Learning. In: L. De Raedt (ed.): Advances in Inductive Logic Programming. Amsterdam: IOS Press, pp. 144{171.
9. Fodor, J., 1983. Modularity and the Mind: An Essay on Faculty Psychology. MIT Press, Cambridge, MA.
10. Garcia, C. 2007. Cognitive Modularity, Biological Modularity, and Evolution. Biological Theory: Integrating Development, Evolution and Cognition, Vol. 2, No. 1, pp. 62-73.
11. Gibbs, R. and Van Orden, G. 2010. Adaptive Cognition without Massive Modularity. Journal of Language and Cognition, Vol. 4, Issue 4.
12. Hutchins, E and Lintem, G. 1995. Cognition in the Wild. MIT Press, Cambridge, MA.
13. Jones B. 1999. Bounded Rationality. Annual Review of Political Science, 1999.2:297-321.
14. Hutchinson, J., Koch, C., Luo, J., and Mead, C. 1988. Computing Motion using Analog and Binary Resistive Networks. CWpirtcr, Vol. 21:3, pp. 52-63.
15. Newell, A. 2003. Unified Theories of Cognition. Harvard University Press, Cambridge, MA.
16. Nishimori, T., Nakamura, T., and Shiino, M. 1990. Retrieval of Spatio-Temporal Sequences in

Asynchronous Neural Networks. Physical Review A, Vol. 41.

17. Raskin, V., Taylor, J. M., & Hemplemann, C. F. 2010. Ausocial engineering. New Security Paradigms Workshop, Concord, MA.

18. Roberts, S. and Tarassenko, L. 1994. A Probabilistic Resource Allocating Network for Novelty Detection. Neural Computation, Vol. 6.

19. Rogers, Y. 1997. A Brief Introduction to Distributed Cognition. University of Sussex Press, Brighton, UK.

20. Taylor, J. M., & Raskin, V. 2010. Fuzzy ontology for natural language. 29th International Conference of the North American Fuzzy Information Processing Society, Toronto, Ontario, Canada.

21. Zadeh, L. 2004. A note of Web Intelligence, World Knowledge, and Fuzzy Logic. Data and Knowledge Engineering, vol. 50, pp. 291-304.

A Parallel Genetic Algorithm to Coevolution of the Strategic Evolutionary Parameters

Eduardo Noronha[1], and **Celso G. Camilo-Junior**[1]
[1]Informatic Institute, Federal University of Goias, Goiania, Goias, Brazil

Abstract—*The strategic choice of parameters (crossover rate, mutation rate, population size, number of generations, among others) in an AG has a direct impact the success of evolutionary search. Therefore the definition of good values in the parameters of evolution can optimize the search process population, finding better solutions in shorter times. Thus a process of self-adaptation constitutes an implicit search with in strategic parameters. Considering the characteristics of evolutionary models and their inherent parallelism, this paper presents a coevolutionary algorithm developed in parallel MPI in order to evolve the parameters of a Multidimensional Knapsack Problem from the evolving capacities of individuals.*

Keywords: Genetic Algorithms, Evolutionary Algorithms, Evolutionary Parameters, Coevolution, Knapsack Problem

1. Introduction

The Genetic Algorithms (GA) are intelligent models inspired by biological evolution through adaptive methods, they can find potential solutions without thoroughly considering all possible solutions to the problem.

The simulation of evolution is guided through the instrument changes, called strategic parameters. Many studious have presented research on the choice of these parameters (crossover rate, mutation rate, number of generations, population size) responsible for controlling the evolutionary process of search.

The focus of this work is to find good values for these parameters, through the development of a Parallel Genetic Algorithm (PGA) coevolutionary.

Several meta-heuristics are used: Evolutionary Strategies [1], Genetic Fuzzy Systems [2], Evolutionary Programming and Genetic Programming [3]. According to Eiben [4], there are basically two ways to choose the parameters in an evolution: by adjusting the parameters, where they are constant from start to finish the process, or by controlling the parameters. In this case, can be deterministic, adaptive and self-adaptive (where the parameters are themselves subjected to an evolutionary algorithm).

The possibility of multiple processors working together to solve a computational problem allows a differentiated approach to impacts from conception and design of algorithms, from the computing platform, to test the mechanisms employed.

The central point of a parallel system is the division of tasks in a single application that can run simultaneously. There are some computer models that allow for parallel computation, and are divided primarily by the presence or not of a synchronization between the processors and the way of communication between them. Parallel computing models were considered in this context primarily by the characteristic of the GA's inherently parallel, allowing multiple combinations of parameters were tested in numerous processors simultaneously. For this, was used the MPI environment for simulating coevolution of strategic parameters, with possible extension of the simulations in CUDA.

In relation to the context of application of the model was named the Knapsack Problem. The motivation for choosing this problem is that, besides being a classic combinatorial problem NP-Hard class, represents numerous possibilities of application in real problems. There are different variations of the knapsack problem (KP 0-1, where each item can be inserted into the knapsack only once, KP unlimited, where each item can be inserted more than once, among others). In this work the variation was chosen Multidimensional Knapsack Problem (MKP).

Considering the behavior of self-adaptive evolutionary systems with their inherent parallelism and potential available in parallel computing, the aim of this paper is to present a coevolutionary algorithm developed in parallel MPI is able to evolve the strategic parameters (crossover rate, mutation rate, population size, number of generation) to a MKP. Some instances of problems in the literature [5] were tested and compared.

2. Literature Review

The self-adaptation was originally introduced by Rechenberg [6] and Schwefel [1] to Evolutionary Strategy, and later by Fogel [7] Evolutionary Programming.

The idea that the parameters of an algorithm can also be submitted to the evolutionary process was approached in 1967 by Bagley [8]. Various methods and techniques of evolutionary self-adaptation are explored, among which the most common for this type of evolution is the Evolution Strategy ([9], [10], [11], [6], [12], [13]), also evolutionary programming, but since the 90's, several studies have been performed Back [14] and Smith [15] showing the capability of doing you with Genetic Algorithms (GA).

In 1970, Weinberg [16] proposed a genetic algorithm external (meta-algorithm) control, responsible for choosing the best strategic parameters for GA slaves. To the master algorithm, each individual in the population represents a possible configuration parameters. To evaluate each individual, the algorithm performs a meta-slave algorithm with that set of parameters, and checks the quality of results. At the end of execution, the individual winner represents the most appropriate configuration parameters. This algorithm has a practical approach Koch [17] for application to problems in the industry.

In his thesis, Potter [18] describes a computational model of cooperative coevolution including the notion of modularity necessary to provide reasonable solutions and evolve interacting coadapted subcomponents. In this approach, subcomponents are represented as species genetically isolated and evolve in parallel. Individuals of each species entering temporarily in collaboration with other members of other species and are rewarded based on the success of the collaboration, in other words, the fitness of the objective function.

In the work of Aguirre et al. al cite is aguirre presented a parallel model of cooperation / competition for genetic operators to develop self-adaptation within the AG's, showing that parallel self-adaptive mutations have better outcomes than serial self-adaptive mutations. The parallelism in this case, is restricted to the application of mutation operators.

In Bartz et. al [19] was developed a method of setting parameters for the output of the algorithm stochastically perturbed Particle Swarm Optimization (PSO) has been applied successfully in many applications. Preuss et. al [20] uses PSO to optimize self-adaptive evolutionary algorithms for binary code, where it is combined classical regression methods and statistical approaches.

A covariance matrix for self-adaptation in evolutionary strategies (CMSA-ES) is presented by Beyer and Sendhoff [21] as an example of successful self-adaptation of real numbers in the solution space. The correlation matrix of mutation sets the rotation axis and scale of change, allowing the adaptation of the characteristics of local fitness. Several operators are worked in the matrix as: mutation, crossover and selection.

3. Proposed algorithm

The main difference in the proposed algorithm presented in relation to the work of literature is the realization of the strategic parameters of the coevolution of MKP Algorithm, using the paradigm of parallel computing. The coevolutionary developed evolutionary process can be decomposed into two subprocesses: master process and slave process.

The slave process is a genetic algorithm that seeks to evolve individuals seeking to find a solution to a particular instance of an MKP. Was chosen to represent the binary representation of individuals. The slave process is a genetic

Table 1: Representation of individuals in the Master Process

Crossover Rate	Mutatio Rate	Population Size	Generations

algorithm that seeks to evolve individuals seeking to find a solution to a particular instance of an MKP. Was chosen to represent the binary representation of individuals. The master process has individuals with 4-position vector representation of real values, as shown in Table 1, with the following meanings respectively: number of generations, population size, crossover rate, mutation rate. The objective of this evolutionary process is to find individuals who return better fitness values when applied to an AG resolver an MKP (runs on master process).

Therefore, the basic purpose of the coevolutionary algorithm developed consists in the evolution of the parameters of the strategy due to changes in attributes of the data structure of each individual. Data structures correspond to individuals (candidate points in the search space). Figure 1 provides a simplified view of the AGP coevolutionary developed.
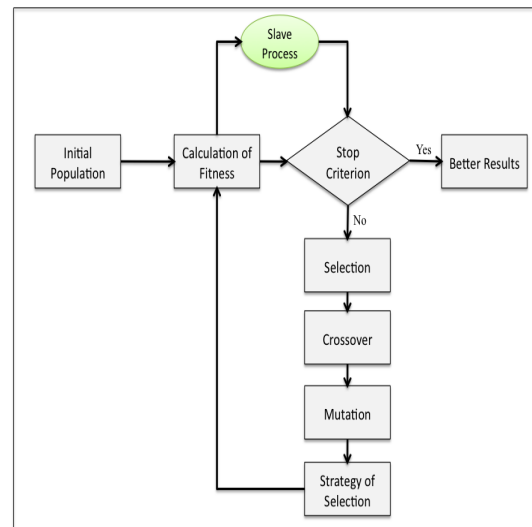
Fig. 1: Genetic Algorithm (AG) Flow of the master process.

In generating the initial population of the master process the following parameters were used: population corresponds to number of processors available slaves, 250 generations, crossover rate 0.85 and mutation rate 0.01. The evaluation is based on the objective function of the master process, which consists in the fitness returned by the slave process execution minus a penalty factor, as shown in Figure 2. Each slave process receives as input an individual master processor with parameters that will be used to generate its initial population, and after the evolution of the slave process, the master processor receives a fitness value, called fitness master, one for each of individuals. Therefore, the system can be seen as two evolutionary processes interconnected and fed back,

featuring a coevolutionary system.
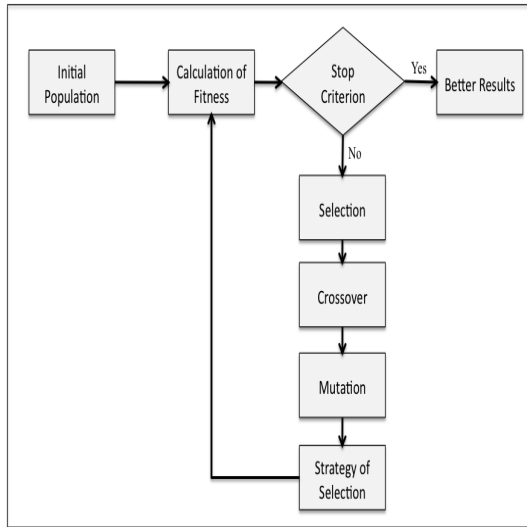
Fig. 2: Flow of the Genetic Algorithm MKP of the process slave.

The stopping criterion established for the AG of the slave process is the number of generations. The desired end result of coevolution is to find individuals with the best fitness values associated master.

Two experiments the composition of the objective function were used in the master processor to perform the coevolution. The first was based primarily on two components [22]: effectiveness and efficiency of processors to solve an MKP. The effectiveness is represented by the evolving capacities of a slave processor and translated by the slave processor performance evolution, which is the difference of fitness value in the first and last generation. This choice was used to replace the fitness value found by the slave processor, since a combination of strategic parameters with small populations and generations are able to find high values of fitness due to the randomness of the generation of initial population and not for their ability evolutionary. Since efficiency is represented here by the number of searches carried out in solution space, which can be simplified by the product of population size and the number of generations. Thus, one can say that a chromosome of the master processor has better performance than another, as it finds the best values of the objective function.

In another composition of the objective function of the master processor, the processor with the configuration parameters provided by the master processor, perform the evolution for the MKP and return to the master found the best fitness and the amount of search space solution that was used for this gym. With this information the master processor performs calculates the fitness of coevolution by applying a penalty on the fitness of the slave processors directly proportional to the number of searches performed.

The value chosen for the penalty for the tests was 20% of the best results found in the literature.

In relation to the master process, the crossover operator used was the arithmetic crossover [23], which is a strategy n-parent recombination, using the weighted average of the parents. Because it is a real representation of individuals, the mutation of genes is accomplished by choosing a random value for replacement of alleles, considering the probability of mutation.

The next session presents the results of the simulations.

4. Analysis of Results

The model proposed in this paper presents a coevolutionary genetic algorithm exploiting the parallelism available in the model of distributed memory parallel computing, implemented in language C and MPI .

The experiments were performed in a staggered platform for distributed memory (MPI). It explored the instances 1 and 2 of the OR-library [5]. The experiments run on a cluster of 7 machines, with the following settings: AMD Phenom 9600 Quad-Core, 4 Gb of RAM, available hard drive partition of 160 Gb .

Experiments were performed with two objective functions in the master processor. The first, according to the evolutionary capacity and efficiency, and the second considering the value of fitness with application of the penalty, based on the number of searches.

After the evolutionary process, two situations can be verified. The first, in which the parameters found in the final representation of the best individuals in the master process is achieved by the very nature of the convergence process. In the second possibility, there are cases in which the best individual has values in their genes (crossover rate, mutation rate, number of generations and population size) forged in the evolutionary process effective. This conclusion is taken from the moment that individuals evolved with a low amount of searches as simulated by 10 times, did not return good slave fitness, reinforcing the randomness as the main influence.

Thus, the analysis of results is performed disregarding the results of searches with a small amount (less than 25 %), and focusing on data that showed potentially interesting in this context, ie, search over 75 %. Simulations were performed with up to 20 slave processors, the master process generating random values between 0 and 200 for the parameters: population size and number of generations. This configuration allows up to 40.000 a search solution solutions space.

The analysis undertaken aimed at identifying the best settings of the input parameters for the resolution of an MKP through a genetic algorithm. For this, the outputs of AGP were subjected to a data mining (using the tool WEKA) seeking to identify characteristics that allow correlation of evolutionary computing specialists, a combination of the

most suitable configuration of the strategic parameters of the GA to solve an MKP.

Applying the mining algorithm selection of attributes, it was found that the parameters with greater influence and a strong correlation in the fitness master are: population size and mutation rate. The level of influence of these parameters can be seen even more clearly in the cluster analysis performed on clusters by applying the algorithm based on centroids, Kmeans on data obtained from simulations with the first objective function, as shown in Table 2.

Table 2: Kmeans Algorithm applied in the strategic parameters

Attributes	Cluster 0	Cluster 1	Cluster 2	Cluster 3
Elements	45	35	168	14
Generations	56.93	42.48	57.42	78.28
Population Size	17.86	26.28	44.52	16.21
Crossover Rate	0.2356	0.5911	0.4702	0.7636
Mutation Rate	0.3967	0.7794	0.4841	0.1857
Fitness Master	33222	27756	1033	34972

The analysis presents data in five groups with similar characteristics (clusters) corresponding to 23% 13% 25% 6% and 33% of the data. It is observed that the clusters 0 and 3, with the lowest values for mutation rates, however, the highest fitness, indicating an inverse correlation. The same is true for the population size as observed in the same clusters. These clusters are those with the best fitness values. The multiple correlation of the parameters was also analyzed showing that the population size influences the rate of mutation, whereas the reverse is not true.

In an analysis with only two clusters (Table 3), it clearly appears that the cluster 0 performs a greater number of searches in solution space (exploring party), generating a greater computational cost, while in cluster 1, is characterized by individuals who carry a smaller number of searches and fitness values are much higher than in cluster 0. This phenomenon can be explained by the randomness of the initial population that can find some guy with a good fitness value associated, but not necessarily the individuals in this cluster a good performance evolution.

Table 3: Analysis of strategic parameters with two clusters.

Attributes	Cluster 0	Cluster 1
Search	6864	889
Crossover Rate	0.434	0.477
Mutation Rate	0.519	0.4088
Fitness Master	52	32400

In numerical terms, it is concluded that a large population (number of slave processors) master process would be necessary to ensure an evolutionary process more accurately. The values close to 0.5 in both clusters for both crossover rate and mutation rate to show a lack of greater number of individuals in the population, especially in the mutation rate, these values indicate the need for high achievement of explotation in search of diversification of candidate solutions. In this case, the use of a larger number of processors, such as the GPU, seems an interesting alternative.

Considering the results with few (less than 10) search (population size versus number of generations), we applied the classification algorithm M5P to obtain the correlation coefficient and the relationship of inference to the value of fitness in terms of strategic parameters 4.

Table 4: Sorting results of the simulations

Number of Searches	Search Space (%)	Classification
Between 110 and 1055	Between 0.25% and 2.6%	High (86.87%)
Between 1055 and 7044	Between 2.6 % and 17%	High (63.30%)
Up to 7044	Up tp 17%	Low (38%)

To a high number of search (> 7444) the linear model does not return good ratings. The first track, one can observe that the independent term has a high value, noting that the minimum value that the master can take fitness is also high (around 13000). It appears that this is a good range to obtain high values of fitness, since the classification is also of considerable value (86%).

In classification using the algorithm REPTree (Table 2) with good correlation coefficient (0.84), the variable search is the factor that most influences the determination of fitness. The range between 115.5 and 639.5 is the only one that another factor (in this case, crossover) is relevant. Interestingly, the highest value for fitness was obtained for the highest values of crossover in the middle range of searches that match the range of the previous algorithm for best results. Another issue to consider is that both for this algorithm as for the other search values represented are well below average (about 5000) showing that a process where you can search many an algorithm is difficult to obtain good results. This was to be expected because of the randomness of genetic algorithms.

Seeking to penalize high fitness values in processors with low amount of searches, we used the objective function with penalty on the number of searches. Whereas the maximum amount possible is 40.000 search, we have assumed the penalty in the amount of 500 (12.5%) for the maximum amount. Thus, if a processor performs 2000 searches will be penalized 25 in the amount of your fitness.

In Table 5 shows the clustering applied in simulations using penalty function. Were used the instance 1 of [5], choosing only the simulations with fitness values greater than 20.500.

It is observed that there is a relationship already expected

Table 5: Clusters for coevolution with search fee.

Attributes	Cluster 0	Cluster 1	Cluster 2
Elements	5	13	11
Search	13932	2800	1648
Crossover Rate	0.394	0.7323	0.1464
Mutation Rate	0.618	0.5446	0.4755
Fitness Master	21261	21113	21370

in the second cluster occurs where the least amount of searches, the mutation rate is high, while the crossover rate is very low. In the first cluster, which has a greater amount of searches, the mutation rate is higher, however, the crossover rate also accompanies growth, signaling a behavior in these settings exploration and explotation. In the first cluster, which represents most of the data, we can see a scenario with a larger amount of searches and crossover rate, however, a smaller value for crossover rate, showing how much more searches to be lower the rate of mutation.

Simulations with the same amount of search present values for crossover and mutation as well different. This can be explained because in smaller populations, it is necessary to have higher values of mutation, while larger populations, the mutation rate tends to be lower. Within this context the data is separated into two: populations greater than 100 and less than 100, seeking to identify some patterns. Table 6 presents all clusters with high mutation rates and similar rates of crossover.

Table 6: Simulations with populations less than 100 individuals.

Attributes	Cluster 0	Cluster 1	Cluster 2	Cluster 3
Elements	3	7	9	10
Search	1566	2059	9173	1372
Crossover Rate	0.52	0.56	0.7	0.38
Mutation Rate	0.66	0.5	0.74	0.30
Fitness Master	19783	21175	20405	20402
Generations	159	77	105	45
Population	10	28	85	78

5. Conclusions

The objective of this study was to develop a Parallel Genetic Algorithm (PGA) for the coevolutionary strategic parameters, applied to the Multidimensional Knapsack Problem (MKP). The inherent parallel feature of Genetic Algorithm (GA), was chosen computing model for distributed memory implementation of coevolution through MPI.

Through analysis of the simulations was found groups of individuals of strategic parameters, reaching them with a good fitness values due to randomness, and the other group showing the ability of the evolutionary model developed. It

created an objective function, which gets the fitness of the master processor through the relationship of efficiency and effectiveness of the AG in the slave processors, and other objective function with penalty is assessed according to the number of searches.

The experiments have shown the potential of parallel computing models, especially MPI, indicating as future work the possibility of promising to test the model on GPU using CUDA implementation, concluding that larger populations tend to master the process to ensure a more evolutionary process accuracy.

References

[1] H. Schwefel, "Numerical optimization of computer models," 1981.

[2] V. H. A. Marques, "Adaptação de parâmetros em meta-heurísticas com sistemsa nebulosos genéticos," Master's thesis, Campinas University, 2011.

[3] E. J. Spinosa, "Adaptação dinâmica de parâmetros em computação evolucionária: o controle do tamanho da população em um sistema de programação genética," Ph.D. dissertation, MSc Dissertation. Federal University of Paraná. Computer Science Department. Curitiba, Brazil, 2002.

[4] A. Eiben, R. Hinterding, and Z. Michalewicz, "Parameter control in evolutionary algorithms," *Evolutionary Computation, IEEE Transactions on*, vol. 3, no. 2, pp. 124–141, 1999.

[5] J. . E. Beasley, "OR-LIBRARY - Collection of test data sets for a variety of operations research problems," 2010. [Online]. Available: http://people.brunel.ac.uk/ mastjjb/jeb/info.html

[6] I. Rechenberg, "Evolutionsstrategie-optimierung technischer systems nach prinzipien der biologischen evolution, stuttgart: Frommann-holzboog, 1973," 1981.

[7] D. Fogel, L. Fogel, and J. Atmar, "Meta-evolutionary programming," in *Signals, systems and computers, 1991. 1991 Conference record of the twenty-fifth asilomar conference on*. IEEE, 1991, pp. 540–545.

[8] J. Bagley, "The behavior of adaptive systems which employ genetic and correlation algorithms," 1967.

[9] T. Back, U. Hammel, and H. Schwefel, "Evolutionary computation: Comments on the history and current state," *Evolutionary computation, IEEE Transactions on*, vol. 1, no. 1, pp. 3–17, 1997.

[10] H. Beyer, "Toward a theory of evolution strategies: Self-adaptation," *Evolutionary Computation*, vol. 3, no. 3, pp. 311–347, 1995.

[11] N. Hansen and A. Ostermeier, "Convergence properties of evolution strategies with the derandomized covariance matrix adaptation: The (=," *Eufit*, vol. 97, pp. 650–654, 1997.

[12] N. Saravanan, D. Fogel, and K. Nelson, "A comparison of methods for self-adaptation in evolutionary algorithms," *BioSystems*, vol. 36, no. 2, pp. 157–166, 1995.

[13] H. Schwefel, *Collective phenomena in evolutionary systems*. Dekanat Informatik, Univ., 1987.

[14] T. Back, "Self-adaptation in genetic algorithms," in *Proceedings of the First European Conference on Artificial Life*. Massachusetts, USA, 1992, pp. 263–271.

[15] R. Smith, E. Smuda, U. S. N. Aeronautics, and S. Administration, "Adaptively resizing populations: Algorithm, analysis, and first results," *Complex Systems*, vol. 9, no. 1, pp. 47–72, 1995.

[16] R. Weinberg and M. Berkus, "Computer simulation of a living cell: Part i," *International Journal of Bio-Medical Computing*, vol. 2, no. 2, pp. 95–120, 1971.

[17] T. Koch, V. Scheer, J. Wakunda, and A. Zell, "A parallel, hybrid meta-optimization for finding better parameters of an evolution strategy in real world optimization problems," in *Proceedings of the 2000 Genetic and Evolutionary Computation Conference Workshop Program*, 2000, pp. 17–19.

[18] M. Potter, "The design and analysis of a computational model of cooperative coevolution," Ph.D. dissertation, Citeseer, 1997.

[19] T. Bartz-Beielstein, C. Lasarczyk, and M. Preuß, "Sequential parameter optimization," in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 1. IEEE, 2005, pp. 773–780.

[20] M. Preuss and T. Bartz-Beielstein, "Sequential parameter optimization applied to self-adaptation for binary-coded evolutionary algorithms," *Parameter Setting in Evolutionary Algorithms*, pp. 91–119, 2007.

[21] H. Beyer and B. Sendhoff, "Covariance matrix adaptation revisited– the cmsa evolution strategy–," *Parallel Problem Solving from Nature– PPSN X*, pp. 123–132, 2008.

[22] H. Barbosa *et al.*, "On adaptive operator probabilities in real coded genetic algorithms," in *Proceedings of the Workshop on Advances and Trends in Artificial Intelligence*, 2000.

[23] Z. Michalewicz and M. Schoenauer, "Evolutionary algorithms for constrained parameter optimization problems," *Evolutionary computation*, vol. 4, no. 1, pp. 1–32, 1996.

Genetic algorithms for management of taxi scheduling.

Lucas P. Rosa, Déborah M. Ferreira, Leonardo L. B. V. Cruciol, Li Weigang and Deng Xi Jun
TransLab, University of Brasília - UnB - Brazil

Abstract— *The increased amount of flights in airports has caused serious problems for passengers and airlines, and brings risks to the system of ground traffic. The sequencing of taxi management is responsible for coordinating the process of rolling the aircraft from gate to track, or conversely, depending on the procedure for takeoff or landing. In this paper, it presents an optimization solution of taxi aircraft at airports using genetic algorithms. For the simulation almost real data from the Congonhas airport in Sao Paulo was used. The simulations demonstrate the validity of the developed model to avoid conflicts and increase efficiency in a congested airport. The model presented shows an improvement of up to 15% in the total time by taxi from Congonhas airport, and enables its application to other airports with simply modification of new routes and flight plans.*

Keywords: Genetic, Algorithms, Management, Taxi , Scheduling

1. Introduction

Taxiing is the process of moving the aircraft with its own engine through a lane. This process can happen before the take-off, when the aircraft follows the path from the gate to the runway, also can happen after the landing, when the aircraft follows the inverse path[1].

The Ministry of Defense of Brazil[2] defined the taxiing as the self-propelled movement of an aircraft on the surface of an aerodrome, excluding the landing and the departure. In the case of helicopters, it includes the movement on the surface of an aerodrome with a low altitude and speed.

The delay in this taxiing process generates great discomforts for the passengers and the airlines companies. It is estimated that an aircraft takes about 10% to 30% of its traveling time with taxiing, it also spends almost 10% of it's fuel in the ground movement[3]. This implies in losses for the airline companies and a higher emission of pollutants, caused by the amount of time the engine is kept turned on while waiting.

Genetic algorithms are defined as a search technique based in the metaphor of the biological process of natural evolution. They are heuristics techniques of global optimization, which put it as opposite of gradient method that follow the derivative of a function. Genetic Algorithms are a way of solving problems by mimicking

the same processes Mother Nature uses. They use the same combination of selection, recombination and mutation to evolve a solution to a problem[4].

This paper presents the solution for the taxiing problem using genetic algorithms. This technique is used to optimize the path that each aircraft will cross, reducing the necessary time for the taxiing process, and at the same time, avoiding the occurrence of accidents in the lane, like the collision of two aircrafts.

This paper is organized in the following manner: in section 2 we present related researches. Section 3 presents the concepts of genetic algorithms. Section 4 presents the solution for this problem, using genetic algorithms. Section 5 presents the case study and the obtained results. Section 6 presents the final considerations.

2. Related Work

The bottleneck of the air traffic is the airport capacity, that is why many researchers always look for methods to improve the related services, for examples, landing, departure, gates allocation, etc. It is extremely important to find ways to optimize the taxing, not only improving the wait time, but, also improving the fuel consumption and the pollutants emission. In this section we present some of the related works in this area.

2.1 DMAN - Departure Manager

As previously mentioned, the use of resources in ground is of extreme importance. The amount of time that an aircraft takes to leave a position of boarding/landing should be optimized. The allocation of resources in the ground, from refueling services to the meal time of the crew are variables that influence in the aircraft flow on the ground, until they are ready for departure [9].

The departures should attend an optimal sequence,less time and less effort, that way the traffic can happen with an easy flow. The flight schedule, the traffic condition on arrival, the size of the aircraft, are examples of conditions that affect the departure sequence in the airports.

Because of the great quantity of variables it is extremely complicated to obtain decisions that proportionate the desired air traffic flow and, because of that, computing specialized systems are researched to support the process

of decision-making.

The systems that take care of the planning of aircraft departure operations from the airports, are named Departure Management Systems or DMAN. These systems must act in an integrated and coordinated way, specially with the Arrival Management or AMAN, for a better optimization of the aircraft flow. The AMAN system is already being develop by Atech [13].

2.2 Ant Colony Algorithm.

When the ants are looking for food, they initially walk randomly until they find something for their colony, in the way back, they release a pheromone that traces the route until the point that the food was found. When another ant find this route, it will probably follow this new route and if it finds another food, it will release the pheromone, reinforcing the previous route[5]

From this idea, the ant colony optimization algorithm has surged and using it is possible to find an optimization for the taxi process[5]. The steps for this algorithm are the following:

1) Create the ant colony , $i = (1, 2, ..., M)$. The maximum number of nodes that each ant can visit is E. Establish the ant taboo table, where each ant can register its path.
2) Start searching. The ants are placed in the initial node B and they start looking for a path.
3) For the ants $k = 0$ until $k = m$, if the ant X_k finds the end E, stop search and select ant $X_k + 1$. The ant chooses the next node j with probability P_{ij} , calculated by the following equation:

$$P_{ij} = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\alpha}{\sum [\tau_{ih}]^\alpha [\eta_{ih}]^\alpha} \qquad (1)$$

Where α and β are constants, such that $\alpha \geq 0$ and ≥ 0. The α is the inspiration factor, and it represents the relative importance of a patch, the β determines the pheromone factor, that represents the relative visibility. If the node j has already been to the node list, the ant must choose another node in the same way. If the ant cannot arrive its destination, then the ant must move to another node j and insert j into the node list.
4) Compute the size of the paths that the ants have crossed, the best solution will already be stored. For ants from $k = 0$ until $k = m$, the size of the path L_k is computed using the nodes list; For the path (S_i, S_j), the quantity of pheromones can be computed by the following equation:

$$\tau(t + n) = (1 - p) * \tau_{ij} + \Delta\tau_{ij}(t) \qquad (2)$$

5) Renew the pheromone. For the path $(S_i, S + j)$, according with the previous equation and the next equation,

update the pheromone value.

$$\Delta\tau_{ij}(t) = \sum \tau_{ij}^k(t) \qquad (3)$$

6) Stop. If $N \geq N_{max}$, display in the output the most optimized path and stop the search.

Using this algorithm, the aircrafts would be treated like ants and they would try to find the shortest path to their destinations.

3. Genetic Algorithms

Genetic algorithms are defined as a search technique based in the metaphor of the biological process of natural evolution. They are heuristics techniques of global optimization, which put it as opposite of gradient method that follow the derivative of a function[4]

It creates individuals populations and this are submitted to the genetic operators of selection, recombination and mutation. Therefore, is possible to perform a qualitative evaluation of the individuals as solution for the problem and generate a process of natural evolution with these individuals, which eventually will generate a descendant that represents a good solution. In another words, genetic algorithms are based in the mechanics of natural selection and genetics, combining the survival of the fittest and a structured form of genetic information exchange.

The search space of the Genetic Algorithm (GA) is the genome, which is constituted by genotypes. Because of the linear structure of these genotypes, they are called chromosomes. The chromosomes are used in the genetic algorithms and are represented as strings (with fixed or variable size) of one or more data types, for example bits and real number. The basic unit of information is the gene, it constitutes the chromosome, and it may be a bit or a char. The allele is the value of a specific gene.

The search of the GA copies the reproduction behavior of the nature. In the sexual reproduction the genotypes of the parents recombine itself and in the asexual reproduction only mutation occurs. For the GA reproduction, we start with an empty population. We select two parents of the original population, which are copied and suffer recombination and then the results will pass through mutation. Therefore, we obtain two children, which are added to the population of children. This process is repeated several times until the number of individuals in the population of children is complete.

The mutation is a very important method to preserve the diversity and variety of candidates to the solution, performing small changes in the chromosome. In strings chromosomes of fixed length, this can be achieved randomly

changing the value of a gene. The recombination is the biggest difference of the Genetic algorithms, it mix and combine parts of the parents to form a children, basically, occurs an exchange of parts between the two genotypes.

The fitness function is defined through a genetic representation and it assesses the quality of the presented solution. The fitness function depends on the concerned problem. It is used to allocate reproductive traits to the individuals in the population and, then, acts as a measure of quality that needs to be increased. It means that individuals with a higher value of fitness will have a bigger probability of being select as a candidate for further examination [7].

Genetic algorithm does not end when they find local maximum. Like natural evolution, they are not a directed process to find one optimal solution. The process consists of making a series of individuals compete between each other in the process of survival of the fittest, that is, the best survive and continue, but is possible that the next generation will be worse than the previous one. Figure 1 illustrates the steps of the genetic algorithm.
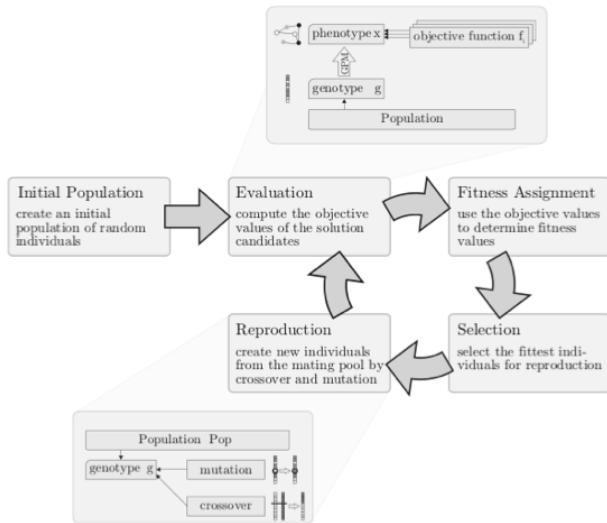
Fig. 1: Steps of the GA(Luke, 2009)

4. Proposal For Taxing Sequence Management

The system for management and sequencing of taxiing was developed using genetic algorithm inspired by [6]. The model is composed by three different modules, presented in Figure 2.

The information gathering module is responsible for storing and accessing all the data generated by flight controllers, like: nodes, routes, flight plans, beyond
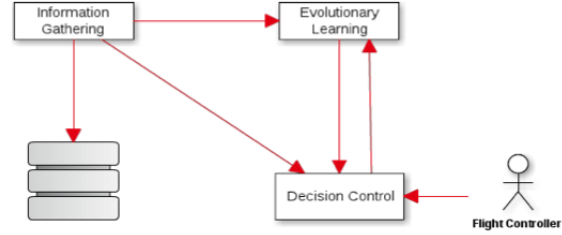
Fig. 2: System Modeling

essential system information, like: possible restrictions, size of the routes and speed on each node.

The prediction module using evolutionary learning is responsible by the use of genetics concepts and information acquired by the gathering of information to compute the best scenario for a moment t, which the gathered data represent.

The decision control module integrates all the parts of the system and presents to the flight controller all the necessary information to evaluate his decision about which flight plan should be used. Besides that, it will also present estimative of the spent time, fitness and cost percentages, and the path each aircraft will cross.

4.1 Genetic Algorithm Modeling

A) The routs will form the genes and each aircraft will be a locus, ilustrated in Figure 3, therefore, a aircraft being in a gate or coming from a runaway can have different values for routes. Each route will be represented by a number from 1 to number of possible routes. The sequence of the aircraft and of the initial routes is determined with help of the previous schedule, the first pair of chromosomes will already be according to the applied restrictions.

B) Fitness Function: When the conflict restrictions do not satisfy the model qualification, a function of penalty will be used. As a result, the objective function as punishment was chosen as the fitness function of the project. The fitness function is defined as:

$$
\begin{aligned}
ObjV = \sum_{k \in N_f} \{ \sum_{i \in (N \cup Q)} \sum_{j \in (N \cup Q)} x_{ijk} t_{ijk} \\
+ M_1 max(t_S - |y_{ik}t_{ik} - y_{ikp}t_{ikp}|, 0) \\
+ M_2 max\{ -\langle x_{ijk}t_{ik}v_k - x_{ijkp}t_{ikp}v_{kp} \rangle \\
\langle x_{ijk}t_{ik}v_k - x_{ijkp}t_{ikp}v_{kp} \rangle, 0 \} \\
+ M_3 \langle \langle x_{ijk}t_{jk} - x_{jikp}t_{jkp} \rangle \\
q + \langle x_{jikp}t_{ikp} - x_{ijk}t_{ik} \rangle 1 - q \rangle \} \\
\forall k_p \in F_k; F_k \subset N_f; q \in \langle 0,1 \rangle; \langle M_1, M_2, M_3 \rangle \in N^+
\end{aligned}
$$

$$(4)$$

C) Selection, Mutation and Crossover: The lowest value of fitness is chosen for this project. The chromosomes suffer crossover with a probability P_c.

D) The mutation must occur with a probability P_m to avoid the system to look for an optimal local solution. This operation chooses a random gene to be changed.

E) It will be adopted a maximum number of solution to stop the algorithm, this happens because of the difficulty of finding the best solution.

F) The algorithm is divided in:

1) Receive flight schedule, start and end nodes.

2) From the start and end point, it is generated a set of possible routes for each aircraft.

3) The initial population is created with the help of schedule, respecting all restrictions.

4) For this solution, we used genetics functions, in Java.
5) Find the best chromosome.

6) The gene that represents the best trajectory for the given set of aircraft is presented.
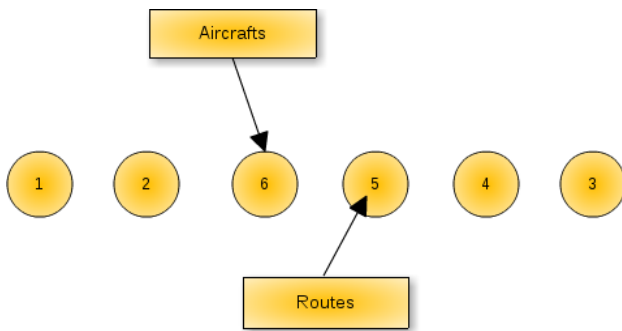
Fig. 3: Chromosome Representation

5. Study Case

Using the data obtained by Atech, we were able to construct a map (using Google Earth) with the taxiing routes and the distance between each node, presented in Figure 4.

Therefore, is possible to determine the approximate time spent by an aircraft in each node. As we can see in Figure 4, the latitude and longitude -23.625808531552053, - 46.66043209999998 represent the point G1 in the image.

Fig. 4: Route constructed with Google Earth

For this work, the Congonhas airport was used for study, first because it is one of the biggest airports on Brazil and second because the data from Congonhas was easier to obtain. It was created a database containing tables that represented 32 routs of the Congonhas airport, in São Paulo, containing the initial and final node of each route, the size of the route, the speed limit for a path(here we assume a speed of 10m/s in straight paths and 5m/s in curves. It was also added a table containing data from the flights: id from the flight, type of aircraft, start hour, if the aircraft is arriving or departing and the initial and final point of its route.

Without taking in consideration the different types of aircrafts, the minimum distance between aircrafts must be 50 meters, in order to avoid crossing conflicts, the minimum time in a crossing are 10 seconds, and the time to avoid conflicts are 600 seconds.

The following table shows the data from an hour of the Congonhas airport:

Table 1: Flight Data

Flight	Type	Prevision	Beginning	End	Arrival/Departure
03936	A320	15:00	G01	RA01	D
03120	A321	15:02	RA01	G16	A
03203	B737	15:05	RA01	G14	A
01339	B767	15:08	RA01	G10	A
03463	A319	15:11	RA01	G01	A
03933	A320	15:14	RA01	G07	A
03222	A320	15:15	G08	RA01	D
03107	B737	15:17	G03	RA01	D
03123	A321	15:19	G05	RB01	D
01353	A320	15:20	RA01	G02	A
01318	B737	15:21	G04	RB01	D
06009	A321	15:23	RA01	G09	A
03055	A320	15:24	G11	RA01	D

We assumed an initial population of $NIND = 100$, a difference between generations $GGAP = 0,9$, a crossover probability $P_c = 0,7$, a mutation probability $P_m = 0,01$ and a maximum number of generations $MAXGEN = 600$. The fitness function converged in generation 300.

6. Final Considerations

It was applied a genetic algorithms strategy to determine the best routes for a finite set of aircrafts in an airport, avoiding conflict between themselves during the taxiing.

The performed simulations demonstrated the validity of the model for avoid conflicts and improving the efficiency of a congested airport. The presented model shows an improvement of 15% in the total taxiing time in Congonhas airport, it is also possible to use this project in others airports.

For future work, we will study the possibility of using this algorithm in other airports and make improvements in this model.

References

[1] Waal, P.R. de; Smeltink, J.W.; Soomer, M.J. and van der Mei, R.D. (2003) Optimization of airport taxi planning. Technical report, National Aerospace Laboratory NLR.

[2] Ministry of Defense(2009) ICA 100-12. [Online]. Available: http://publicacoes.decea.gov.br/download.cfm?d=2558.

[3] Balakrishnan, H. and Deonandan, I. (2010) Evaluation of strategies for reducing taxi-out emissions at airports. In the AIAA Aviation Technology, Integration, and Operations (ATIO) Conference.

[4] Linden, R. (2006) Algoritmos Genéticos. Brasport.

[5] Changyou, L.; Yongcheng, W. (2011) "Aircraft Taxi Path Optimization Based on Ant Colony Algorithm," Computational Intelligence and Design (ISCID), 2011 Fourth International Symposium on vol.2.

[6] Changyou, L; Kaifeng, G. (2010) "Airport Taxi Scheduling Optimization Based on Genetic Algorithm," Computational Intelligence and Security (CIS).

[7] Goldberg, E. D(1989). Genetic Algorithms in Search, Optimization, and Machine Learning.

[8] Luke, S. Essentials of Metaheuristics. Lulu, 2009. Available: http://cs.gmu.edu/ sean/book/metaheuristics/.

[9] DCEA, Departamento de Controle de Espaco Aéreo. (2009) Regras do ar e servícos de tráfego aéreo. Technical report. Ministério da Defesa.

[10] Doganis, R. (2006) The airline Business. Routledge, Taylor and Francis Group, London.

[11] Bastos, A. (2008) Impacts of flight delays on airline pricing: Evidence from the us market. Review of Transportation Literature, Vol. 2, No. 1, pp. 39-60.

[12] Matias, J. A.; Macário, M. and Reis, V. (2011) Proposals for mitigation of pollutant emissions from aircraft: application to lisbon airport. Transportes Journal vol: 19, issue: 2:34-41.

[13] Atech (2011) Detalhamento de projeto SIGEPA. Technical report, TransLab - UnB/Atech

Air Traffic Flow Reduction Using Genetic Algorithms in Brazilian Airspace

Leonardo L. B. V. Cruciol and Li Weigang
Department of Computer Science, University of Brasilia, Brasilia, DF, Brazil

Abstract—*Air Traffic Flow Management (ATFM) is a process to treat an online problem, which is related to many complex attributes. The air traffic controllers are responsible to handle data and acquire knowledge from currently airspace scenario to detect possible risks situations and take some actions to reduce air traffic flow congestions. Recently, Reinforcement Learning (RL) and Multiagent System (MAS) were used to solve this kind of problem. As an alternative, Genetic Algorithms (GA) technique was also used to select the best group of actions, using an evolutionary approach when happens new situations. This paper presents a case study applied in Brazilian Airspace with better suggestions of restrictive measures. The main goal is reduce the impact of the action on airspace scenarios using genetic algorithms to choose the actions for specific scenario. As initial studies, Brasilia International Airport (SBBR) and Congonhas Airport (SBSP) in São Paulo, are involved. The developed method improved the ATFM with the better performance of 8% to 21% with the implementation of Decision Support System (DSS).*

Keywords: Genetic Algorithms, Air Holding Problem, Air Traffic Flow Management, Real Time Problem

1. Introduction

Air Traffic Flow Management (ATFM) is a real time problem, which is related to many complex attributes. The air traffic controllers are responsible to handle data and acquire knowledge about currently airspace scenario to detect possible risks situations and take some actions to reduce air traffic flow. However, these specialists need to verify and decide in a short time interval, making harder to take the best solutions.

In ATFM domain there is the Air Holding Problem (AHP), which occurs when aircraft in flight route needs to wait in airspace for some reasons, such as closed airport or natural phenomenon. It is a complex real time problem that handles with aircraft in departure and arrival situations. Thereby even with a planning of departure and arrival for one day of an airport, if an unexpected situation happens in other airport, it will probably impact in this one. This impact can be reduced during flight by reducing or increasing the time between air control points. Thus, air traffic controllers get to apply some decisions on aircraft and arranging aircraft schedule as planned.

In last years, it was researched manners to improve air traffic management and support decision process of air traffic controllers, using artificial intelligence concepts to achieve better results. The main researches used Reinforcement Learning (RL) and Multiagent System (MAS) [2,3]. The RL is responsible to make possible the system learns by the time with acquiring knowledge from decisions of air traffic controllers. The MAS was used to model the agents' architecture to handle with specific airspace positions, thus it was constructed more specialist agents for each situation.

It was chosen the Genetic Algorithms (GA) technique to select the best group of actions, using an evolutionary approach when happens new situations. This approach can be used in new situations, i.e., the Decision Support System (DSS) never handle it, so the RL approach did not have enough knowledge to achieve great results in majority situations. Thus, it is achieved a new step to ATFM decision support aiming a full DSS, which it will have capability to self-adaptation in real air traffic flow situations.

This paper will present a case study applied in Brazilian Airspace to reduce the air traffic flow by better suggestions of restrictive measures. The main goal is to reduce the action impact on airspace scenarios using genetic algorithms to choose the best group of actions for a specific scenario. It was chosen flights between two important airports of Brazil, Brasilia International Airport (SBBR) in Brasilia and Congonhas Airport (SBSP) in São Paulo.

The paper is organized in the following manner. In section 2, there is a brief review of related researchers and concepts of Genetic Algorithms and Air Traffic Flow Management in Brazil. Section 3 presents the proposed application of GA in ATFM domain. Section 4 presents results achieved. Section 5 concludes the paper and proposes the direction of the future researches.

2. Related Concepts and Researches

Recent researches in ATFM combine Reinforcement Learning and Multiagent techniques. It uses this combination to incorporate the human agent's experience into automated ATM processes.

It developed metrics to evaluate the efficiency of traffic flow control measures generated by Reinforcement Learning agents. As a result, it was possible to improve the actions taken by an agent conceived to simulate the behavior of

air traffic controllers, given the suggestions by the ATFM agent and as such suggestions were accepted by the air traffic controller [2,3,4,5].

In the proposed system [2], the ATFM agent uses a Reinforcement Learning algorithm that seeks to maximize the reward that derives from the scenarios generated, which will be evaluated based on the total delays generated and amount of aircraft in the ATC sectors, both resulted from the flow control policy. The air traffic control is characterized as a reward maximizing Reinforcement Learning agent that seeks only the minimization of the amount of aircraft in the ATC sectors. This assumed behavior for the air traffic controller is not entirely valid, as the minimization of traffic in the ATC sectors is not the only ATC goal [3,5].

In reality, ATC seeks a balance between safety, which benefits from a low number of aircraft in the ATC sectors; and operational efficiency, which depends on the maximization of flows. This assumed behavior is, therefore, a significant limitation of the model.

The next subsections present the concepts of Genetic Algorithms and Air Traffic Flow Management used in this research.

2.1 Genetic Algorithms

The Genetic Algorithms (GA) uses concepts of natural selection based on human evolution. This concept can be used to solve problems related to optimization by an easy manner. GA is a complex and robust technique to achieve great results in this kind of problem.

Its work can be defined in some steps such as initialize population, evaluation, selection, crossover, mutation, reproduction and final population. First of all, it is generated a population of possible solutions to the problem. So, this population is evaluated to choose the best individuals to be reproduced for the next generation.

It generated a crossover and/or mutation between the individuals to generate new others. Thus, it is generated a new population. This process will be executed until occurs the maximum amount of generations defined. The Fig. 1 presents the Genetic Algorithms process.

2.2 Air Traffic Flow Management

ATFM is a complex procedure to avoid exceeding air traffic capacity and focuses on the supply of information to maintain the traffic flow with safety and less impact on scenarios that are necessary to take unexpected measures. The ATFM environment can be organized into three phases:

- Strategic Level: Considering tactical planning of flights and covering the period of forty-eight hours until the time before the flight.
- Operational Level: Focusing on strategic decision making and covering the period from forty-eight to two hours before the flight.
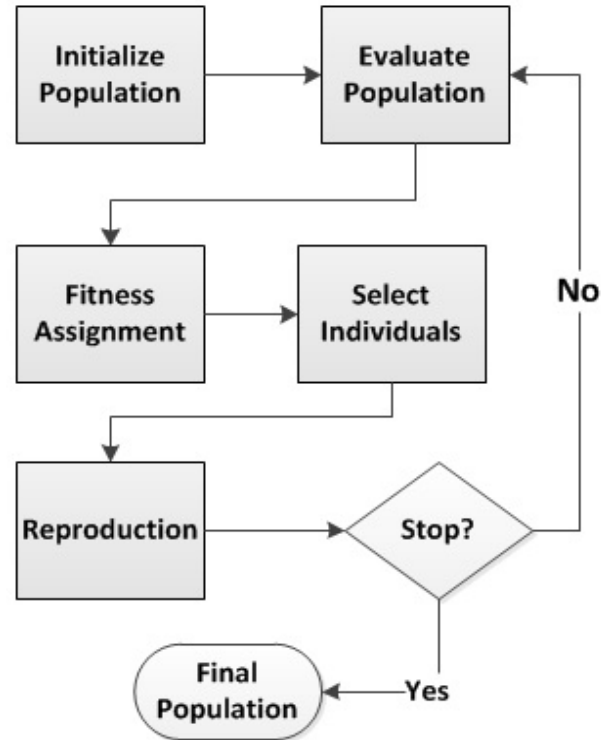
Fig. 1
GENETIC ALGORITHMS PROCESS

- Tactical Level: Considering tactical decision making and covering the period from two hours before the flight until the aircraft arrives at its destination.

This study concerns in ATFM Tactical Level. The main problem is related to aircraft in flight, which increases the problem severity because the occurrence of the problem and its solution happens on real time environment.

ATFM can guarantee that flights are conducted in a safe, quick, orderly and economic way. It is possible to avoid overloading in the air traffic capacity, optimize airspace and provide information to responsible authority.

2.2.1 Air Holding Problem

The air traffic flow management needs to be prepared to maximize the opportunities of the available structure, so that resources can be put into good condition of use. In this case, it can be mentioned the best flow management of the aircraft while in airspace and, therefore, besides improving the safety level of the Brazilian aviation. It is also possible to reduce negative indicators and achieve great results for all involved in a situation [5,8].

ATFM environment is a domain with many complex activities, such as the air holding problem. It occurs when aircraft in flight route needs to wait in airspace for some

reasons. One of these reasons could be that an airport had to be closed, for example. Situations involving natural phenomenon or terrorist acts may be motive to excess the safety amount of aircraft on a particular sector. Thus, other aircraft need to be retained for some time in other sectors.

The problem becomes much more serious, in some critical scenarios, e.g., when there is a scenario under responsibility of an Integrated Center of Air Defense and Air Traffic Control (CINDACTA), and specific local scenarios under responsibility of another one. When an action is chosen for a sub scenario, as the time goes by, the result of these actions can make better a sub scenario and aggravate another one, and then the whole situation can become an issue of great risk to the safety of everyone involved.

2.2.2 Brazilian Scenario

Brazilian airspace covers the entire land area of the country, including part of the Atlantic Ocean. There are five Flight Information Region (FIR): FIR - Amazônica, FIR - Recife, FIR - Brasilia, FIR - Curitiba e FIR - Atlântico. In this paper, it will be considered only FIR-Brasilia (FIR-BS), where are located both two airports. The airspace of FIR - Brasilia used in this case study is presented in Fig. 2.
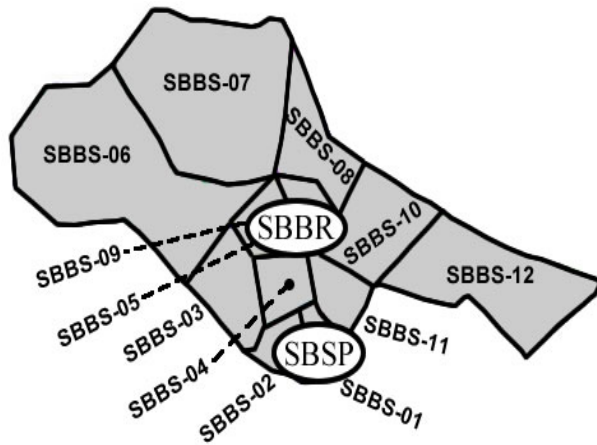
Fig. 2
AIRSPACE OF FIR - BRASILIA

Each FIR is divided into some control sectors, in order to separate management activities and improving control. Currently, there are 12 sectors in FIR-Brasilia. These sectors are carried out by air traffic controllers, which work in Area Control Center (ACC) that is responsible for the FIR. Thus, it is possible to realize the complexity of management activities and necessity for subdivision to manage so many factors, e.g., the amount of aircraft by airspace sector influences the complexity of managing, i.e., more aircraft flying in the same sector, more and greater safety risks involved in ATFM.

3. Genetic Algorithms Application in ATFM

The Genetic Algorithms allows to handle and simplify problems of optimization using the concepts of evolutionary theory, e.g., only the best group of actions will exist after a determinate generation from initial flight planning. Taking into the GA model, it was defined attributes to construct the parameters of this application such as initial population, evaluation methods, fitness assignment, stopping conditions, and others.

The initial population was defined from real data of departures and arrivals from two airports: SBBR (Brasilia) and SBSP (São Paulo). SBBR is the capital airport and SBSP is the main business airport from Brazil. Moreover, these airports are responsible for many connections flights which increase the impact of a decision in other airspace scenarios.

Thus, it was possible to choose a time interval with large amount of flights to research the critical scenarios, i.e., suggest actions for scenarios which there are more aircraft and the air traffic controller needs to verify more complex scenarios data.

It was chosen the time interval between 5pm and 6:59pm, because there were more arrivals and departures flights. During this time, there were 96 flights from or for these airports. The Table 1 presents a piece of data chosen for this experiment.

Table 1
FLIGHTS OF SBBR AND SBSP

$Aircraft$	$ICAO_D$	$Time_D$	$ICAO_A$	$Time_A$
A1	SBCG	14:23	SBSP	17:08
A2	SBRJ	15:43	SBBR	17:19
A3	SBGL	15:38	SBBR	17:24
A4	SBRJ	16:10	SBSP	17:05
A5	SBRJ	16:15	SBSP	17:11
A6	SBRJ	16:40	SBSP	17:24
A7	SBFL	16:40	SBSP	17:31
A8	SBCF	16:43	SBSP	17:59
A9	SBRJ	16:45	SBSP	17:34
A10	SBFL	16:55	SBSP	17:56
A11	SBSP	17:07	SBCT	18:10
A12	SBRJ	17:10	SBSP	17:53
A13	SBSP	17:10	SBRJ	18:08
A14	SBBR	17:20	SBSP	18:52
A15	SBSP	17:21	SBPA	18:58
A16	SBSP	17:23	SBCF	18:43
A17	SBSP	17:25	SBPA	19:02
A18	SBSP	17:29	SBRJ	18:32
A19	SBSP	17:40	SBRJ	18:38
A20	SBBR	17:40	SBPA	20:18

A21	SBSP	17:42	SBCF	18:53
A22	SBBR	17:45	SBGL	19:26
A23	SBBR	17:54	SBSP	19:28
A24	SBSP	17:55	SBFL	19:10
A25	SBSP	17:58	SBFL	19:10
A26	SBSP	18:00	SBRJ	19:04
A27	SBBR	18:00	SBGR	19:40

Understanding Table 1, there are the International Civil Aviation Organization (ICAO) code that identifies every airport in the whole world, departure time from departure airport, destination airport and its arrival time. Between departure and arrival airports, there are defined routes that aircraft must follow. In these routes, there are many control points which aircraft must passing in a defined time.

When some restrictive measure is applied, the defined time to check in a control point or landing can be modified. So, arrival time probably will be changed and this will generate an impact in other aircraft or/and other airport. This impact can be minimal or not. It is very difficult to know exactly the real impact, but it is possible to predict based on the first planning and the experience acquired by the system.

For each individual, i.e., flight aircraft, was defined 11 genes. Each gene, i.e., restrictive measures, represents a delay or forward which the gene in the middle is the planning flight. The genes between 1 and 5 are the minutes of forward for each aircraft and the genes between 6 and 11 are the minutes of delay. For each interval was considered 3 minutes of difference from previous gene. Fig. 3 presents the chromosome structure. The ideal population is when the aircraft is set in gene 0, so there is not delay and the flow is following as planned.
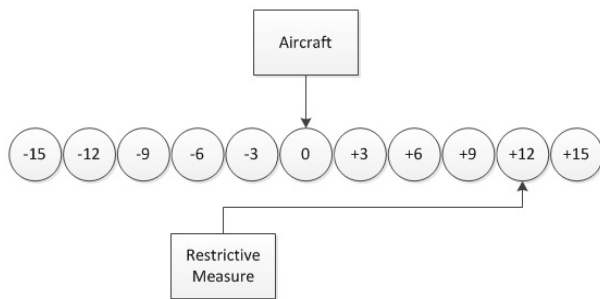
Fig. 3
CHROMOSOME STRUCTURE

The initial population was defined as the airspace scenario at 5pm. The best final population is achieved by minimal delay or forward of each individual, i.e., the group of analyzed aircraft achieved the minimal deviations of initial planning, e.g., Chromosome A (0000-3000000) means a forward of 3 minutes for aircraft 5.

The selection criterion to improve the next generation is defined by a fitness function that uses a rank selection, which will choose the best individuals for the next generation. Fitness function is defined as:

$$F = \sum_{i=1}^{n}(1 - |T|) - P_1 - P_2 \tag{1}$$

Where:

- T is a module time of delay or forward for each aircraft.
- P_1 is a penalty value equals 50% of T, if the restrictive measure is delay.
- P_2 is a penalty value equals 50% of T, if the restrictive measure is forward.

The recombination technique chosen was the one point and its probability was defined as 0.3. The maximum amount of generations was defined as 350.

The algorithm defined for this experiment can be summarized as below:

1) Receive the flight planning and deviations verified in the airspace;
2) It is generated three possible restrictive measures for each flight;
3) It is generated the initial population and evaluated by the genetic algorithm;
4) After end generation, it is found it the best chromosome.

4. Results

Based on the final population of this experiment, it was compared with the final results that air traffic controllers achieved in the real environment. This approach achieved about 8% and 21% of improvement in air traffic flow. Fig. 4 presents how the generation behaved in previous tests.
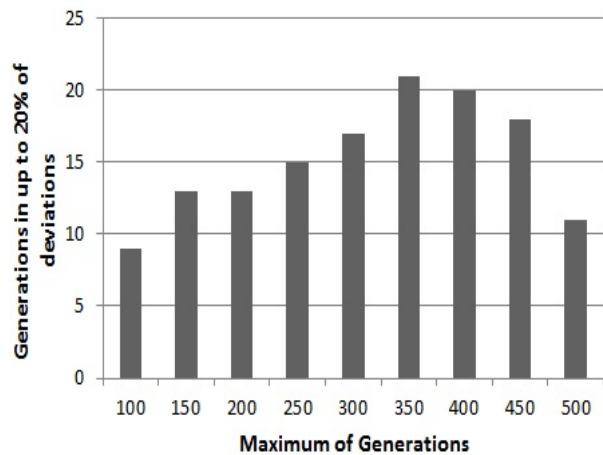
Fig. 4
GENERATION BEHAVIOR IN PREVIOUS TESTS

To evaluate the results of generation, the amount of generations was defined in an interval of 20%, which covers the summation of deviations from the planning time. There were generations with better results with less maximum of generations but these were particular events.

In average, better results were around 350 generations. Fig. 5 presents the behaviors of these generations.
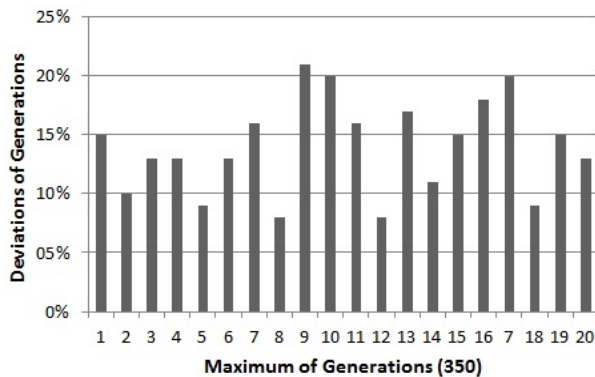
Fig. 5

BEHAVIOR WITH MAXIMUM OF 350 GENERATIONS

To maximum of generations set as 350, it was achieved between 8% and 21% of improvement. However, taking into all tests the average was about 14% of improvement of air traffic flow, which is less than maximum result but it is a great level too.

5. Conclusions

To improve the decision support process of air traffic controllers is need to develop models using specific combinations of some techniques, also known as hybrid system or model. It is very hard task to handle with all related problems using only one. Thus, it is important new researches to achieve results by system as a high specialist group.

Previous researches used Reinforcement Learning and Multiagent System to handle with air holding problem. A specific problem in ATFM domain which is harder to solve because there are many complex attributes besides it happens in real time environment. This point makes a great difference when it is developed a solution model because the model needs to be generic enough to self-adaptation for new unknown situations.

The Genetic Algorithms appears as a great tool to improve some models. In this case, it was used to suggest solutions for new scenarios, i.e., when the system does not have previous knowledge to decide the best restrictive measures to air traffic flow. It was chosen flights between two important airports of Brazil, Brasilia International Airport (SBBR) and Congonhas Airport (SBSP) in São Paulo. In initial studies,

it was achieved an improvement in air traffic flow between 8% and 21% for these situations.

The future research is to introduce new approaches to incorporate in a DSS system and develop more strategies to acquire knowledge and provide information to air traffic controller. It will be extend to other airports from Brazil and verify better techniques to improve a hybrid model using Reinforcement Learning, Genetic Algorithms and Multiagent System, especially to air holding problem.

References

[1] L. D. Davis, "Handbook of genetic algorithms", *Van Nostrand Reinhold*, 1991.

[2] A. Agogino and K. Tumer, "Learning Indirect Actions in Complex Domains: Action Suggestions for Air Traffic Control Advances in Complex Systems", *Advances in Complex Systems*, Vol. 12, pp. 493-512, 2009.

[3] L. L. B. V. Cruciol and L. Weigang, "Air Holding Problem Solving by Reinforcement Learning to Reduce the Congestion in Airspace Sectors", *The 2012 International Conference on Artificial Intelligence (ICAI'12)*, Vol. 1, pp. 272-278, 2012.

[4] G. Weiss, "Multiagent Systems, Artificial Societies and Simulated Organizations", *The MIT Press*, 2005.

[5] H. Balakrishnan and I. Deonandan, "Evaluation of strategies for reducing taxi-out emissions at airports", *The AIAA Aviation Technology, Integration, and Operations Conference*, 2010.

[6] D. E. Goldberg, "Genetic algorithms in search, optimization and machine learning", *Addison-Wesley*, 1989.

[7] X. Zhang, Y. Zhou, B. Liu and Z. Wang, "The Air Traffic Flow Management with Dynamic Capacity and Co-evolutionary Genetic Algorithm", *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference*, Vol. 1, pp. 580-585, 2007.

[8] D. Bertsimas, G. Lulli and A. Odoni, "An integer optimization approach to large-scale air traffic flow management", *Operations Research*, Vol. 59, No. 1, pp. 211-227, 2011.

[9] S. Caprì and M. Ignaccolo, "Genetic algorithms for solving the aircraft-sequencing problem: the introduction of departures into the dynamic model", *Journal of Air Transport Management*, Vol. 10, No. 5, pp. 345-351, 2004.

Reconfiguration of Distribution Systems to Reduce Voltage Sags by Using Genetic Algorithms

R. Tapia-Juárez, E. Espinosa-Juárez, M. Graff
Faculty of Electrical Engineering, Universidad Michoacana de San Nicolás de Hidalgo
Morelia, México

Abstract - *This paper describes the development of genetic algorithms for their application to the reconfiguration problem of radial distribution electrical networks, taking into account the expected occurrence of voltage sags. The reconfiguration is an optimization problem, which is formulated to obtain a radial topology representing the least voltage sags in certain buses. The electric system is represented as a graph, and the population can be generated by providing the mesh information of the system. Case studies are presented for the IEEE-69 buses test system, demonstrating the effectiveness of the implemented methodology.*

Keywords: Reconfiguration; genetic algorithms; power quality; voltage sags; electrical networks.

1 Introduction

With further technological development and the increasing energy demand, supply companies must provide energy to the quality that customers require. Furthermore, within the scope of smart grids, an important goal is to design electrical systems integrating information technology systems, control, and communication technologies in order to optimize the grid [1][2][3]. The term smart grid refers to the way in which the electrical system can operate using communication technology, power electronics, and data storage to monitor and have control over an entire power system. [4].

In this context, the power quality becomes more important, being a problem of great importance and interest, due to the integration of instrumentation devices and sensitive loads, this sensible equipment is becoming more common in industries. It demands a reliable power supply with a good quality. Problems related to power quality may impact the output of an industrial process resulting in significant economic losses.

Voltage sags are one of the most frequent phenomenon of power quality, because of this is important to characterize this phenomenon and also the urge of new strategies that can avoid them or mitigate the effects related to this problem. A voltage sag is a sudden reduction in the rms voltage value between 90% and 10% of the rated voltage, and it last between 0.5 and 1 minute cycle [5].

Several methodologies have been proposed to tackle the problem of voltage sags in the electrical system. Among the stochastic methods one can find, for example, the fault location method [5], critical distances method [5][6], as well as, the analytical methods for stochastic estimation [7]. There are also methods of state estimation of voltage sags, these use electrical network data, and measured data, in order to predict the voltage sags at buses where there is no metering [8][9][10].

On the other hand, authors in [11] and [12] were some of the pioneers on using a reconfiguration strategy in order to reduce power losses. More recently, other authors have used different optimization techniques, both traditional and non-traditional, such as: simulated annealing and genetic algorithms (GA) to solve these reconfiguration problems [13]-[18].

In this paper, it is proposed to solve the reconfiguration of radial distribution systems using a GA. The proposal uses a novel set of genetic operators that allows to only generate individuals that satisfy the radiality constraint. By doing so, it is effectively reducing the search space. In order to test the proposal, the performance of the developed algorithm is applied to the IEEE-69 buses distribution test system.

2 Formulation of the reconfiguration problem

In this paper, the purpose of electrical system reconfiguration is to find the state (on/off) of existent switches leading to a topology that minimizes the number of voltage sags with respect to specified values of voltage sags at system buses. This specified voltages sags are set based on statistical data or according to standard values of voltage sags on these buses.

In order to formulate the reconfiguration problem, we need to define the next vectors,

$$V_{ref} = \begin{bmatrix} v_{ref1} & v_{ref2} & v_{ref3} & \cdots & v_{refn} \end{bmatrix}$$

$$V_{reconf} = \begin{bmatrix} v_{reconf\,1} & v_{reconf\,2} & v_{reconf\,3} & \cdots & v_{reconf\ldots n} \end{bmatrix}$$

$$V_B = \begin{bmatrix} v_{B1} & v_{B2} & v_{B3} & \cdots & v_{B\ldots n} \end{bmatrix}$$

where V_{ref}, is the vector of reference values of voltage sags at buses, V_{reconf} is the vector of voltage sags at buses after reconfiguration, V_B is a binary vector where v_{Bi} is 1 if $v_{reconfi} > v_{ref\,i}$ and v_{Bi} is 0 otherwise, and n is the number of system buses.

According to this, the optimization problem is formulated as:

$$\min \sum_{i=1}^{n} v_{Bi} \tag{1}$$

Subject to:

There are not isolated parts in the system (2)

$$M = n - n_f \tag{3}$$

where:

n_f Number of feeders or sources

M Number of radial net branch.

In this paper the number of voltage sags is estimated by means of the fault position method [5][19]. The radiality constraint is partially considered by (3), an as consequence cannot assure the radiality in the system.

Usually, the way to represent the connection/disconnection of a transmission line is by means of a single switch operation. According to this, a vector Y is defined which consists of y_i binary elements that can change their state. The number of elements of Y is equal to the considered number of switches (SN) that can change their state.

For a generic system of N buses, where it is agreed that SN switches can operate (open/close) without leaving any system areas de-energized, it could be possible to represent a configuration as

$$Y_1 = \begin{matrix} l_1 & l_2 & l_3 \cdots & l_{SN} \\ \begin{bmatrix} 1 & 0 & 1 & \cdots & \cdot \end{bmatrix} \end{matrix} \tag{4}$$

where it can be observed that switches of lines l_1 and l_3 are closed, and the switch of line l_2 is open. In this case, this means that line l_2 is not energized.

For sake of simplicity a vector S contains only the number of disconnected lines. Then for the vector in (4) the corresponding vector S is

$$S = [1\ 3\ \ldots]$$

3 Implementation of the proposed methodology by using genetic algorithms

The problem being solved is an optimization problem with constraint. Generally, this problem is address by introducing by penalizing the infeasible solutions (see for example [20]); however, in this contribution we decided to follow a more direct approach. Instead, of penalizing unfeasible individuals, we decided to modify the genetic operations (i.e., mutation and crossover) to always produce feasible individuals. In addition to this, we propose a method to generate only feasible individuals in the initial population.

In order to explain the proposed methodology, the basic distribution system in Fig. 1 is used. The data of the system depicted in the figure is provided by [11].
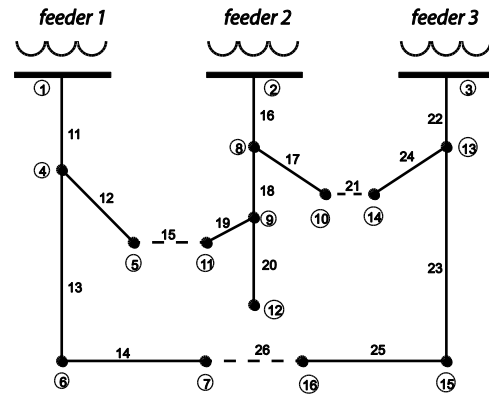
Fig. 1. Distribution system

3.1 Basic loops

The feeders are integrated into a single bus with the purpose of a cycle is not formed between them (see Fig. 2). After a successful reconfiguration feeders return to their original position, so it can be observed how each feeder supplied to each corresponding load.

For a generic system is possible to find a number PL of basic loops that is defined by

$$PL = n_l - n - 1 \tag{5}$$

where n_l is the total number of system lines

Note that (5) also indicates the number of lines that must be out of operation in order to assure that the system has a radial configuration. In Fig. 2 the basic loops are shown and also the tie-lines which are normally open (looping branches).
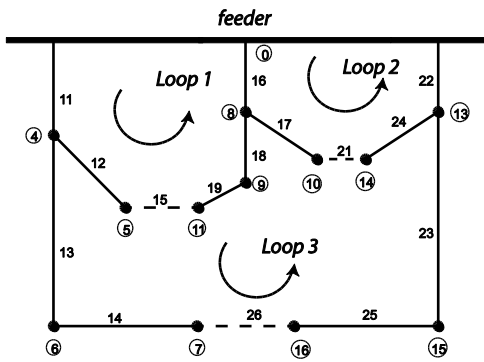
Fig. 2.   Basic distribution system

It is noteworthy that in a distribution system to change the operating state of a tie-line normally open to close, it must change the state of a line belonging to the corresponding basic loops, to normally open, in order to maintain the radiality of the system.

Then for the system in Fig. 2 the basic loops are:

Loop 1 = [11 12 16 18 19 15]

Loop 2 = [16 17 22 24 21]

Loop 3 = [11 13 14 22 23 25 26]

Then, for connection shown in Fig. 2 the corresponding vector *S* is,

$$S = [15\ 21\ 26]$$

Note that line 20 is not part of any basic loop and its status change only cause that bus 12 is without energy or disconnected. It is for this reason that in Figure 2 is not taken into account, but when the reconfiguration is performed the line 20 is incorporated into the generated topology.

3.2   Chromosome encoding

The reconfiguration encoding has been extensively studied in [16] the most common encodings are mentioned. The encoding proposal in this paper is a combination of the proposals of [16] and [21].

For the example, in Fig. 1 the vector $S = [15\ 21\ 26]$ is encoded as

$$S = [1\ 1\ 1\ 1\ 0\ \ 1\ 0\ 1\ 0\ 1\ \ 0\ 1\ 0\ 1\ 1]$$

It can be observed that the least significant bit is the first line for each line.

3.3   Initial population

There exists several algorithms to generate radial networks. In [22] the authors generated radial topologies implementing Prim's algorithm and Kruskal. In this paper the proposed method consists of the following steps:

1) Generate a random vector for loop selection.

2) For each basic loop a line is chosen randomly.

3) The line is deleted from the following sets of basic loops with the main purpose of which cannot be selected.

4) Repeat steps 2 and 3 for all remaining loops.

5) When the solution vector S is full, then its radiality is checked.

The process is repeated if the vector does not fulfill the radiality condition. This will generate radial topologies, which always respect the positions of the loop in the vector *S*.

3.4   Crossover operation

The crossover operation used is a uniform crossover; we implement this by using a binary mask. That is, if the i-th element of the mask is 1, the gene located in that position of parent 1 passes to the descendant, similarly if the element is 0 the gene is copied from parent 2 to the descendant in its corresponding position. This is shown in Fig. 3.
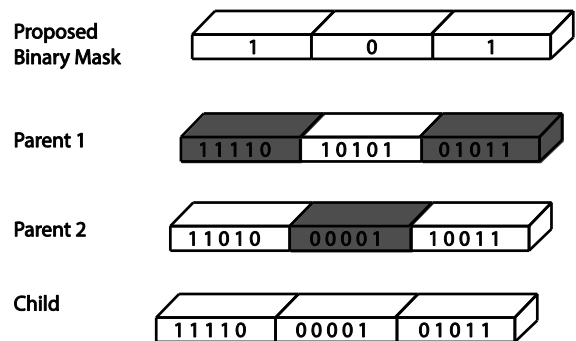
Fig. 3.   Crosses of individuals

This type of crossover ensures that most of transfers of genetic material into new descendants. After performing the crossover the new individual is verified to check the radiality of the system. In the case, this new individual does not represent a radiality system then a new operator is implemented as follow.

Auxiliary operator of crossover: A random binary mask is generated, which complies with the same criteria described above. Fig. 4 shows the crossing described above.
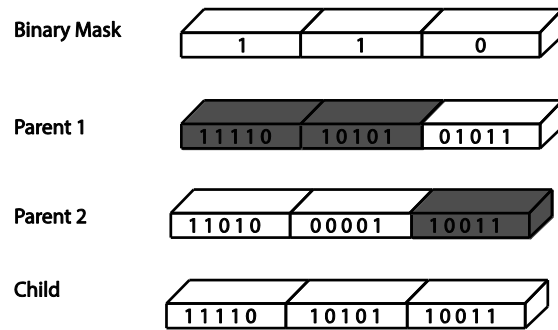
Fig. 4.   Auxiliary crossing operator

3.5   Mutation operator

To perform this operation, it is randomly selected a normally open switch to change its status. This is described in the following steps:

1) Select a line of the chromosome randomly.

2) Identify the basic loop to which it belongs and remove the assembly line.

3) Select a random line of the loop assembly and replacing the main line into the chromosome.

4) Check the system radiality.

If the system does not meet the radiality condition is repeating the process, in this way the mutation operator ensures that the algorithm does not stagnate in a local minimum, and to perform a search in a larger space of solutions.

To illustrate this process, line three was randomly selected chromosome at line 26 which is removed from the main loop three:

Loop 3 = [11 13 14 22 23 25 ~~26~~]

To replace this line, a line of the set of the main loop 3 is randomly selected the line 23,

Loop 3 = [11 13 14 22 <u>23</u> 25]

Figure 5 shows the individual obtained by the mutation operation.
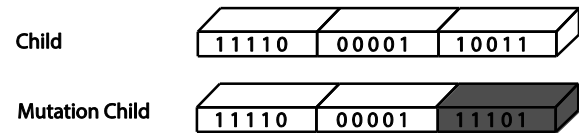
Fig. 5.   Mutation of individuals

3.6   Fitness function evaluation

In this optimization problem to evaluate the fitness function is required to evaluate the voltage sags in the system which is performed by using the fault position method [5][19]. This method consists of determining the voltage sags from short-circuit calculations considering different fault positions dispersed throughout the system. With the data of the fault probability at each fault position it can be obtained the number of expected voltage sags at the buses of the electrical system.

The procedure for application of the proposed reconfiguration method is described as follows:

1) Read the electrical system's parameter data and data of reference values of voltage sags for a considered voltage threshold (t_{spec}) at system buses, this is, f_{ref}.

2) Identify the base system and provide it as an individual in the initial population.

3) To generate the population as described in subsection *C*

4) Carry out operations the genetic operations, described above, to generate new individuals.

5) For each solution vector to identify the lines that are out of operation and change your status on the system parameters.

6) To evaluate the voltage sags in order to obtain the vector f_{est}. considering the current operating status of the system.

7) If not meet the criteria of stop, return to step 4 and re-evaluate possible solutions.

In this work, the proposed reconfiguration method has been implemented in *Matlab*® by using the implemented functions described in this section. In addition, functions *@fitscalingprop* and *@selectionroulette* from the *toolbox Global Optimization Toolbox* [23] are used.

4   Case studies

In order to show the performance of the proposed methodology, several case studies are presented. A IEEE-69 buses test systems have been used. It has been assumed that voltage sags are caused by three-phase balanced faults.

The IEEE 69 buses test system is a 12.66 KV radial distribution system (Fig. 6) consists of one substation, 69 buses interconnected by 73 lines, of which 5 are for tie line. The total of load active and reactive is 3802 KW and 2694 KVA respectively. The data system is provided in [24].
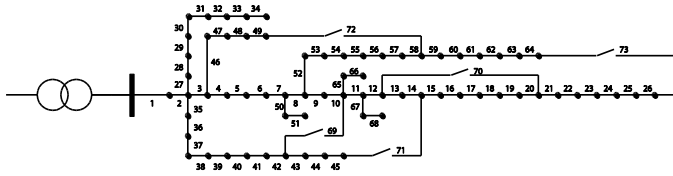
Fig. 6.   Distribution system of 69 buses

The fault position method was used in order to carry out the stochastic estimation of voltage sags, considering ten fault positions for each line, a uniform distribution of fault probability at lines, the original configuration of the distribution system, and a voltage sags threshold of 0.8 p.u.

For a voltage threshold of 0.7 p.u. initial system buses has 29 buses having voltage sags problems, in Table 1 system buses having problems of voltage sags are shown, and also the lines that are usually open are shown. Fig. 6 shows the distribution system 69 buses. It can be seen that five switches through normally open condition meets the radiality of the system.

Figure 7 shows the system after the application of the procedure, the reconfigured system for the voltage threshold of 0.7 p.u. in which it is observed that the system has a radial final topology

TABLE 1. ACTUAL STATUS OF THE 69 BUSES SYTEM BEFORE RECONFIGURACION

Voltage sags threshold	0.7 p.u.
Buses with values above reference voltage sags value	8,9,12,13,14,15,16,17,21,24, 25,27,51,52,53,54,55,56,57,58 59,60,61,64,65,66,67,68,69
Open lines	69 70 71 72 73
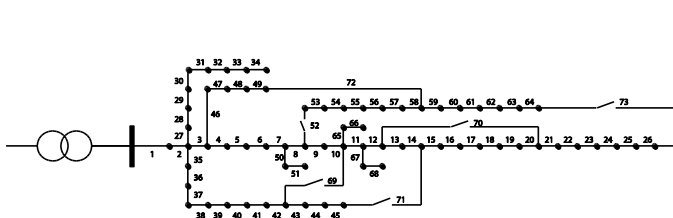
Fig. 7.   Distribution system of 69 buses after reconfiguration, considering the 0.7 p.u. voltage threshold

In Fig. 8 the voltage sags reference values, the initial voltage sags (SagsBase) and those obtained after the system reconfiguration for the distribution nodes 69 are graphed, considering a voltage threshold of 0.7 p.u. It can be seen that performing the reconfiguration voltage sags decrease depressions in several buses, but in some others buses the values of voltage sags increase. This is due to the change of network topology; however, the system operates correctly, and with a reduction in voltage sags with respect to the initial topology of the system.
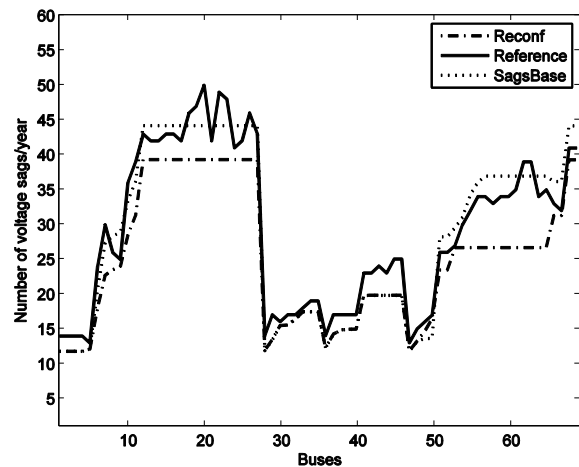
Fig. 8.   Reference, sags initial configuration and after reconfiguration values of voltage sags/year considering a 0.7 voltage threshold.

In Fig. 9 has been graphed the system of 69 buses when the voltage threshold of 0.8 p.u. is considered. In Table 2, it can be seen that initially had 38 nodes that did not meet the specified value. After reconfiguration (Fig. 10), there is a significant reduction when comparing the base system, with the founded reconfiguration. Some buses have voltage sags problems, but even so, the reduction in the system means better condition thereof. The results of this reconfiguration are summarized in Table 3, it can be seen that initially had 38 nodes that did not meet the specified value and in 26 of them has been reduced to appropriate values, leaving only 12 nodes that do not fulfill the specification.
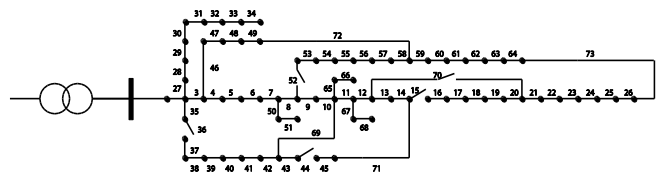
Fig. 9.   Distribution system of 69 buses after reconfiguration, considering the 0.8 p.u. voltage threshold

TABLE 2. ACTUAL STATUS OF THE 69 BUSES SYTEM BEFORE RECONFIGURACION

Voltage sags threshold	0.8 p.u.
Buses with values above reference voltage sags value	2,3,4,7,8,11,12,13,15,16,17,19,20,23, 24,26,27,29,30,31,33,34,37,38,40,41, 42,43,44,45,58,59,60,62,63,66,67,69
Open lines	69 70 71 72 73

TABLE 3. RESULTS FOR THE 69 BUSES RADIAL SYTEM AFTER RECONFIGURACION

Voltage sags threshold	0.7 p.u.	0.8 p.u.
Buses with values above reference voltage sags value	0	37,38,39,40, 41,42,43,44, 45,46,49,50
Open lines	69 70 71 52 73	36 70 44 52 15

In Fig. 10 has been graphed the voltage sags corresponding to initial system of 69 nodes for one threshold of 0.8 p.u. There is a significant reduction when comparing the base system, with the founded reconfiguration. Some buses have voltage depression problems, but even so, the reduction in the system means better condition thereof.
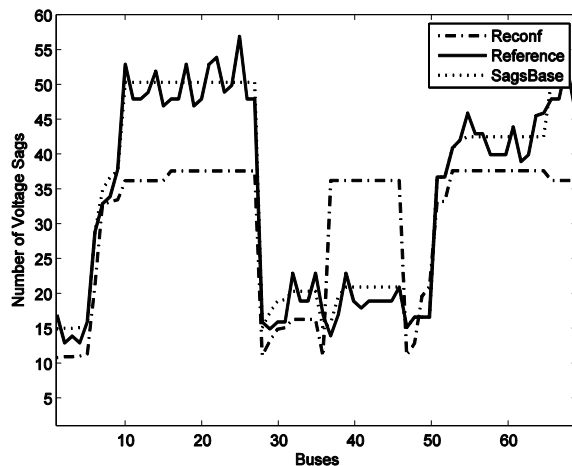
Fig. 10. Reference, sags initial configuration and after reconfiguration values of voltage sags/year considering a 0.8 voltage threshold.

Based on studies that have been conducted with the same objective function with a limited number of switches in the same case studies found that the search space and the computational cost has been reduced, reflecting the proper functioning of proposed algorithm and its evolution to a good solution.

For the IEEE 69 buses test system were used 10 possible switches, 10 generations and a population of 35 individuals. The proposed algorithm uses 57 possible lines to find a better solution with 4 generations and a population of 10 individuals.

5   Conclusions

This paper presents a methodology for the reconfiguration of radial distribution electrical systems in order to reduce voltage depressions, using a genetic algorithm. This algorithm generates and reproduces individuals in tree form or radial topologies given that initial population, crossover and mutation operators were designed for that purpose. This is reflected in the reduction of the search space for valid topologies. The proposed algorithm was applied to IEEE-69 buses test system and the results have shown the efficiency of the proposed method.

6   Acknowledgment

7   References

[1]   M. Kezunovic, *"Smart Fault Location for Smart Grids", IEEE Transaction on Smart Grid*, vol. 2, no. 1, March 2011, pp. 11-22.

[2]   J. A. Momoh., "Smart grid design for efficient and flexible power networks operation and control", *IEEE/PES PSCE'09 Power Systems Conference and Exposition*, Seattle - USA, March 15-18, 2009.

[3]   V. Barrera, A. Pavas, J. Meléndez,. "Power Quality Assessment of the Bogotá Distribution Network Focused on Voltage Sag Analysis"*, IEEE PES, Innovative Smart Grid Ttechnologies (ISGT) - Europe conference 2011*, Manchester, UK, December 2011.

[4]   M. H. J. Bollen, J. Zhong, F. Zavoda, J. Meyer, A. McEachern, and Felipe Córcoles López. "Power Quality aspects of Smart Grids", *International Conference on Renewable Energies and Power Quality*, March 2010.

[5]   M. H. J. Bollen, "Understanding power quality problems. Voltage sags and interruptions", *IEEE PRESS Series on Power Engineering*, 2000.

[6]   M. H. J. Bollen, "Method of critical distances for stochastic assessment of voltage sags", *IEEE Proc. Generation, Transmission and Distribution*, vol. 145, no. 1, pp. 70-76, Jan. 1998.

[7]   E. Espinosa, A. Hernández., "An analytical approach for stochastic assessment of balanced and unbalanced voltage

sags in large systems", *IEEE Trans. on Power Delivery*, vol. 21, no. 3, pp. 1493-1500, Jul. 2006.

[8] E. Espinosa-Juárez, A. Hernández, "A method for voltage sag state estimation in power systems", *IEEE Trans. on Power Delivery*, vol. 22, no.4, pp. 2517-2526, Oct. 2007.

[9] B. Wang, W. Xu, Z. Pan, "Voltage sag state estimation for power distribution systems", *IEEE Trans. on Power Delivery*, vol. 20, no. 2, pp. 806-812, May 2005.

[10] J. Lucio, E. Espinosa-Juárez, A. Hernández, "Voltage sag state estimation in power systems by applying genetic algorithms", *IET Generation, Transmission & Distribution*, vol. 25, no. 2, pp. 223-230, Feb. 2011.

[11] S. Civanlar, J. Grainger, H. Yin, S. Lee, "Distribution feeder reconfiguration for loss reduction," *IEEE Trans. on Power Del.*, vol. 3, no. 3, pp. 1217–1223, Jul. 1988.

[12] M. E. Baran and F. F.Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans .on Power Del.*, vol. 4, no. 2, pp. 1401–1407, Apr. 1989.

[13] C. Wang, H. Zhong, "Optimization of network configuration in large distribution system using plant growth simulation algorithm, " *IEEE Trans. on Power Systems,* vol. 23, no. 1, pp 119-126, February 2008.

[14] A. C. B. Delbem, A. C. Carvalho, N. G. Bretas, "Main chain representation for evolutionary algorithms applied to distribution system reconfiguration", *IEEE Trans. on Power Systems,* vol. 20, no. 1, pp 425-436, February 2005.

[15] E. M. Carreno, R. Romero, and A. Padilha-Feltrin, "An efficient codification to solve distribution network reconfiguration for loss reduction problem," *IEEE Trans. on Power Systems,* vol. 23, no. 4, pp. 1542- 1551, november 2008.

[16] J. Mendoza, R. Lopez, D. Morales, E. Lopez, P. Dessante, and R. Moraga, "Minimal loss reconfiguration using genetic algorithms with restricted population and addressed operators: Real application," *IEEE Trans. on Power Syst.*, vol. 21, no. 2, pp. 948–954, May 2006.

[17] R. S. Rao, S. V. L. Narasimham, M. R. Raju, A. S. Rao, "Optimal network reconfiguration of large-scale distribution system using harmony search algorithm", *IEEE Trans. on Power Systems,* vol. 26, no. 3, pp1080-1088, August 2011.

[18] M. Lavorato, J. F. Franco, M. J. Rider, R. Romero "Imposing radiality constraints in distribution system optimization problems", *IEEE Trans. on PowerSystems*, vol. 27, no. 1, pp. 172-180, February 2012.

[19] M. R. Qader, M. H. J. Bollen, "Stochastic prediction of voltage sags in a large transmission system", *IEEE Trans. on Industry Applications*, vol. 35, no. 1, pp.152-162, Jan. 1999.

[20] A. Homaifar, X. Q. Charlene, H. L. Steven, "Constrained Optimization Via Genetic Algorithms", *Simulation, vol.* 62, no. 4, pp. 242-253, April 1994.

[21] J. Z. Zhu, "Optimal reconfiguration of electrical distribution network using the refined genetic algorithm", *Elect. Power System Research*, vol. 62, pp. 37–42, 2002.

[22] J.C. Cebrian, N. Kagan, "Evolutionary algorithms for the reconfiguration of radial distribution networks considering long interruption and voltage sag disruption costs", *13th International Conference on Harmonics and Quality of Power, 2008.* Sept 2008.

[23] Global Optimization Toolbox. User's Guide, The MathWorks, Inc., 2012

[24] J. S. Savier, D. Das, "Impact of network reconfiguration on loss allocation of radial distribution systems", *IEEE Trans. on Power Delivery*, vol. 22, no. 4, pp 2473-2480, October 2007.

Adaptive Diversity of Genetic Algorithm for QoS-based Cloud Service Selection

Chengwen Zhang[1]**, Zhaoyong Xun**[2]**, Jian Kuang**[3]**, Bo Cheng**[4]**, and Lei Zhang**[5]

[1]Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts & Telecommunications, Beijing, China

[2]Sun Kaisens (Beijing) Technology Ltd., Beijing, China

[3]Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts & Telecommunications, Beijing, China

[4]State Key Lab of Networking and Switching Technology, Beijing University of Posts & Telecommunications, Beijing, China

[5]Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education, Beijing, China

Abstract - *An elitist selection adaptive Genetic Algorithm is presented to select an optimal cloud service composite plan from a lot of composite plans on the basis of global Quality-of-Service (QoS) constraints. In this Algorithm, some strategies are provided to improve convergence and population diversity and raise fitness result. These strategies include an elitist selection strategy, an individual selection method based on ranking selection and an adaptive mutation strategy. The elitist selection strategy gives a guarantee that the degradation of population fitness can be avoided. The individual selection method provides a simple and effective way to control selective pressure. The adaptive mutation strategy uses a special population measurement to decide individuals' similarity. The value of the dynamic mutation probability changes on the basis of population diversity. Some simulation results on cloud service selection with global QoS constraints have shown that the elitist selection adaptive Genetic Algorithm can gain better composition service plan.*

Keywords: Diversity, Genetic algorithm, QoS-aware, Cloud service selection

1 Introduction

With cloud computing [12, 14, 22], new Internet services can be developed and deployed without capital acquisitions of hardware or large human integration expenses. Cloud computing environments offer cloud services.

Some services can compose a composition service with more powerful functions [4]. In [26], a conceptual overview of the QoS-aware service selection problem was given. A composition service has specific functions that can be divided into some component functions. These component functions can be accomplished by some component services respectively. These services can combine tens of thousands composite services with same functions and different QoS.

That is, there are many different combination plans. How to select the most suitable composite service among many available candidate services for consumers is an interesting practical issue [23, 24]. The service selection with global QoS constraints possesses a considerably big proportion in the problem of QoS-based cloud service selection. QoS-based service selection plays an important role [9, 10] in service combination.

QoS-based cloud service selection problem is one of the hot research areas. The existing QoS-based service selection algorithms include calculation algorithms [1-3, 5-8, 15, 17-21, 47] based on QoS properties. The QoS-based calculation methods can better meet the combination of services with the global QoS constraints. Exhaustive methods and approximate algorithms are two kinds of QoS properties calculation methods. Finding the optimal composition requires enumerating all possible combinations of service candidates. Performing exhaustive search can be very expensive in terms of computation time and, therefore, inappropriate for run-time service selection in applications with many services and dynamic needs [26]. The problem of finding the best service composition without enumerating all possible combinations is considered as an optimization problem. QoS-based cloud service selection is NP-hard problem [3], therefore, approximate algorithm is more suitable to solve optimization combinatorial problems. Genetic Algorithm is a kind of approximate algorithm. It is a good method to solve optimization combinatorial problems [13].

Based on the above analyses, this paper presents an elitist selection adaptive Genetic Algorithm (EaGA). Firstly, an elitist selection strategy is proposed to raise the probability of convergence to the optimal solution. Then an individual selection method based on ranking selection is provided to improve the evolution direction of GA. Thirdly, an adaptive mutation strategy is described to meet the needs of various stages of Genetic Algorithms.

The remaining sections of this paper are as follows. Section two described researches of QoS-based cloud service selection computing. The proposed GA with improved diversity was discussed in detail in section three. Section four presented some simulation works and discussed the simulation results. Section five came to conclusions and noted that the next step in research content.

2   QoS-aware service selection

Based on all global QoS constraints, to select the best plan from a large number of service composition plans is in the area of combinatorial optimization. To solve such problems, the calculation methods based on QoS attributes are divided into two categories. One category is exhaustive algorithm. In this kind of algorithm, all of candidate plans are calculated according to certain rules in order to choose the best plan. The other is approximate algorithm. In this type of algorithm, an ideal composition plan is infinitely close to the best one. At last, a plan that meets all QoS requirements but is not the best one will be gained.

[8] presented a run-time services choice method in dynamic service composition. But, it was only a local optimization algorithm, rather than a global optimization algorithm. In [18, 19], a multi-constrained QoS service selection problem was modeled as an integer programming problem, and the integer programming expressions were presented for a variety of QoS constraints. The methods in [19] was modified by [5]. It analysed the conditions of triggering service re-selection in detail, gave the idea of the service re-selection and gave the constraint expression for a stateful service selection.

The above solutions are only acceptable if the number of service candidates is very limited. As a result, we can see the limitations of exhaustive calculations. In the field of combinatorial optimization, there is also a random search algorithm based on probability. Genetic Algorithms fall into this category. Genetic Algorithm is suitable for solving such problems [13], and it can effectively prevent exhaustive algorithm limitations. In [2], some comparison experiments were made between GA and Integer Programming (a kind of exhaustive search method). The numerical results showed that GA is better selection algorithm than Integer Programming when they are employed to select the optimal composition plans from a great deal of plans. The solution based on Genetic Algorithm is a novel global optimization one. [6, 15, 20, 21] used Genetic Algorithm for the optimization of service composition.

[21] used Genetic Algorithm to solve the QoS-based service selection. [3] also used an one-dimensional chromosome encoding method to describe the combination of services, but the expression of gene loci was different from [21]. [2, 3] proposed a combination service method based on Genetic Algorithm. Through Genetic Algorithm, it could be ensured that the results of services choice met the restrictive conditions.

In order to improve the "premature" phenomenon of Genetic Algorithm, an elitist selection adaptive Genetic Algorithm (EaGA) is proposed in this paper. It includes an elitist selection strategy, an individual selection method based on ranking selection and an adaptive mutation strategy.

3   Elitist selection adaptive Genetic Algorithm

In this section, we present an elitist selection adaptive Genetic Algorithm (EaGA) to solve quality-driven service selection, mainly including the design of an elitist selection strategy, an individual selection method based on ranking selection and an adaptive mutation strategy.

3.1   Model of service selection

In GA selection algorithm, a composition plan and an individual in population are one-one correspondence. An individual is a concrete composition plan. A population in an evolution generation makes up of many concrete composition plans that are selected by GA. A fitness function is the formula that is used to compute the overall quality score for each execution plan. Crossover operation is done at two chosen composition plans and mutation operation is made at one elected composition plan.

3.2   Elitist selection strategy

Elitist selection is a selection strategy where a limited number of individuals with the best fitness values are chosen to pass to the next generation, avoiding the crossover and mutation operators. Elitism prevents the random destruction by crossover or mutation operators of individuals with good genetics.

To ensure the optimal composition plan can be obtained, an elitist selection is adopted to guarantee that the best individual can survive after every evolution operation. In elitist strategy, the offspring have to compete with the parents to gain admission for next generation of GA [38]. The outstanding advantage of this environment is it always preserves the best solutions in every generation.

In initial population, individual with approximatively best fitness for every path is included. An individual with best fitness for one path may have smaller fitness than other individuals for other paths. They fail to compete with other individuals and can not get admission for next generation of GA. Maybe the information about the defeated path disappears in next population. The diversity of population will be disrupted. In our algorithm, to prevent the stagnation of GA and hold the diversity of population, a special elitist selection method is presented.

Firstly, in every generation of GA, the individual with best fitness for every path is obtained. They don't join in the crossover and mutation operations and directly enter into the next generation.

Secondly, according to the proportion of plans in every path to all plans, some more individuals with second best fitness become elite individuals. The total number of elite individuals should vary with the size of population and should not be too high, otherwise the population will tend to degenerate.

Cross through the establishment of the special elitist selection method, it is given a guarantee that the individual with best fitness value can be obtained and the optimal solution can be gotten. The better direction of evolution can be made. The degradation of population fitness can be avoided. Thus, population fitness can be enhanced with population evolution. Finally, higher fitness can be achieved.

3.3 Individual selection method based on ranking selection

Stochastic universal sampling (SUS) is a technique used in GA for selecting potentially useful solutions for recombination [42]. SUS is a development of fitness proportionate selection [43] which exhibits no bias and minimal spread. Where fitness proportionate selection chooses several solutions from the population by repeated random sampling, SUS uses a single random value to sample all of the solutions by choosing them at evenly spaced intervals.

The genetic algorithm community often cites the need to balance exploitation and exploration [39]. To achieve a better tradeoff between selective pressure and population diversity, ranking selection [39-41] is adopted to select individuals in each generation.

In ranking selection, the population is sorted according to the objective values. The fitness assigned to each individual depends only on its position in the individuals rank and not on the actual objective value.

Stagnation is in the case where the selective pressure is too small. Premature convergence is in the area where selection has caused the search to narrow down too quickly. The reproductive range is limited, so that no individuals generate an excessive number of offspring. Ranking introduces a uniform scaling across the population and provides a simple and effective way of controlling selective pressure. Ranking selection overcomes the scaling problems of the proportional fitness assignment.

In our GA selection algorithm, the ranking selection [39] employing the SUS algorithm [42] has been chosen to select individuals.

3.4 Adaptive mutation strategy

A mutation operation is that the values of the genetic loci in a chromosome are replaced by the other alleles of the genes loci in a certain probability (P_m, called the mutation probability) to form new individuals. The mutation operation is an essential operation because it determines the local search

ability of Genetic Algorithm. There are two main reasons for the mutation operation to be used in Genetic Algorithm. On one hand, the mutation operation can restore the lost alleles of the genes loci, so individual differences are maintained in a population and the premature phenomenon is prevented to occur. On the other hand, when the population is large, the introduction of appropriate mutations can improve the local search efficiency of Genetic Algorithm. The greater the mutation probability P_m is, the more the number of produced new individuals is. The diversity of population is better. But, it is more possible that a lot of good models of individuals will be destroyed and the performance of Genetic Algorithm is similar to random search algorithm. If the value is too small, the ability of generating new individuals and inhibiting "premature" will be poor.

If Genetic Algorithm uses fixed mutation probability, it is difficult to find a suitable fixed mutation probability to meet the needs of various stages of Genetic Algorithms and the needs of different individuals.

The parameter choice for mutation is critical to GA performance. It impacts on population diversity and the ability of GA to escape from local optima. Large value of mutation will transform GA into a purely random search algorithm, while the mutation is required to prevent the premature convergence of GA to suboptimal solutions. Aiming at achieving the trade-off between exploration and exploitation, the value of mutation probability should be able to vary.

Based on the No Free Lunch (NFL) Theorem [44], some mutation adaptation techniques are listed and discussed their effect on the algorithmic performance in presence of composition service selection landscapes having some special features.

Population level, individual level, and component level adaptation are three types of mutation adaptation techniques employed in GA [34]. In [33], some comparative analysis of different population-level and gene-level adaptive mutation operators for GAs were done. These adaptive mutation technologies [33-36] are functions of fitness, not include diversity.

The running of GA is easy to fall into "premature", that is, a running Genetic Algorithm has not the ability to further search more locations in the solution space and can obtain the final solution with not very good fitness. There are a lot of reasons about the "premature". One of the reasons is that the diversity of individuals in population is poor. When the diversity is bad, all of individuals in a population will be confined within some ranges in the solution space. In general, if population's scale is certain, the greater population diversity is, the more likely to produce better offspring. Conversely, as the diversity of population is reducing, the premature convergence phenomenon occurs more easily. So keeping population diversity has become an important method of genetic algorithm's effective operation and the correct

appraising of population diversity plays an important role in improving the performance.

In [16, 25, 27, 28, 32, 37], their adaptive mutation technologies are based on fitness diversity. More specifically, measurement of the fitness diversity is used to dynamically set the algorithmic parameters [30, 31] in order to prevent premature convergence and stagnation and therefore guarantee a more robust algorithm. A measurement of fitness diversity over the individuals of the population, namely ξ, was proposed in [28]. It has been successfully applied to a control problem in electrical machinery. In [27], another kind of measurement of fitness diversity, namely ψ, has been employed. A third kind of measurement of fitness diversity, namely v, has been employed in [25]. In [29], the above three fitness diversity methods were discussed and compared.

It must be noted that the three fitness diversity control methods shown above only take values of the best, average and the worst fitness over the individuals of the population.

In [32, 37], an adaptive mutation probability that adopted another kind of diversity that uses the individual distance to measure the similarity of individuals was employed.

How to select these adaptive mutation probability methods is very important for the landscape of QoS-aware service selection. Before it, the process of QoS-aware service selection is analyzed. In service computing, cloud services will be composed as part of workflows to build complex applications to achieve client requirements. Several services may be available with the same function, however they surely exhibit different QoS. The choice between different but semantically equivalent services is a function of QoS attributes.

From the above describe, the individuals with same fitness value may be different composition plans and the individuals with different fitness values may be in same composition path. So, in the context of composition service, fitness diversity is not enough to control population diversity. On the contrary, since composition service has the characteristic about workflow, it is a good way that the similarity between individuals is measured through the individual distance. Thus, we adopt the adaptive genetic algorithm proposed in [32].

Based on the different between calculated two individuals, there are two formulas to compute diversity.

First one is that the two individuals are two composition plans in same composition path. The distance equation (11) in [32] is employed.

Second one is that the two individuals are two composition plans in two different composition paths. Because the two individuals are dissimilar, the distance value $d(i, j)$ between two individuals i and j is set the predefined threshold D that is used in [32].

4 Tests and analyses of diversity GA

The proposed EaGA in this paper improves service selection algorithm based on Genetic Algorithm in three ways. The first one makes better direction of evolution through the elitist selection strategy. The second one ensures that the optimal solution has higher possibility to be obtained. The third one enriches the population diversity.

Through the above improvements, the EaGA has better search ability. Here are tests and test analyses through which the capacity and efficiency of EaGA will be validated.

4.1 Test data preparation

In order to verify the effect of services choice done by EaGA, some comparison tests between the Genetic Algorithm [3] (called SGA) and the EaGA algorithm were made.

In order to fairly test these algorithms, they would run in the same hardware and software operating environment. The size of population is 500, the crossover probability is 0.7. The number of iterations is 5000. For these two algorithms, 50 simulation experiments have been executed. Here, the bigger the fitness value of individual is, the better the composition plan is. For each of the 50 experiments, the maximum fitness for each algorithm has been saved. The average maximum fitness over the 50 experiments has been calculated for SGA and EaGA, this average value is also called Average Maximum Fitness (AMF).

Based on the above preparation of test data, SGA and EaGA algorithms were run respectively. The test results were analyzed from search capability. Search capability is that the algorithm can find the optimal solution in a solution space. It can be measured by the quality of the solution that the algorithm searches. In Genetic Algorithm, the algorithm search capability can be measured through the fitness value of the final selected individual.

4.2 Tests and analyses

In order to verify these strategies, this section presents the AMF result of 50 independent runs of each algorithm.

At some scale of problems (that is, the number of different tasks and different number of candidate services), the SGA and EaGA were run for 50 times respectively. A few of test result data are listed in Table 1. The statistic data in table 1 came from two GAs running for 50 times.

Table 1. Average maximum fitness value (EaGA : SGA)

Task Num	Average Maximum Fitness	Standard Deviation of Maximum Fitness
10	0.265 : 0.132	0.078 : 0.039
25	0.233 : 0.056	0.012 : 0.016
30	0.362 : 0.073	0.018 : 0.022

For each algorithm under examination, Table 1 gives the AMF over the 50 experiment runs.

As can be seen from table 1, EaGA adopts the elitist selection strategy and avoids that the optimal individual is replaced during population evolution. Meanwhile, EaGA

reduces the randomness of evolutionary computation on the basis of the elitist selection strategy. Meanwhile, some individuals with approximatively best fitness for every path are included in initial population in EaGA. The starting point of EaGA is higher than SGA. The introduction of the elitist selection strategy is the reason why individual fitness is better and better after evolution operations. This is the first reason why EaGA is better than SGA.

Because ranking selection introduces a uniform scaling across the population and provides a simple and effective way of controlling selective pressure, stagnation and premature convergence can be overcome. The individual selection method based on ranking selection in EaGA is good at searching optimal composition plans. This is the second reason why EaGA is better than SGA.

The population diversity varies during all phases of running time of GA. The mutation probability should also change with the variation of the population diversity. At the beginning stage of GA, the population diversity is better and the mutation probability should be smaller. At the late running stage of GA, the population diversity gets worse and worse, the mutation probability of GA should be bigger. The decrease of the mutation probability avoids random search and the increase of the mutation probability leads to rich composition plans. From the formula of the dynamic mutation probability of EaGA, the value of the dynamic mutation probability changes on the basis of population diversity. So, the mutation operation can raise the individual fitness and improve the population diversity at the same time. The evolution direction of population can be guaranteed and the search scope of composition plans can be expanded. Thereby, the fitness of EaGA is larger than SGA. This is the third reason why EaGA is better than SGA.

As can be seen from the above simulation results, it can be seen that the best composition plan is obtained by EaGA. This phenomenon is due to successful runs of the elitist selection strategy, the individual selection method based on ranking selection and the adaptive mutation strategy. The elitist selection strategy, the individual selection method based on ranking selection and the adaptive mutation strategy in EaGA can work together very well. EaGA outperforms SGA. In particular, EaGA is proved to have better performance than SGA in avoiding premature convergence. The proposed EaGA can become a good method to select service plan based on global QoS constraints.

5   Conclusions

With the cloud services technologies have become more sophisticated, more and more easily used cloud services with the stability characteristics are shared on network. But a single atomic cloud services can provide limited functionalities. In order to more fully utilize the shared cloud services, it is necessary to combine shared cloud services to form a new combination of cloud services to provide more powerful service functions.

This paper presents an elitist selection adaptive Genetic Algorithm (EaGA). Based on the analyses how to enhance fitness result and diversity of GA, some methods are provided. They include an elitist selection strategy, an individual selection method based on ranking selection and an adaptive mutation strategy. The elitist selection strategy gives a guarantee that the optimal individual can be gained. The degradation of population fitness can be avoided. The individual selection method is based on ranking selection. This way provides a simple and effective way of controlling selective pressure. The adaptive mutation strategy uses an individual distance to measure the similarity of individuals. And, two formulas to compute diversity are proposed on the basis of the characteristics of composition service.

Through these strategies working together, the search ability can be improved. After testing and analyses of test results, some strong validations of the proposed algorithm in capacity and efficiency effects were done. The provided EaGA can be a good solution to QoS-driven cloud services selection.

To inhibit the premature convergence and stagnation and therefore to guarantee a more robust algorithm, the algorithmic parameters should be set dynamically [45]. The population size of GA is one of the most important algorithmic parameters. When the population is highly diverse, a small number of solutions need to be exploited. And, when the population is converging and a larger population size is required to increase the exploration. So, the size of the population is therefore dynamic [46]. Therefore, to really modify the population diversity, the next study will present a population resizing mechanism on the basis of the special features of QoS-aware service selection.

6   References

[1]   Laiping Zhao, Yizhi Ren and Mingchu Li. "Flexible service selection with user-specific QoS support in service-oriented architecture" ; Journal of Network and Computer Applications, 35, 3, 962—973, 2012.

[2]   Canfora G, Dipenta M and Esposito R. "An approach for QoS-aware service composition based on genetic algorithms". Proc. Int. Conf. Genetic and Evolutionary Computation, Washington, 1069—1075, 2005.

[3]   Canfora G, Penta M D, Esposito R and Villani M L. "A lightweight approach for QoS–aware service composition". Proc. 2nd Int. Conf. Service Oriented Computing, ICSOC'04, New York, USA, 36—47, 2004.

[4]   Curbera F, Khalaf R, Mukhi N, Tai S and Weerawarana S. "The next step in web services" ; Commnincation of the ACM, 46, 10, 29—34, 2003.

[5]   Danilo A and Barbara P. "Adaptive service composition in flexible processes" ; IEEE Transactions on Software Engineering, 33, 6, 369—384, 2007.

[6]   Gao C M, Cai M L and Chen H W. "QoS-aware service composition based on tree-coded Genetic Algorithm". 31st Annual International Computer Software and Applications Conference (COMPSAC 2007), 1, 361—367, 2007.

[7]   Jin C H, Wu M H, Jiang T and Ying J. "Combine automatic and manual process on web service selection and composition to support QoS". Proc. 12th Int. Conf. Computer Supported Cooperative Work in Design, CSCWD'08, Xi'an, China, 459—464, 2008.

[8]   Liu Y, Ngu A H and Zeng L. "QoS computation and policing in dynamic web service selection". Proc. 13th Int. Conf. World Wide Web, WWW'04, ACM Press, New York, USA, 66—73, 2004.

[9]   Menascé D A. "QoS issues in web services" ; IEEE Internet Computing. 6, 6, 72—75, 2002.

[10] Menascé D A. "Composing web services: a QoS view" ; IEEE Internet Computing, 8, 6, 88—90, 2004.

[11] Rojas I, González J, Pomares H, Merelo J J, Castillo P A and Romero G. "Statistical analysis of the main parameters involved in the design of a genetic algorithm" ; IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews, 32, 1, 31—37, 2002.

[12] Rochwerger B, Breitgand D and Levy E, et al. "The reservoir model and architecture for open federated cloud computing" ; IBM Journal of Research and Development, 53, 4, 1—17, 2009.

[13] Man K F, Tang K S, and Kwong S. "Genetic algorithms: concepts and applications" ;IEEE Trans. Industrial Electronics, 43, 519—534, 1996.

[14] Armburst M, et al. "Above the clouds: a Berkeley view of cloud computing". Tech. report UCB/EECS-2009-28, Electrical Eng. and Computer Science Dept., Univ. of California, Berkeley, 2009.

[15] Yan Y H and Liang Y. "Using genetic algorithms to navigate partial enumerable problem space for web services composition". Proc. 3rd Int. Conf. Natural Computation, ICNC'07, China, 475—479, 2007.

[16] Weber M, Tirronen V and Neri F. "Fitness diversity parallel evolutionary algorithms in turtle race game".

Applications of Evolutionary Computing, Lecture Notes in Computer Science, Springer, 5484, 303—312, 2009.

[17] Yu T, Zhang Y and Lin K J. "Efficient algorithms for web services selection with end-to-end QoS constraints" ; ACM Transactions on the Web, 1, 1, 1—26, 2007.

[18] Zeng L Z, Benatallah B, Dumas M, Kalagnanam J and Sheng Q Z. "Quality driven web services composition". Proc. 12th Int. Conf. World Wide Web, WWW'03, ACM Press, Budapest, Hungary, 411—421, 2003.

[19] Zeng L Z, Benatallah Boualem, Ngu A, Dumas M, Kalagnanam J and Chang H. "QoS-aware middleware for web services composition" ; IEEE Transactions on Software Engineering, 30, 5, 311—327, 2004.

[20] Zhang C W, Su S and Chen J L. "DiGA: population diversity handling Genetic Algorithm for QoS-Aware web services selection" ; Computer Communications, Elsevier, 30, 5, 1082—1090, 2007.

[21] Zhang L J, Li B, Chao T and Chang H. "On demand web services-based business process composition". IEEE, Proc. Int. Conf. System, Man, and Cybernetics, SMC'03, Washington, USA, 4057—4064, 2003.

[22] Hayes B. "Cloud computing" ; Comm. ACM, 51, 7, 9—11, 2008.

[23] Liang Q, Wu X, and Lau H C. "Optimizing service systems based on application-level QoS" ; IEEE Transactions on Services Computing, 2, 108—121, 2009.

[24] Sheng Q Z, Benatallah B, Maamar Z and Ngu A H H. "Configurable composition and adaptive provisioning of web services" ; IEEE Transactions on Services Computing, 2, 34—49, 2009.

[25] Tirronen V, Neri F, Karkkainen T, Majava K and Rossi T. "A memetic differential evolution in filter design for defect detection in paper production". Applications of Evolutionary Computing, Lectures Notes in Computer Science, Berlin, Germany: Springer, 320–329, 2007.

[26] Alrifai M and Risse T. "Combining global optimization with local selection for efficient QoS-aware service composition". 18th international conference on World Wide Web (WWW), 2009.

[27] Neri F, Toivanen J, Cascella G L and Ong Y S. "An adaptive multimeme algorithm for designing HIV multidrug therapies" ; IEEE/ACM Transactions on Computational Biology and Bioinformatics, Special Issue on Computational Intelligence Approaches in Computational Biology and Bioinformatics, 4, 2, 264—278, 2007.

[28] Caponio A, Cascella G L, Neri F, Salvatore N and Sumner M. "A fast adaptive memetic algorithm for off-line and on-line control design of PMSM drives" ;IEEE Transactions on Systems, Man and Cybernetics - Part B, Special Issue on Memetic Algorithms, 37, 1, 28—41, 2007.

[29] Neri F, Tirronen V, Karkkainen T and Rossi T. "Fitness diversity based adaptation in multimeme algorithms:a comparative study". IEEE Congress on Evolutionary Computation (CEC 2007), 2374—2381, 2007.

[30] Back T. "The interaction rate of mutation rate, selection, and selfadaptation within a genetic algorithm". Proceedings of Parallel Problem Solving from Nature (PPSN-II). Elsevier Science, 85–94, 1992.

[31] Eiben A E, Hinterding R, and Michaelwicz Z. "Parameter control, evolutionary computation, advanced algorithms and operators". Institute of Physics Publishing, 170–187, 2000.

[32] Tang Zhihua, Zhu Youtuan, Wei Guo and Zhu Jinkang. "An elitist selection adaptive genetic algorithm for resource allocation in multiuser packet-based OFDM systems". Journal of Communications, 3, 27—32, 2008.

[33] Korejo Imtiaz, Yang Shengxiang, and Li Changhe. "A comparative study of adaptive mutation operators for Genetic Algorithms". MIC 2009: The VIII Metaheuristics International Conference, Hamburg, Germany, 2009.

[34] Angeline P J. "Adaptive and self-adaptive evolutionary computations, computational intelligence: a dynamic systems perspective". IEEE Press, New York, 152—163, 1995.

[35] Libelli S M and Alba P. "Adaptive mutation in genetic algorithms" ; Soft Computing, 4, 76—80, 2000.

[36] Srinivas M, and Patnaik L M. "Adaptive probabilities of crossover genetic in mutation and algorithms" ; IEEE Transactions on systems, man and cybernetics, 24, 4, 656—667, 1994.

[37] Zhang Jinhua and Hu Tiesong. "Adaptive genetic algorithm based on population diversity". Computer Engineering and Applications, 9, 49—51, 2002.

[38] Thierens D. "Selection schemes, elitist recombination and selection intensity". International conference of genetic algorithm, 152—159, 1997.

[39] Whitley D. "The genitor algorithm and selection pressure: why rank-based allocation of reproductive trials is best". Proc. of the Third International Conference on Genetic Algorithms, 116—121, 1989.

[40] Baker J E. "Adaptive selection methods for genetic algorithms". Proc. of the First International Conference on Genetic Algorithms and Their Applications, 101–111, 1985.

[41] Runarsson T P, Yao X. "Stochastic ranking for constrained evolutionary optimization" ;IEEE Trans. Evol. Comput., 4, 3, 284—294, 2000.

[42] Baker J E. "Reducing bias and inefficiency in the selection algorithm". Proc. of the Second International Conference on Genetic Algorithms, 14—18, 1987.

[43] Eiben A E and Smith J E. "Introduction to evolutionary computation". Berlin: Springer-verlag, 2003.

[44] Wolpert D and Macready W. "No free lunch theorems for optimization" ;IEEE Transactions on Evolutionary Computation, 1, 1, 67–82, 1997.

[45] Bäck T. "The interaction rate of mutation rate, selection, and self-adaptation within a genetic algorithm". Proc. PPSN-II, 85–94, 1992.

[46] Arabas J, Michalewicz Z, and Mulawka J. "GAVaPS—a genetic algorithm with varying population size". Proc. 1st IEEE Conf. Evolutionary Computation, and IEEE World Congr. Comput. Intell., 1, 73–78, 1994.

[47] Jun Huang, Yanbing Liu and Ruozhou Yu. "Modeling and algorithms for QoS-aware service composition in virtualization-based cloud computing" ; IEICE Transactions on Communications, E96B, 1, 10—19, 2013.

Multiple Determination of Chemical Elements Using a Multi-objective Evolutionary Algorithm and Multiple Linear Regression

Anderson da Silva Soares,
Telma Woerle de Lima,
Daniel Vitor de Lucena,
Rogerio Lopes Salvini
Intitute of Informatics
Federal University of Goiás
Goiânia, Brazil

Gustavo Teodoro Laureano
Department of Electrical Engineering
University of São Paulo
São Carlos, Brazil

Clarimar José Coelho
Computer Science Departament
Pontifical University Catholic of Goiás
Goiânia, Brazil

Abstract—The multiple determination task of chemical properties is a classical problem in analytical chemistry. The major problem is concerned in to find the best subset of variables that better represents the compounds. These variables are obtained by a spectrophotometer device. This device measure hundreds of correlated variables related with physicochemical properties and that can be used to estimate the component of interest. The problem is the selection of a subset informative and uncorrelated variables that help the minimization of prediction error. Classical algorithms select a subset of variables for each compound considered. In this work we propose the use of a multiobjective algorithm SPEA-II. We would like to show that the variable selection algorithm can selected just one subset used for multiple determination using multiple linear regression. For the case study is used wheat data obtained by NIR spectrometry where the objective is the determination of a variable subgroup with information about e Protein content (%), test weight (Kg/Hl), PSI (wheat kernel texture) (%) and farinograph water absorption (%). The results of traditional techniques of multivariate calibration as the Successive Projections Algorithm (SPA), Partial Least Square (PLS) and mono-objective genetic algorithm are presents for comparisons. For NIR spectral analysis of protein concentration on wheat, the number of variables selected from 775 spectral variables was reduced for just 10 in the SPEA-II algorithm. The prediction error decreased from 0.2 in the classical methods to 0.09 in proposed approach, a reduction of 37%. The model using variables selected by SPEA-II had better prediction performance than classical algorithms and full-spectrum partial least-squares (PLS).

Keywords: Multi-objective algorithms, variable selection, linear regression

I. INTRODUCTION

Quantitative chemical analysis is the science of the concentration determination of one or several substances present in a sample. The most modern techniques for this task is the use of spectrophotometric technic whose measure the interaction between the object in analysis and radiated energy supported by Lambert-Beer law [18], [17]. Such Figure show the sample receiving radiation P_0, and passing thru with a smaller energy P. The absorbed energy of the sample could be measure with spectrophotometer and related with the propriety concentration [10].

To obtain the concentration of entire sample, it is necessary to radiate different wavelengths simultaneously. In this scenario is normal wavelengths overlapping and consequently two or more signals sending the same information. In algebra terms the waves overlapping means high correlation among variables and can induce to mathematical problems in the regression model process [16].

Let a sample including two absorbances (A and B) with spectral overlapping $\lambda(1)$ and $\lambda(2)$, is possible to get y_A and y_B like as

$$x(\lambda_1) = k_A(\lambda_1)y_A + k_B(\lambda_1)y_B$$
$$x(\lambda_2) = k_A(\lambda_2)y_A + k_B(\lambda_2)y_B \qquad (1)$$

$$\begin{bmatrix} x(\lambda_1) \\ x(\lambda_2) \end{bmatrix} = \begin{bmatrix} k_A(\lambda_1) & k_B(\lambda_1) \\ k_A(\lambda_2) & k_B(\lambda_2) \end{bmatrix} \begin{bmatrix} y_A \\ y_B \end{bmatrix}$$
$$\begin{bmatrix} y_A \\ y_B \end{bmatrix} = \begin{bmatrix} k_A(\lambda_1) & k_B(\lambda_1) \\ k_A(\lambda_2) & k_B(\lambda_2) \end{bmatrix}^{-1} \begin{bmatrix} x(\lambda_1) \\ x(\lambda_2) \end{bmatrix} \qquad (2)$$
$$y_A = b_A(\lambda_1)(\lambda_1) + b_A(\lambda_2)(\lambda_2)$$
$$y_B = b_B(\lambda_1)(\lambda_1) + b_B(\lambda_2)(\lambda_2)$$

In general terms, the multivariate model is given by

$$y = x_0 b_0 + x_1 b_1 + \ldots + x_{J-1} b_{J-1} + \epsilon \qquad (3)$$

or in vectorial notation,

$$Y = X\beta + \epsilon \qquad (4)$$

with $x = [x_0 \ x_1 \ldots x_{J-1}]$ is the vector of measured values, $\beta = [b_0 \ b_1 \ldots b_{J-1}]^T$ is the vector to be determined and ϵ is a part of random error.

In the case of i samples are available with n wavelength, we can arrange in pairs $(x_i, y_i) \in \mathbf{R}^J \times \mathbf{R}$ like as

$$Y = \begin{bmatrix} y_1^a \\ y_2^a \\ \vdots \\ y_i^a \end{bmatrix} \quad X = \begin{bmatrix} x_1^1(\lambda_1) & \ldots & x_1^j(\lambda_n) \\ x_2^1(\lambda_1) & \ldots & x_2^j(\lambda_n) \\ \vdots & \ddots & \vdots \\ x_i^1(\lambda_1) & \ldots & x_i^j(\lambda_n) \end{bmatrix}, \quad (5)$$

where $x_i^j(\lambda_n)$ is the i-th sample of object in the wavelength λ_n and y_i^a is the concentration of a in the i-th sample. Where the relation between the absorbance and concentration can be estimate by a coefficient matrix β that multiply X for to obtain $\hat{Y}$ estimate. The matrix X and Y are divided in X_{cal} and Y_{cal} for obtain the coefficient matrix β and X_{teste} and Y_{teste} are used to test the accuracy of prediction model. The coefficients β can be obtained by linear regression model according the Equation(6).

$$\beta = (X_{cal}^T X_{cal})^{-1} X_{cal}^T Y_{cal} \qquad (6)$$

and $\hat{Y}$ can be estimate like as

$$\hat{Y} = X_{teste}\beta \qquad (7)$$

The problem happens because the devices have been developed to more accurately measure the absorbance, generating a lot of variables. As a consequence there are more wavelengths (variables) than samples (equations), in the case study of this work for example, we have 775 variables and 389 samples in X_{cal} matrix using a device not much modern. The most modern devices generate thousands of variables. In the Equation 6, if the number of variables is major than number of sample, the inversion is not possible or ill-conditioned. One solution is the use of variable selection algorithms like as genetic algorithm to choice a variable subset not redundant and without collinearity from the original set or the use of new variables obtained from linear transformations like as Partial Least Square (PLS) Algorithm.

The literature about this problem (see references [1], [15], [21], [19]) indicates that the genetic algorithm select a number of variables lager than classical methods like as partial least square algorithm and Successive Projections Algorithm (SPA). In this sense we propose the use of a multi-objective formulation to variable selection problem. We use the error of prediction and the number of variables in the fitness evaluation method. Like as discuss in [9], the multi-objective formulation can improve the regression model generalization ability. In the section III we show that the use of just error prediction can guide the genetic algorithm for a model with excess of variables and low generalization power. Additionally a decision maker method based on statistical test for choice a final solution from the pareto front is proposed.

II. Background

A. Multicollinearity Problem and Variables Selection

The existence of linear correlation between two or more independent variables in a multiple regression model is defined as multicollinearity [3]. This problem may cause difficulty with the reliability of the estimates of the model coefficients and difficulty in understanding the values obtained in response variable [8], [2].

In prediction problems when the regression model have many variables, the larger part can contribute little or nothing to prediction precision, therefore, select a reduced set with the variables that do influence positively in the regression model is crucial. To define a smaller set of independent explanatory

variables to be included in the final regression model is a frequent problem in regression problem. The problem of determining an appropriate equation based on a subset of the original set of variables include the criterion used to analyze the variables and select a subset and to estimate of the coefficients in the equation (6).

According to Miller [14] the reasons for using only some of the available or possible predictor variables include: a) to estimate or predict at lower cost by reducing the number of variables on which data are collected, b) to predict accurately by eliminating uninformative variables, c) to describe a multivariate data set parsimoniously, d) to estimate regression coefficients with small standard errors (particularly when some of the predictors are highly correlated). The proposed strategy to the problem of variables selection for multiple linear regression is the use of genetic algorithm to solve the multicollinearity problem, reduce cost by reducing the number of variables and minimize the residuals errors.

B. Classical Methods for Variable Selection in Calibration Problems

There are three classical algorithms for variable selection in calibration problems: the Successive Projections Algorithm (SPA), Genetic Algorithm (GA) and Partial Least Square Algorithm (PLS) [19]. The SPA and GA works in the original domain of variables whereas PLS instead of finding hyperplanes of minimum variance between the response and independent variables, it finds a linear regression model by projecting the predicted variables and the observable variables to a new space combining new variables from principal component analysis.

The successive projections algorithm (SPA) is a forward variable selection technique designed to minimize collinearity problems in multiple linear regression (MLR) [11]. SPA comprises two main phases: The first consists of projection operations carried out on the matrix X_{cal}. These projections are used to generate chains of variables. Each element in a chain is selected in order to show the least collinearity with the previous one. In the next phase the candidate subsets of variables are evaluated according to the Root Mean Square Error of Prediction (RMSEP) (Equation 8) predictive performance in the MLR model. The RMSEP evaluates how much the concentration predicted by the model approximates from the expected concentration.

$$\text{RMSEP} = \frac{\sum_{i=0}^{N}(\hat{y}_i - y_i)^2}{N} \qquad (8)$$

where $\hat{y}$ is the predicted value obtained by Equation 7, y is the real value of the concentration and N the total number of samples.

The RMSEP guide the evaluation of subset of variables used in the calibration model and allows us chose models more suitable to prediction. In this sense this measure is used also in fitness function of genetic algorithm.

The last results of multivariate calibration literature showed that the SPA-MLR has the better results in terms of RMSEP and parsimony (number of variables selected) when compared with the classical genetic algorithm and PLS [11], [5], [4], [21]. However in this work we proposed a new implementation

of genetic algorithm that include the use of multi-objective fitness.

III. Multi-objective Formulation of Variable Selection Problem

The classical genetic algorithm is designed to minimize the same function of SPA, that is, the Equation (8). However, as soon as the RMSEP reduce, more variables are include in the model. In [6] we demonstrate that RMSEP and the number of variables are conflicting goals. In spite of the RMSEP is reduced as soon as more variables can be included in the model. On the other hand if the number of variables is lager, the Equation (6) have bad condition and consequently bad generalization in new samples. In this sense we proposed the multi-objective formulation in the genetic algorithm where the first objective is minimize the Equation 8 and the second objective is the minimization of number of variables selected.

We proposed the use of NSGA-II and SPEA-II algorithms. Developed by Deb et. al [7], NSGA-II, as the first NSGA version, implements the dominance concept, classifying population in fronts accordingly to its dominance level []. The best solutions of each generation are located at the first front while the worst are located at the last front. The process of classification occurs until all population individuals are located at a front. Finalized this process of classification, individuals belonging to first front are non-dominated, but dominate individuals from second front and the individuals from the second front dominate the individuals from the third front and so on. The main difference from NSGA-II to a simple GA is the way the selection operator is applied, and this operator is subdivided in two process: Fast Non-Dominated Sorting and Crowding-Distance. The other operators are applied on traditional way.

Strength Pareto Evolutionary Algorithm (SPEA) is an extension of the Genetic Algorithm for multiple objective optimization problems [22]. It is related to sibling Evolutionary Algorithms such as Non-dominated Sorting Genetic Algorithm (NSGA), Vector-Evaluated Genetic Algorithm (VEGA), and Pareto Archived Evolution Strategy (PAES). There are two versions of SPEA, the original SPEA algorithm and the extension SPEA-II. The objective of the algorithm is to locate and and maintain a front of non-dominated solutions, ideally a set of Pareto optimal solutions. This is achieved by using an evolutionary process to explore the search space, and a selection process that uses a combination of the degree to which a candidate solution is dominated (strength) and an estimation of density of the Pareto front as an assigned fitness. Algorithm maintains an external population at every generation storing all nondominates solutions obtained so far. At each generation external population is mixed with the current population. All nondominated solutions in the mixed population are assigned fitness based on the number of solutions they dominate.

A. Multi-objective Decision Maker Method

Multi-objective algorithm present a set of solutions for multi-objective problem at its first front. To help choosing a solution within this set, it were applied the Wilcoxon signed rank test [12] as a multi-objective decision maker method.

The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used when comparing two related samples

on a single sample to assess whether their population mean ranks differ [12]. It can be used as an alternative to the paired Student's t-test for dependent samples when the population cannot be assumed to be normally distributed.

Let $\hat{Y}^i = y_1^i, y_2^i, \ldots, y_N^i$ the $i - th$ estimated vector of protein content and ϵ^i the difference vector between the estimated value $\hat{Y}^i$ and the real value Y and ϵ^j the difference vector between the estimated value $\hat{Y}^j$ and the real value Y. The decision maker algorithm in the first step choice the chromosome with the less value in the pareto front calculated from validation set. We like to know if the ϵ^i obtained with K variables not have significative difference with ϵ^j obtained with $K - P$ variables. That is, the less variable number without decrease the ability prediction. The null hypothesis is formulated by a two-sided test of the hypothesis that ϵ^i - ϵ^j comes from a distribution whose median is zero, that is, the difference not can be significative.

IV. Experimental

Samples are from whole grain wheat, obtained from vegetal material from occidental Canadian producers. The standard data were determined at the Grain Research Laboratory [9]. The data set for the multivariate calibration study consists of 775 VIS-NIR spectra of whole-kernel wheat samples, which were used as shoot-out data in the 2008 International Diffuse Reflectance Conference (http://www.idrc-chambersburg.org/shootout.html). Protein content, test weight, PSI (wheat kernel texture) and farinograph water absorption were chosen as the properties of interest. Test weight is used as an indicator of general grain quality and is a measure of grain bulk density. Test weight, but not overall grain weight, normally increases during drying. Spectra were acquired in the range 400-2500 nm with a resolution of 2 nm. In order to remove undesirable baseline features, first derivative spectra were calculated by using a Savitzky-Golay filter with a 2nd order polynomial and an 11-points window [20].

The Kennard-Stone (KS) [13] algorithm was applied to the resulting spectra to divide the data into calibration, validation and prediction sets with 259, 258 and 258 samples, respectively. The validation set was employed to guide the selection of variables in SPA-MLR, MONO-GA-MLR, NSGA-II-MLR and SPEA-II-MLR. The prediction set was only employed in the final performance assessment of the resulting MLR models. In the PLS study, the calibration and validation sets were joined into a single modeling set, which was used in the leave-one-out cross-validation procedure.

A. Environment and Tools

For executing the NSGA-II-MLR, SPEA-II-MLR, Mono-objective GA, SPA and PLS algorithm were used the Matlab software version 7.10 (R2010a). Table I shows the configuration for NSGA-II-MLR and SPEA-II-MLR algorithms. MONO-GA-MLR has the same parameters of multiobjective algorithms.

V. Results and Discussion

Figure 1 presents the derivative spectra of wheat sample. As can be seem, there are several of spectral variables available for selection with different absorbance (λ).

TABLE I.　　MULTI-OBJECTIVE ALGORITHMS NSGA-II AND SPEA-II CONFIGURATION.

	NSGA-II and SPEA-II
Population Size	100
Generations Number	100
Selection Operator	Binary Tournament
Mutation Operator	Flip
Mutation Probability	0.5 in the individual and 0.05 in the gene
Crossover Operator	Uniform Crossover and One Point
Crossover Probability	0.5 and 1
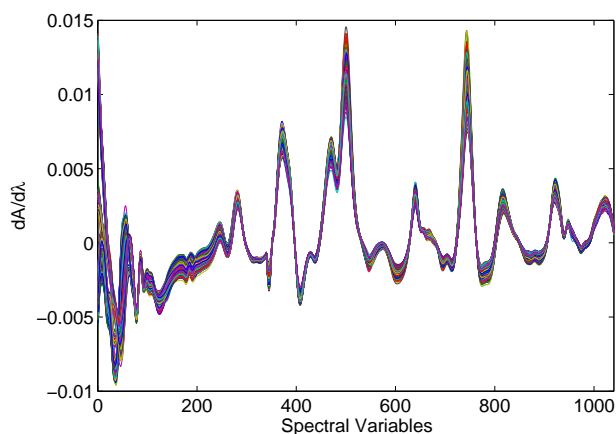
Fig. 1.　Derivative NIR spectra of the wheat samples.

TABLE II.　　RESULTS OF TRADITIONAL TECHNIQUES PLS, SPA-MLR AND MONO-GA-MLR. THE RESULTS ARE EXPRESSED IN RMSEP TERMS.

Protein content (%)		
	RMSEP	Number of Variables
PLS	0.21	15[*]
SPA-MLR	0.20	13
MONO-GA-MLR	0.21	146
test weight (Kg/Hl)		
	RMSEP	Number of Variables
PLS	1.23	5[*]
SPA-MLR	1.2	29
MONO-GA-MLR	1.38	112
PSI (wheat kernel texture) (%)		
	RMSEP	Number of Variables
PLS	2.76	11[*]
SPA-MLR	2.57	36
MONO-GA-MLR	2.69	157
Farinograph water absorption (%)		
	RMSEP	Number of Variables
PLS	2.11	7[*]
SPA-MLR	2.14	18
MONO-GA-MLR	2.41	96

[*]Number of latent variables.
Range of protein content in the prediction set: 10.2-16.2 % m/m.
Range of test weight in the prediction set: 78.2-84.7 (Kg/Hl).
Range of PSI (wheat kernel texture) in the prediction set: 48-73 % m/m.
Range of PSI (wheat kernel texture) in the prediction set: 53.1-75.6 % m/m.

First of all, we describe the results of the classical algorithms PLS, SPA-MLR and MONO-GA-MLR. These results are presented on Table II. As can be seem the RMSEP are similar for the three algorithms in all of elements studied. However the MONO-GA-MLR uses a expressive number of variables when compared with SPA-MLR. This result can be explained by the fact of MONO-GA-MLR use just one objective, the RMSEP in the validation set. In practice the SPA-MLR is used because it uses less variables than MONO-GA-MLR and PLS. Worth noting that PLS use all original variables to build the new latent variables. The next paragraphs presents the results obtained by the proposal algorithms, NSGA-II-MLR and SPEA-II-MLR under 30 executions each.

Figure 2 shows one of the Pareto front obtained by NSGA-II-MLR (Figure 2(a)) and SPEA-II-MLR (Figure 2(b)) for test weight concentrations. As can be seem, both algorithms minimized the two objectives, the number of variables and the RMSEP in the validation set. However, the SPEA-II-MLR algorithm have a boundary better distributed in the objectives space. For this property NSGA-II-MLR found solutions with the minimum number of 15 variables, while SPEA-II-MLR found solutions with just 3 variables. These Figure also shows the decision maker result for both algorithms in this execution.

The selected variables in the chromosome, result of the decision maker, can be observed on Figure 3. In general, the number of variables are lower in SPEA-II-MLR than NSGA-II-MLR. In spite of SPEA-II-MLR selected less variables, both algorithms cover the same spectral regions. This similarity indicates that these regions are the most promising to use in the spectrophotometer. In practice, this result implies a

smaller number of wavelengths measures in spectrophotometer for quantify the test weight property in real samples. For the other properties of interest the results for NSGA-II-MLR and SPEA-II-MLR are similar for those presented in Figures 2 and 3.

The solutions obtained by the decision maker in each of the executions will be used in the next paragraphs in order to calculate the RMSEP measure in the prediction set. Table III shows the resume of results of NSGA-II-MLR and SPEA-II-MLR. The results were obtained by 30 executions of each of theses algorithms for each property. Worth noting that results refer to solution selected by decision maker in each execution applied in the prediction set. The prediction set was not used at any stage of the proposed algorithms. This set is used to measure the generalization ability of the obtained solutions. As can be seem the NSGA-II-MLR and SPEA-II-MLR had a small difference in RMSEP average. For protein content and test weight NSGA-II-MLR has better RMSEP values than SPEA-II-MLR, but for PSI and Farinograph water absorption SPEA-II-MLR obtained better RMSEP values. However, for all the properties SPEA-II-MLR found solutions with a lower number of variables selected than NSGA-II-MLR.

Analyzing all results obtained by NSGA-II-MLR and SPEA-II-MLR algorithms we infer that SPEA-II-MLR has a best behavior. SPEA-II-MLR selects a fewest number of variables and it has a small difference in RMSEP of NSGA-II-MLR. The few number of variables is important in others applications of calibration problems where the spectroscopy measure can be expensive. In this cases the expert can choice a solution with a prediction error a little high but with few variables.

Now, we compared the results obtained by SPEA-II-MLR with the classical algorithms. First of all, we can see that the multi-objective formulation resolved the problem of excessive number of variables in the mono-objective approach. For example, in the test weight, while the MONO-GA-MLR
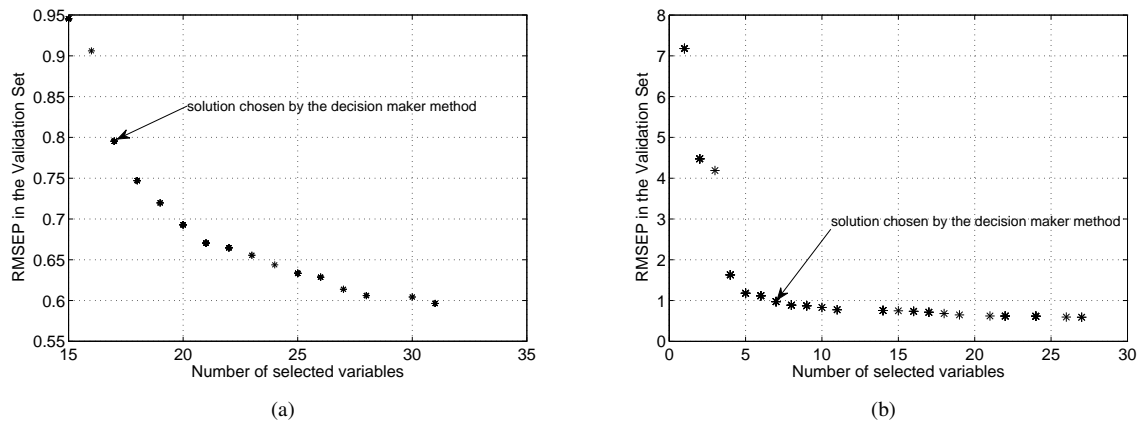
Fig. 2.   Pareto Front in the NSGA-II-MLR (a) and SPEA-II-MLR (b) Algorithm.
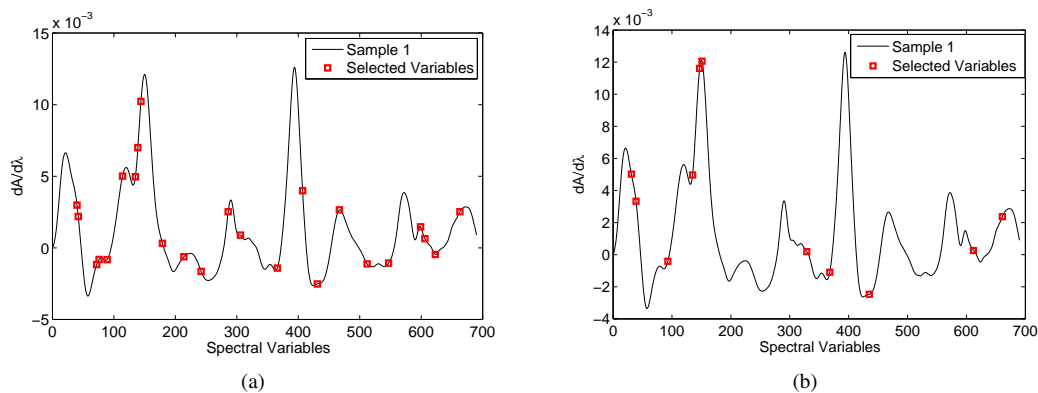
Fig. 3.   Variables Selected by NSGA-II-MLR (a) and SPEA-II-MLR (b) Algorithm.

selected 112 variables with a prediction error of 1.38, SPEA-II-MLR chose just 9 variables with a prediction error of 1.01. And for the PSI, MONO-GA-MLR selected 157 variables with RMSEP of 2.69 while, SPEA-II-MLR selects 9 with a prediction error of 2.19. In comparison with SPA-MLR, SPEA-II-MLR also use a less number of variables in average for all the properties of interest, SPA-MLR uses 13, 29, 36 and 18 variables while SPEA-II-MLR selects 10, 9, 9 and 5, respectively. The average RMSEP result of SPEA-II-MLR was 57% better than PLS and MONO-GA-MLR and 55% better than SPA-MLR. In the test weight property, SPEA-II-MLR was 15.8%, 17.8% and 26.8% better than SPA-MLR, PLS and MONO-GA-MLR respectively. In the PSI (wheat kernel texture) property, the improvement of SPEA-II-MLR in relation SPA-MLR, PLS and MONO-GA-MLR was 11.6%, 17.7% and 15.6% respectively. And finally, in the farinograph water absorption property SPEA-II-MLR was 1.8%, 2.3% and 2.3% better than SPA-MLR, PLS and MONO-GA-MLR respectively.

Figure 4 shows the result of prediction of test weight versus the real test weight by the solution of SPEA-II with less RMSEP value. In the ideal case the points are arranged on a straight line. As can be seem the predicted values are close of real values. In this sense the regression model obtained using the selected variables can be used in practice. The results for the other properties were similar.
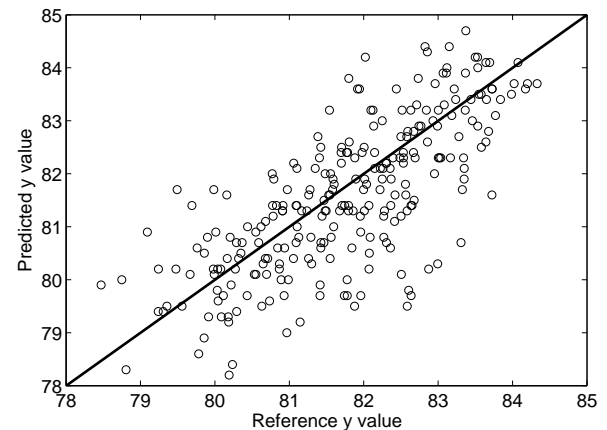
Fig. 4.   Comparison between real and predicted test weight by model builded by SPEA-II.

VI.   CONCLUSION

In this work we proposed a multi-objective formulation of variable selection problem in multiple determination problem of chemical properties using NSGA-II-MLR and SPEA-II-MLR algorithms. A case study based on chemical properties of wheat was presented. The results obtained showed that the

TABLE III.   RESULTS OF MULTI-OBJECTIVE TECHNIQUES
NSGA-II-MLR AND SPEA-II-MLR. THE RESULTS ARE EXPRESSED IN
RMSEP TERMS IN THE PREDICTION SET.

	NSGA-II-MLR	SPEA-II-MLR
Protein content (%)		
Average RMSEP	0.087	0.090
Maximum RMSEP	0.129	0.145
Minimum RMSEP	0.059	0.068
Average Number of Variables	19	10
Maximum Number of Variables	24	17
Minimum Number of Variables	12	7
test weight (Kg/Hl)		
Average RMSEP	0.76	1.01
Maximum RMSEP	0.88	1.13
Minimum RMSEP	0.70	0.89
Average Number of Variables	22	9
Maximum Number of Variables	30	10
Minimum Number of Variables	17	7
PSI (wheat kernel texture) (%)		
Average RMSEP	2.27	2.19
Maximum RMSEP	2.39	2.45
Minimum RMSEP	2.19	2.11
Average Number of Variables	19	9
Maximum Number of Variables	29	15
Minimum Number of Variables	14	5
Farinograph water absorption (%)		
Average RMSEP	2.17	2.10
Maximum RMSEP	2.39	2.35
Minimum RMSEP	2.09	2.08
Average Number of Variables	12	5
Maximum Number of Variables	16	7
Minimum Number of Variables	9	4

multi-objective formulation resolved the classical problem of
mono-objective. While mono-objective GA formulation use a
bigger number of variables with prediction error similar to
classical algorithms the multi-objective algorithms use fewer
variables with the less prediction error. The SPEA-II-MLR
improved the prediction error for all properties analyzed.

ACKNOWLEDGMENT

REFERENCES

[1] Masamoto Arakawa, Yosuke Yamashita, and Kimito Funatsu, *Genetic
algorithm-based wavelength selection method for spectral calibration*,
Journal of Chemometrics **25** (2011), no. 1, 10–19.

[2] Il-Gyo Chong and Chi-Hyuck Jun, *Performance of some variable
selection methods when multicollinearity is present*, Chemometrics and
Intelligent Laboratory Systems **78** (2005), no. 1-2, 103–112.

[3] J. Cortina, *Interaction, nonlinearity, and multicollinearity: Implications
for multiple regression*, Journal of Management **19** (1993), no. 4, 915–
922.

[4] Anderson da Silva Soares, Roberto Kawakami Harrop Galv ao, and
Mário César Ugulino Araújo, *Multi-core computation in chemometrics:
Case studies of voltammetric and nir spectrometric analyses.*, Journal
of the Brazilian Chemical Society **21** (2010), 1626–1634.

[5] Anderson da Silva Soares, Arlindo Rodrigues Galvo Filho, Roberto
Kawakami Harrop Galv ao, and Mário César Ugulino Araújo, *Improving
the computational efficiency of the successive projections algorithm by
using a sequential regression implementation: A case study involving
nir spectrometric analysis of wheat samples*, Journal of the Brazilian
Chemical Society **21** (2010), 760–763.

[6] Daniel Vitor de Lucena, Anderson da Silva Soares, Telma Woerle
de Lima, Alexandre Claudio Botazzo Delbem, Arlindo Rodrigues Gal-
vao Filho, and Clarimar Jose Coelho, *Multi-objective evolutionary
algorithm for variables selection in calibration problems: A case study
for protein concentration prediction*, Proceedings of IEEE Congress on
Evolutionary Computation (2013), 1123–1130.

[7] Kalyanmoy D. Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan,
A fast and elitist multiobjective genetic algorithm : NSGA-II, Evolu-
tionary Computation, IEEE Transactions on **6** (2002), no. 2, 182–197.

[8] Donald E. Farrar and Robert R. Glauber, *Multicollinearity in Regres-
sion Analysis: The Problem Revisited*, The Review of Economics and
Statistics **49** (1967), no. 1, 92–107.

[9] Arlindo R. Galvao Filho, Roberto K. H. Galvao, and Mario Cesar U.
Araujo, *Effect of the subsampling ratio in the application of subagging
for multivariate calibration with the successive projections algorithm*,
Journal of the Brazilian Chemical Society **22** (2011), 2225 – 2233.

[10] Richar H. Guy, Jurij J Hostynek, Robert S Hinz, and Cynthia R Lorence,
Metals and the skin, Marcel Dekker Incorporated, 1999.

[11] Roberto Kawakami Harrop Galvao, Mario Cesar Ugulino Araujo, Wal-
lace Duarte Fragoso, Edvan Cirino Silva, Gledson Emidio Jose, Sofacles
Figueredo Carreiro Soares, and Henrique Mohallem Paiva, *A variable
elimination method to improve the parsimony of mlr models using
the successive projections algorithm*, Chemometrics and intelligent
laboratory systems **92** (2008), no. 1, 83–91.

[12] J. L. Hodges, P. H. Ramsey, and S. Wechsler, *Improved Significance
Probabilities of the Wilcoxon Test*, Journal of Educational and Behav-
ioral Statistics **15** (1990), no. 3, 249–265.

[13] R.W.; Stone L. A. Kennard, *Computer aided design of experiments*,
Technometrics **11** (1969), 137.

[14] Alan J. Miller, *Selection of Subsets of Regression Variables*, Journal of
the Royal Statistical Society. Series A (General) **147** (1984), no. 3.

[15] Edilene D. Moreira, Márcio J. Pontes, Roberto K. Galvão, and
Mário C. Araújo, *Near infrared reflectance spectrometry classification
of cigarettes using the successive projections algorithm for variable
selection*, Talanta **79** (2009), no. 5, 1260–1264.

[16] T. Naes and B. H. Mevik, *Understanding the collinearity problem
in regression and discriminant analysis*, Journal of Chemometrics **15**
(2001), no. 4, 413–426.

[17] Miho Shimada, Yuji Masuda, Yukio Yamada, Masahide Itoh, Motoji
Takahashi, and Toyohiko Yatagai, *Explanation of human skin color by
multiple linear regression analysis based on the modified lambert-beer
law*, Optical Review **7** (2000), no. 4, 348–352.

[18] D. A. Skoog, *Princpios de anlise instrumental*, Bookman, 2002.

[19] Sófacles F. Soares, Adriano A. Gomes, Mario C. Araujo, Arlindo R. G.
Filho, and Roberto K. H. Galvao, *The successive projections algorithm*,
TrAC Trends in Analytical Chemistry **42** (2013), 8498.

[20] David J Thornley, *Anisotropic multidimensional savitzky golay kernels
for smoothing, differentiation and reconstruction*.

[21] Araujo M.C.U.; Saldanha T.C.B.; Galvao R.K.H.; Yoneyama T.; Chame
H.C.; Visani V, *The successive projections algorithm for variable
selection in spectroscopic multicomponent analysis*, Chemometrics and
Intelligent Laboratory Systems **57** (2001), no. 2, 65–73.

[22] Eckart Zitzler, Marco Laumanns, and Lothar Thiele, *Spea2: Improving
the strength pareto evolutionary algorithm*, Tech. report, 2001.

Artificial systems evolutionary learning

I.Tregubenko[1]

[1]Faculty information technology and systems, Cherkasy State Technological University, Cherkasy, Ukraine

Abstract - *The idea and technology of artificial systems evolutionary learning is developed. The concept of intellect is dissociated from physical realization of thinking processes. New understanding of learning process discrecity taking into account the artificial system life cycle limitedness principle is formulated. The two-module concept of artificial system modeling is proposed. It consists of two connected modules which have different properties and life cycle. New concept of B-System is introduced. The artificial systems evolutionary learning task formalization on the base of resource approach is executed. The principles formulated allow to simplify technologically the artificial systems learning processes, to decrease the "artificial system-learning environment" pair connections and interaction factors complexity.*

Keywords: Intellectual artificial systems, Evolutionary learning, B-System.

1   Scientific issue analysis and formulation

The modern evolutionary artificial intelligence techniques are traditionally based on physical laws and known world part organization methods. The scientists concentrate their efforts on researching the physical parts of matter, their connections and transitions. While modelling evolution processes in the variety of aspects, for instance, using cognitive approach [1,2], the researches are grounded on study and modelling the real physical, physiological processes which are in natural systems. The thinking processes concerning the intellectual systems are considered as the possibility to use the knowledge and deductions based on the knowledge for effective and rational system behaviour implementation [3]. Information analysis, systematization, and storing are not detached from system physical platform. There are many theories and techniques for thinking processes in natural systems reproduction [4-6], for example, neurons and neuron systems modelling [7-10]. But full-fledged artificial intelligence creation issue is not solved yet.

For creation the artificial systems with intellectual features the most substantial and problematic are the processes of the intellectual parts learning, specifically control modules [11]. For designing this kind of modules intellectual agents technologies can be used. But for their adaptation in an environment the reinforcement learning technique is considered as the most effective and reliable

[12]. It is based on reinforcement learning theory [13] which can be considered as adaptive behavior theory development. But there are many reasonable obstacles because of the learning task complexity and multiple criteria existence.

2   Objectives

For the further development of artificial intelligence creation task solution methods the new conception and technology for artificial systems evolutionary learning is needed. Also, it is necessary to formulate the principles for artificial systems learning process technological simplification, connections and interaction in "artificial system - learning environment" pair operating factors complexity reduction, and artificial system appropriate model creation.

3   Artificial systems evolutionary development

Considering this issue the word "intellectual" is not used deliberately concerning to the systems in general and artificial systems specifically. Any system is considered as intellectual with possible margin states with no intellect (for example, in the beginning and the end of the system's life cycle). And any system is considered as origin independent; natural systems (NS) are different from artificial systems only with author (creator) attribute.

Looking rigorously on natural systems, especially systems in evolutionary development, one can notice some common principles: development is where some division and tension because of incompleteness (imperfection) exist; the systems without indivisibility, self-sufficiency, functional completeness evolve.

So, there are two most important postulates:

1. Evolutionary development is possible only with interaction.
2. Interaction is possible only with incomplete system form.

Further, with more attentive world consideration (at least, of human perception perspective) it is seen that any property has its opposition. So, one can guess that the world is based on binary organization, and the matter is only in detalization degree. (For example, there is no discussion that OS Windows is a set of binary codes in its foundation). And

almost always the world can be decomposed to some binary (or couple) component system. The components can be any characteristics, fact, concepts, properties, parts etc. These interesting facts facilitate view that binary principle with algorithmic approaches can be used for artificial evolution description. In this case a vast and very interesting research field is being opened.

But for the beginning let's consider one aspect development. Let's take any intellectual system binary model a artificial evolution basement. Let's consider any intellectual system as *B-System* [14] with carrier and intellectual part separated.

Let's consider the classical spiral evolutionary development idea (figure 1.a). We understand the world change as going from simple to complex and this reflects evolution spiral. Evolution spiral shows world structure complication the most complete way but isn't a definite mirror for direction (technologies) and development linearity (an individual, some intellectual object evolution linearity). An artificial system development is a non-linear process. While developing a system can move step by step on evolution spiral (figure 1.b) but also can move back to some state for repeated study. If intellectual part is saved in this process there is almost eternal artificial intelligence development process.
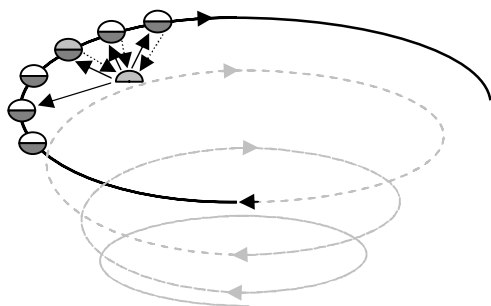
a) evolutionary spiral;
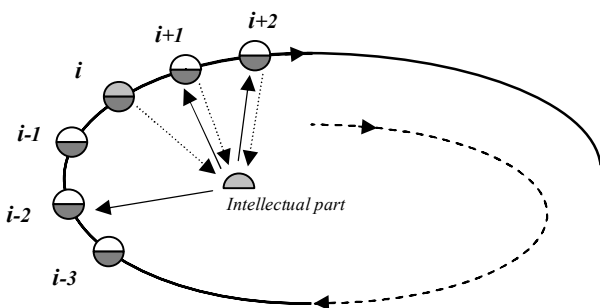
b) artificial system evolutionary development mechanism.

Figure. 1. An artificial system evolutionary development.

Then it's necessary to explain our understanding of the world as a discrete system. The continuity and discrecity concepts are relative and subjective. So, defining the process as continuous or discrete is relative and depends on the observer. One more important factor influencing artificial systems development process is understanding the aggregativeness of knowledge and skills gained during learning.

So, the following artificial systems evolution main features can be selected: discrecity, aggregativeness, and non-linearity.

Let's consider an artificial system development in more detail. Let's accept that artificial system development is possible only as a result of learning or self-learning. The main challenge for effective learning organization is the complexity of the correct statement of learning task.

There is evident correlation between the time needed for the system learning necessary for its stable state and functioning self-sufficiency on one hand and system complexity on the other hand. The more complex systems need more time for learning than simpler systems. Besides, domain also influences learning time specifically the functions the artificial system must execute after learning. There is no sense to consider learning out of functions the system must implement context. The learning task can be considered only in the context of functions, which the system under learning must be able to run, set limitations.

Taking into account that we have not the abstract learning task without any limitations there is a challenge in formulating the learning objective and criteria system creation (the criteria must confirm learning effectiveness). Besides, it is necessary to take into account that so-called self-learning systems are senseless and can be dangerous if learning is not under control of external environment or learning subject.

4   Artificial systems evolutionary learning subjects

There are three parts of artificial systems learning process (figure 2): object, subject, and learning environment. A learning object is an artificial system (AS) which has to study, specifically gain some knowledge and some set of skills. A subject is a system author (A). It's not possible to set objectives and initiate the learning process itself without understanding the author participation and functions. Of course, the author may have a complex structure, be a separate individual or a union, both natural or artificial, what is not so important.

The third part of the process under consideration is learning environment (C). Depending on author and learning environment correlation there are three different in foundation types of an artificial system learning organization

(figure 1). Case a) in figure 1 demonstrates the situation where the learning environment is detached from the author. Case b) in figure 1 demonstrates the situation where learning environment includes the author in whole or partially. Case c) in figure 1 presents the situation where the author is identical to learning environment, actually, the learning environment is the author. The last case matches artificial system self-learning.

a)   b)   c)
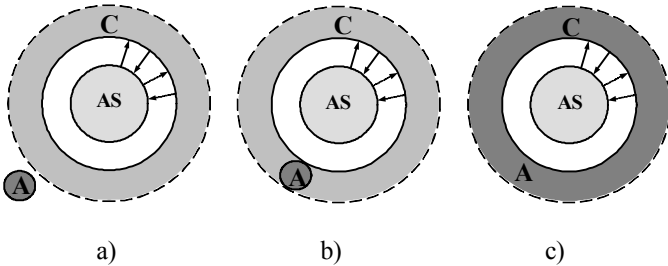
Figure 2. Artificial system learning organization types: a) – exterior independent author; b) – author as learning environment type; c) – author is identical to learning environment.

In the field of information technologies and their realization in programs the technological details of the platform organization can be different for the same information system. It is evident that the same algorithm can exist in different forms which are different by syntax and technology. But with this the algorithm actually doesn't change. The same way let's detach the intellect concept from system under learning implementation.

5   Evolutionary learning concept

Let's consider in more detail the artificial systems evolutionary learning idea. The modern artificial systems learning methods are various but usually connected with physical characteristics, structure, and learning object and its environment properties. Let's abstract from technical platform and specific artificial system implementation. For making the task simpler let's ignore the inner structure, specific design at the first phase. The task is to describe artificial systems learning technology as an abstraction.

Let's formulate the conception of artificial systems evolutionary learning [15]. The conception is based on the following principles:

1. Learning limitedness.
2. Life cycle finiteness.
3. Learning iteration-hierarchical structurization.
4. Reboot availability.
5. Evolutionary system life cycle specificity.
6. System modularity.

5.1   Learning limitedness

For limited in time life cycle the system can learn some limited functions set and acquire some limited knowledge. Looking at the natural world we can deduce that there is no universal intellectual system. Every intellectual system has consciousness of some level and can perform some limited functions set. The life cycle of any system is limited and finite. Certainly, during some finite life cycle an artificial system can learn some limited functions set and acquire some limited knowledge. So, it is reasonable to consider increasing an artificial system life cycle making it conditionally infinite.

5.2   Life cycle finiteness

It's not possible to create a system with one infinite life cycle. So, conditionally infinite existence cannot be provided on the base of limited physical reality. Let's accept apriori that for one artificial system existence life cycle it can learn some limited set of functions and skills. Then for learning proceeding the system must get new data and task – the new set of functions to master. So, artificial system learning task comes to interactive learning process creation and the process goes on different hierarchical levels of functioning taking into account the necessity to preserve the part of knowledge and ability to accomplish some functions while going to a new hierarchical level. In this process the system also loses some features and capabilities (technological interface functions) which are not necessary at the next level but were used before get knowledge and skills and gets some new features and capabilities (technological functions) and initial data allowing to operate new skills in new environment.

It's not possible to create a system with one infinite life cycle. So, conditionally infinite existence cannot be provided on the base of limited physical reality. Let's accept apriori that for one artificial system existence life cycle it can learn some limited set of functions and skills. Then for learning proceeding the system must get new data and task – the new set of functions to master. So, artificial system learning task comes to interactive learning process creation and the process goes on different hierarchical levels of functioning taking into account the necessity to preserve the part of knowledge and ability to accomplish some functions while going to a new hierarchical level. In this process the system also loses some features and capabilities (technological interface functions) which are not necessary at the next level but were used before get knowledge and skills and gets some new features and capabilities (technological functions) and initial data allowing to operate new skills in new environment.

5.3 Learning process iteration-hierarchical structurization

Taking into account mentioned above let's give the learning process complexity simplification. System evolutionary learning process (figure 3) is divided into some hierarchical iteration processes (stages). At the stages some limited learning will be. It will be connected with some concrete environment physical parameters and limited learning tasks. It will cause the finiteness of system life cycle at the current iteration-hierarchical level.
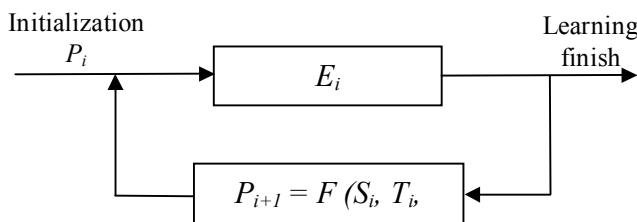
Рис.3. Evolutionary learning stage number i

5.4 "Rebooting" mechanism

Let's introduce "reboot" mechanism concept [15]. After the current system life cycle finish system preparation and transition to the next evolutionary learning stage is provided. During that learning results are saved, the knowledge gained are aggregated, the structure and volume of knowledge gained are optimized, learning results are analyzed for matching the objectives set, the decision of transition to the next learning level (with the possibility to interactively model the level) is made, learning and forming the current technologies of adaptation to the new physical environment tasks are set.

5.5 Artificial system life cycle

Let's accept the following chain as a life cycle:

initialization (creation) – learning – end(destruction) – rebooting

During first initialization procedure (figure 3) the first evolutionary stage modeling is executed, the learning tasks are defined, the current technologies of adaptation to physical environment are formed. Then learning process is run; in general case the i number learning evolutionary stage (E_i) is executed. At this stage the technologies of reinforcement learning are quite acceptable [12]. After the current evolutionary learning stage number i (E_i) three main "reboot" mechanizm procedures should be executed:

1. Saving the artificial system learning results (S_i) – optimization and safety of data received as the learning results in form independent on the environment and physical form.

2. Ignoring non-necessary technical data and service functions (T_i) which were used for the system adaptation to the concrete physical realization and concrete tasks of the concrete learning stage.

3. Analysis (A_i) that is the assessment of matching the learning results to the objectives set, making decision on moving to the next learning stage with interactive modeling possibility, setting new learning tasks, and forming the current technologies of adaptation to the new physical environment.

It's worth noting that both new ($i+1$) and current (i) evolutionary stage can be selected as the next level (the last is equal to the repeated learning). In a general case an artificial system learning next stage any hierarchical level can be selected.

5.6 System modularity

Let's create an artificial system model considering the evolutionary learning technologies mentioned above. The artificial system is divided into two components. One component includes the intellectual part and is persisted as long as necessary. The other component must be flexible enough and provide system adaptation the environment being changed. This component is not always stable and often is variable, with short existence time. The evolutionary learning system model is represented in two significantly different modules: intellectual module and environment adaptation module (figure 4).
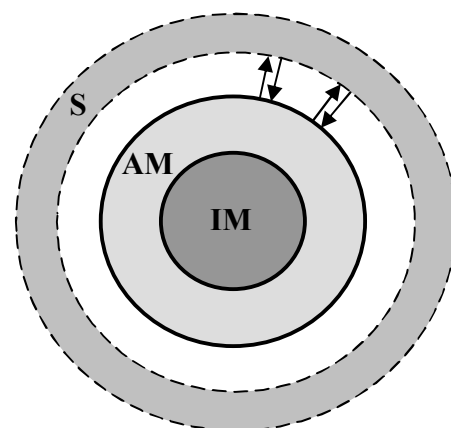
Рис.4. An artificial system module structure.

Intellectual module (IM) is considered as a constant component and is the gist of the system, and actually we can consider it as the system itself. Its' existence time is unlimited. Adaptation module (AM) is a variable component

and provides intellectual module functioning in the current physical conditions. This module must ensure the intellectual part adaptation to the environment current conditions, to the current learning tasks, and artificial system knowledge acquisition stages. Actually, this module is the interface between the intellect and the environment and provides the intellectual module functioning in concrete physical realities. Going through evolutionary stages the adaptation module persistence is not stipulated.
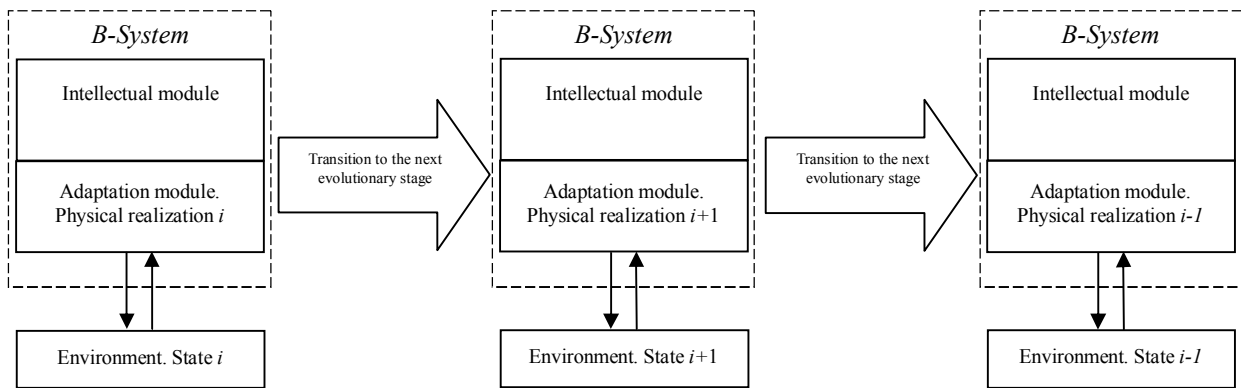
Рис. 5. B-System adaptation mechanism to new environment physical parameters

This view on an artificial system provides systems adaptability to new environment physical parameters (figure 6) in time of transition to the new iteration evolutionary stage preserving the intellectual part. So, it allows to provide evolution process beginning from the elementary levels and to provide knowledge accumulation. It is evident that in this view on a system non-linear evolutionary process can be easily realized.
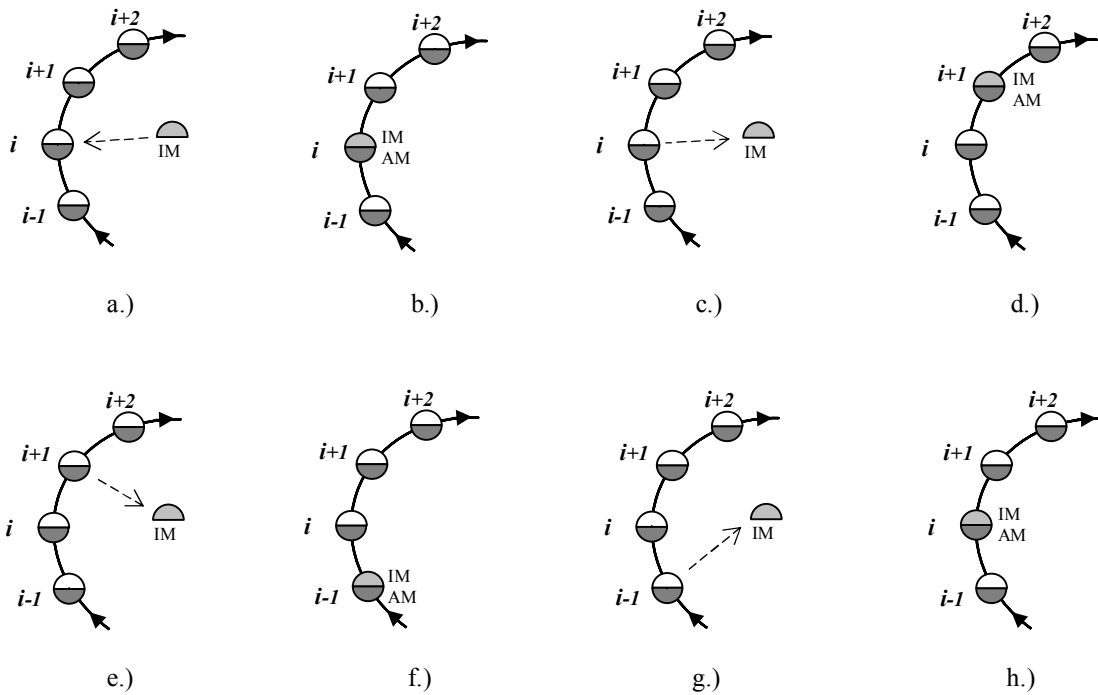
Рис. 6. B-System in the process of adaptation to the new environment parameters taking into account the evolution processes non-linearity

6   Learning task formalization. Resource approach

Let's make artificial system learning task formalization at iteration-hierarchical learning process evolution stage number i.

At initialization time at evolution stage i the learning tasks discretization is executed. The finite set of functions which the artificial system must master is determined. So, at the evolutionary stage i the artificial system must implement M learning processes. Some set of resources R^i is given to the artificial system for executing every process at evolutionary learning stage i:

$$R^i = \sum_{k=1}^{M} r_k^i \qquad (1)$$

For one k number process execution it is necessary to master some limited resource set N_k^i. It is possible to define the executed learning process condition using full mastering the existing resource R^i condition at i evolutionary learning stage:

$$\sum_{k=1}^{M} \sum_{j=1}^{N_k^i} r_{kj}^i N_k^i = R^i \qquad (2)$$

where r_{jk}^i - the amount of j resource processed with k learning process.

The objective function of learning at i evolutionary stage can be represented as:

$$F^i = \sum_{k=1}^{M} \sum_{j=1}^{N_k^i} z_{kj}^i N_k^i + \sum_{k=1}^{M} \varphi_k^i (N_k^i, N_k''^i) \qquad (3)$$

where z_{kj}^i -expenditures of resource j in process k at artificial system learning evolutionary stage i.

It's evident that N_k^i is always less than $N_k''^i$ - the maximum of possible number of an artificial system learning processes realized, that is resources consumed at evolutionary stage i.

φ_k^i is a fine function for the executed learning processes number deviation from maximal number of executed learning processes at evolutionary stage i.

So, an artificial system learning task at evolutionary stage i can be formulated the following way:

to minimize the objective function (3) observing the limitation (2).

7   Conclusion

Artificial systems learning organization on the evolutionary techniques base can increase learning effectiveness. The idea and technology of artificial systems evolutionary learning is developed. New understanding of learning process discrecity taking into account the artificial system life cycle limitedness principle is formulated. The two-module concept of artificial system modeling is proposed. It consists of two connected modules which have different properties and life cycle. New concept of B-System is introduced. The artificial systems evolutionary learning task formalization on the base of resource approach is executed. The principles formulated allow to simplify technologically the artificial systems learning processes, to decrease the "artificial system-learning environment" pair connections and interaction factors complexity. It allows to develop artificial intellectual systems evolutionary learning building and organization methods in future.

8   References

[1]   Red'ko V.G. , Anokhin K.V., Burtsev M.S., Manolov A.I., Mosalov O.P., Nepomnyashchikh V.A., Prokhorov D.V. Project "Animat Brain": Designing the Animat Control System on the Basis of the Functional Systems Theory // In Butz, M.V., Sigaud, O., Pezzulo, G., & Baldassarre, G. (Eds.), Anticipatory Behavior in Adaptive Learning Systems: From Brains to Individual and Social Behavior. LNAI 4520, Berlin Heidelberg: Springer Verlag. 2007. PP. 94-107.

[2]   Burtsev M.S. Adaptive learning through variation and selection // In Seel, Norbert M. (Ed.) Encyclopedia of the Sciences of Learning, т.1, pp. 116-118. Springer, 2012.

[3]   Tregubenko I.B. Adaptive behaviour of intellectual agents of protection on the base of theory of cognitive evolution // XVII internatinal conference on automatik control "AUTOMATICS – 2010". Vol 2. – Kharkov: KNURE, 2010. – 306 p. – PP.202-203

[4]   Victor M. Kureichik. Genetic algorithms and their application. Scientific edition. TSURE publishing, second and enlarged edition, 2002. – 242 p.

[5]   Hingston, Philip; Barone, Luigi; Michalewicz, Zbigniew (2008). Design by Evolution: Advances in Evolutionary Design. Springer. ISBN 978-3540741091.

[6]   L.A. Gladkov, Victor M. Kureichik, V.V. Kureichik. Genetic Algorithms. Moscow, Physmatlit, 2010. – 317 p. ISBN: 9785922105101

[7]   Bhadeshia H. K. D. H. (1999). "Neural Networks in Materials Science". ISIJ International 39 (10): 966–979. doi:10.2355/isijinternational.39.966

[8] Lakhmi C. Jain; N.M. Martin Fusion of Neural Networks, Fuzzy Systems and Genetic Algorithms: Industrial Applications. — CRC Press, CRC Press LLC, 1998

[9] Red'ko V.G. The Natural Way to Artificial Intelligence // In B. Goertzel, C. Pennachin (Eds.), Artificial General Intelligence. Springer. Berlin, Heidelberg, New York. 2007. PP. 327-351.

[10] Ripley, B. D. (1996) Pattern Recognition and Neural Networks. Cambridge University Press. 403 pages. ISBN 0-521-46086-7

[11] Tregubenko I.B. Concept of intellectual control in complex distributed systems. // Intellectual analysis of information. 13th international scientific conference named after prof. T.A.Taran IAI-2013. Kyiv, Ukraine, May 15 - 17, 2013 – K: PROSVITA, 2009. – 464 p. – PP. 391-393

[12] Tregubenko I.B. Reinforcement learning intellectual agent of protection for adapting to surrounding environment // SIN '10: Proceedings of the 3rd international conference on Security of information and networks/ - ACM New York, NY, USA ©2010 - pp: 110-112. ISBN: 978-1-4503-0234-0 DOI=10.1145/1854099.1854122
http://dl.acm.org/citation.cfm?id=1854099.1854122&coll=DL&dl=GUIDE&CFID=87567568&CFTOKEN=25740824

[13] Richard S. Sutton, Andrew G. Barto. Introduction to Reinforcement Learning,1st edition, MIT Press, Cambridge, MA, USA ©1998 P. 342. ISBN: 0262193981 http://dl.acm.org/citation.cfm?id=551283&CFID=87567568&CFTOKEN=25740824

[14] Tregubenko I.B. Systems conceptual model in evolutionary learning theories. // International Scientific-Practical Conference "Information Technologies in Education, Science and Technique" (ITEST-2012), Cherkasy, Ukraine, April 25-27th, 2012. – Cherkasy: CHDTU, 2012. – V.1. – 195 p. – PP. 54-55.

[15] Tregubenko I.B. Concept of artificial systems evolutionary learning. // System analysis and information technologies: 14-th International conference SAIT 2012, Kyiv, Ukraine, April 24, 2012. Proceedings. ESC \ IASA "NTUU \KPI", 2012. – 443 p. – PP. 245

A Solution for Forecasting PET Chips Prices for both Short-Term and Long-Term Price Forecasting, Using Genetic Programming

Mojtaba Sedigh Fazli[1], Jean-Fabrice LEBRATY[2]

ABSTRACT

Nowadays, forecasting on what will happen in economic environments plays a crucial role for managers to invest correctly on appropriate items. We showed that in PET market how a neuro-fuzzy hybrid model can assist the managers in decision-making [13]. In this research, the target is to forecast the same item through another intelligent tool which obeys the evolutionary processing mechanisms. Again, the item for prediction here is PET (Poly Ethylene Terephthalate) which is the raw material for textile industries and it is highly sensitive against oil price fluctuations and also some other factors such as the demand and supply ratio. The main idea is coming through AHIS model which was presented by M.S. Fazli and J.F. Lebraty in 2013 [13]. In this communication, the hybrid module is substituted with genetic programming. Finally, the simulation has been conducted and compared to three different models answers which were presented before. The results show that Genetic programming results (acting like hybrid model) which support both Fuzzy Systems and Neural Networks satisfy the research question considerably.

KEYWORDS:

Efficient Market Hypothesis, Financial Forecasting, Chemicals, Artificial Intelligence, Genetic Programming, Decision Support System, Hybrid Neuro Fuzzy Model.

1. INTRODUCTION

Innovation of Artificial Intelligence opened new horizons to Financial Forecasting issues. Unfortunately, there are a lot of financial managers who do not believe in forecasting but the method of AI tools which follow and predict the time series trends is still a hot issue in management and mathematics. We suppose that we can capitalize on the previous work in order to provide current decision maker in a specific field with an adapted decision support system. In this paper we want to answer the following research question *"How to forecast PET chips prices in short-time and long-time?"*

---

[1]*Doctorate Student , University of Montesquieu Bordeaux 4 , Mojtabafazli@yahoo.com*
[2] *Full Professor , Lyon3 University – IAE Business school – Lab. Magellan EA3713 , jean-fabrice.lebraty@univ-lyon3.fr*

1.1 Why the AI methods are appropriate for this issue?

To handle this project, there are 2 major categories: one uses traditional methods, in this category there are 2 major methods named Fundamental Analysis and Technical Analysis. The second solution is to use novel tools such as AI tools. Due to the nature of price trends in stock markets which follow a chaotic process [1], the research seems to be compatible drawing on AI tools. A chaotic system includes two different parts: one is stochastic and another part is deterministic. When the market trend is not too noisy, the deterministic part will be more than 50%, in this case for the remaining part, obviously there are a lot of parameters which affect the price direction and fluctuations. Because of the variety of factors which control and affect the curve, it's considered that this part is stochastic and random. Here, our aim is to find a formula which will be able to determine the next day's prices.

The 70s decade was a start point for mathematicians in terms of applying the new mathematics, time series and even some advanced tools, such as Artificial Intelligence, to verify the forecasting ability of stock and other market prices. Today, the prices of chemicals which are used as raw materials in lots of industries usually are determined in stock exchange markets, or they directly depend on some other prices, which are determined in stocks such as oil price, exchange rate etc. Researchers have done a lot of tests and experiments on price information and stock exchange index in some countries such as USA, UK, Canada, Germany, Japan Turkey, India and etc.[5-8], to find existence or non-existence of defined structure in stock price information. At that time, the most important thing for researchers was to reject the Random Walk Hypothesis [2]. Stock markets are affected and surrounded by lots of extremely interrelated parameters such as economic, social, political and even psychological indicators [3]. These mentioned indicators interact with each other in a sophisticated manner; therefore it is normally very difficult and even some times impossible to forecast the fluctuations of price trends in stock markets.

There are lots of forecasting tools which are applied to this field in both traditional and modern techniques [4-7]. With the development of artificial intelligence, researchers and investors hope that the market complexities can be untied. Previously in 90s, there was

a research conducted by Johnson and his colleagues [8] who identified a lot of potential uses for neural networks in financial institutions, corporate finance and investments. Over the last 20 years, the applications of AI tools in financial solutions have been increased dramatically.

2.   LITERATURE REVIEW

In this section a review about the components of our research question will be presented. Firstly, the notion of price behavior on a chaotic market will be explained and as a consequence, a review of the main AI models and proposed tool which are currently possible to apply for this problem are discussed.

2.1 Efficient Market Assumption and Chaos Theory

Price behavior (especially stock price) is a challenging issue which researchers have always faced [9]. The main challenge is whether market price behaviors are predictable or not. Some researchers believe that prices do not follow a specific trend, rather act in a "random walk" and cannot be predicted at all [3]. They are mostly advocates of a hypothesis called "The Efficient Market Hypothesis (EMH)". It has been proposed in the Efficient Market Hypothesis that in an efficient market the opportunities for profit are discovered so quickly that they seem to be opportunities [8]. Therefore there are no advantages of exclusivity and thus negating its potential performance. There has been a sense of doubt and uncertainty about the validity of the EMH, and some researchers attempted to use neural networks and other intelligent tools to validate their claims [2].

Markets are, in general, chaotic and usually the market curve follows chaos attitudes. A modern approach to modeling nonlinear dynamic systems like the market price trend which is fully relevant is named "*Chaos Theory*". Chaos theory considers a process under the assumption that "part of the process is deterministic and another part of the process is stochastic" [1]. Chaos is a nonlinear process which appears to be random. Various theoretical tests have been developed to test if a system is chaotic (has chaos in its time series). The deterministic part can be characterized using regression fitting, while the random process can be characterized by statistical parameters of a distribution function [7].

2.2 Genetic Programming [10-12]

It is an evolutionary method used mostly for optimization problems and works based on genetic operations. Genetic programming is a little different from genetic algorithms, the task and the aim of GP is to be able to reproduce computer programs. Genetic Programming follows Darwin's theory of evolution and his famous phrase "survival of the fittest". There is a population of individuals who marry each other and reproduce the new generation. After passing the time and reproduction cycles, the produced items try to survive and just the best and the fittest one will survive [10, 11].

2.1.1 Generating a Random Population

According to Koza [11], there are three techniques to generate the random population called: Grow, *Full* and *Ramped-half and-half*. Here the third method is selected.

2.1.1.1 The Genetic Operations

The evolutionary process will start by applying fitness test to all the individuals in the initial random population. The new population is formed by applying three main methods: *reproduction, mutation* and *crossover*. After completing the new population (i.e. the same size as the old) the old population will be eliminated.

2.1.1.2 Mutation

This process is applied on one individual. It happens when the new generation faces a deadlock, and after applying all other operations, the fitness function does not achieve better result.

2.1.1.3 Reproduction

Reproduction is where a selected individual copies itself into a new population. It effectively works the same as surviving an individual into the next generation. According to Koza [12], normally 10% of the population is selected for reproduction.

2.1.1.4 Crossover

Crossover requires two individuals and generating two different individuals for the new population. Figure 3-11 describes the Cross over process. Koza uses crossover on 90% of the population. The crossover plays the most important role in this process, since it generates the source of new individuals. There are a few other evolutionary operations: *editing*, *mutation*, *permutation, encapsulation…*
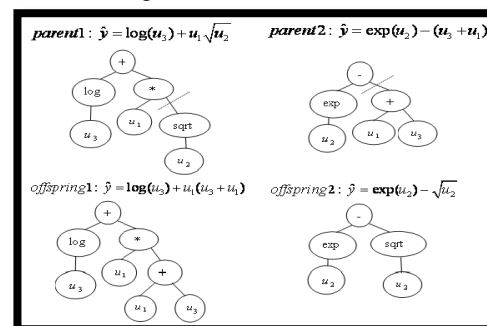
Figure 2-1: crossover genetic operation.

These methods would result in only a random search if it were not for the selection function.

2.1.2 Fitness-Proportionate Selection [9-11]

There is an algorithm which calculates the probability of selection, based on the nature of genetic programming and the definitions of this subject; it seems that the best individual of a population will be selected more frequently than the worst. The probability of selection is calculated with the following algorithm:

1. There is a raw fitness which will be restated in terms of *standardized fitness*. A lower value of standardized fitness denotes a better individual. By decreasing the individuals improvement through raw fitness, the standardize fitness will be equal to individual 'raw fitness. If the raw fitness decreases as an individual improves, standardized fitness for an individual is equal to the individual's raw fitness. And in case of increasing, an individual's standardized fitness is the maximum raw fitness minus the individual's raw fitness.

2. Standardized fitness is then reiterated as *adjusted fitness*, where a higher value indicates better fitness. The formula used for this is:

$$adj(i) = \frac{1}{1 + std(i)}$$

(1)

Where *adj*(i) is the adjusted fitness and *std*(i) is the standardized fitness for individual *i*. The application of using this adjustment is due to its benefits for separating individuals who have the near zero value of standardized fitness.

3. *Normalized fitness* is the form used by both selection methods. It is calculated from adjusted fitness in the following manner:

$$norm(i) = \frac{adj(i)}{\sum_{k=1}^{M} adj(k)}$$

(2)

Where *norm* (i) is the normalized fitness for individual *i*, and *M* is the number of individuals in the population.

4. The probability of selection (*sp*) is:

$$sp(i) = \frac{norm(i)}{\sum_{k=1}^{M} norm(k)}$$

(3)

This can be implemented by [10]:
(a) Order the individuals in a population by their normalized fitness.
(b) Chose a random number, r, from zero to one.
(c) From the top of the list, loop through every individual keeping a total of their normalized fitness values. As soon as this total exceeds stop the loop and select the current individual.

2.3 AHIS Model [13]

The model used in this research previously was coined by us as AHIS. AHIS is an approach obtained from NORN which is presented by Ted Lee and colleagues on 2001[1]. But, finally the model is different from NORN due to some modifications which are applied for gaining more advantages and changes in this specific application. Moreover, some parts of that model are eliminated and we called it AHIS which stands for Adaptive Hybrid Intelligent System. This system makes the prediction stronger and more accurate in this specific application [13, 9].
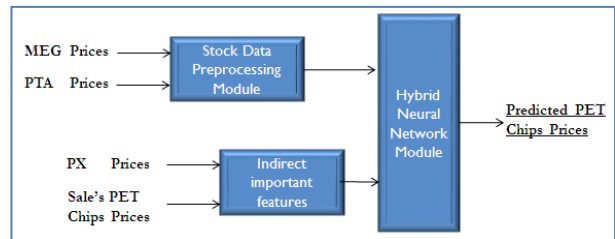
Fig 2 – 1: AHIS Model

As it was mentioned above there are 3 modules in AHIS model, in stock data preprocessing module, some preprocesses which are needed to be done on raw data take place. It is generally done for normalizing the data. Also, there is a module for applying indirect important features, which is normally effective in Neural Network methods, but here it's meaningless. In AHIS model PX price is applied as one of the most important features which indirectly affect PET chips prices. Another selected feature is sold PET chips prices, which is aimed to consider order and demand factors in the network. Since the cost price is produced by combining PTA and MEG in the first module through applying a specific formula. This formula is illustrated in Figure 2.2 [9]

Fig. 2 -2: Relationship between PX, PTA, MEG and PET chips

Finally, the last module is the most vital part of this model, where we previously tested pure neural network methods, such as a recurrent NN, MLP, TDNN, RNN, NARX. Besides that, some Neuro-Fuzzy models like ANFIS and LoLiMoT have been tested in our previous work. Here we changed our intelligent tool and used the Genetic programming approach instead of a Hybrid neural networks.

3.   THE METHODOLOGY

3.1 Data

Input data are historical data of PTA, MEG. They are gathered through 2 reputed sources: one is ICIS which is well-known in statistics and the analysis of chemical market and another one in RECRON Company in Malaysia which is the biggest supplier of yarn in Asia. This issue is a big challenge in Asian yarn suppliers. The data set includes 347 price samples which are classified in 2 sub sets: one subset includes 247 samples which are used in training process and the remaining 100 are used in testing process for 1 step prediction. By increasing the prediction steps to 10 and 15 days, the training set size is increased and the test set is decreased. Random generation process follows Ramped half and half method [9].

3.2 Desired Prediction results criteria

Here there is a need to determine the acceptable error. In order to find a good idea in this issue, some in depth interviews have been done with experts in this field among East Asian chemical managers. Based on those interviews, the fitness factor and criteria could be explained as follows: If the error value which is the difference between real value and predicted value is lower than 80 USD/Ton the result is acceptable and fewer than 50 USD/Ton is desired. It means that such a difference is not very crucial on this market and will not have a big effect on the next item which will be produced from PET chips:

$Err = |Fv - Rv| < 50$ *Desired* (4)

$Err = |Fv - Rv| < 80$ *Acceptable* (5)

Err = Error Ratio, Fv = Forecasted Value, Rv = Real Value

3.3 Model

In G.P. based solution, regarding the suitable characteristics of Genetic Programming and the aim of this research, it seems that it's possible to introduce a good and appropriate model for forecasting the mentioned items. However, simulating through Genetic Programming is a time-consuming process, it has been done in this research and the results were considerable .For running the simulation, the collected data are prepared in 15 days windowing .The data set is divided into two sets: first set (80% of total data) is used for training process and the second part (including 20% of total data) is used as test data. By considering the definition of the problem and our characteristic space of genetic programming, our research model will be changed to the following Model (See Fig. 3-1)
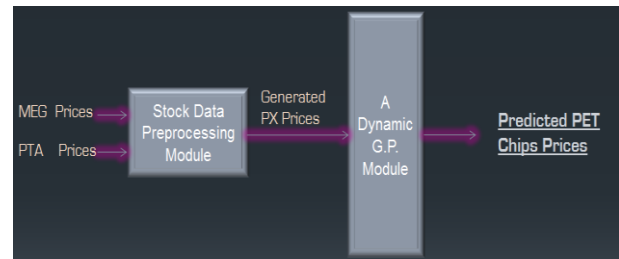
Figure 3-1: the research model is AHIS which is changed and modified to use for simulation of G.P.

As it is obvious in the above figure, the main module here is a dynamic G.P. module, running the G.P. in dynamic situation; it means that genetic operators are not fixed. In next step the trained data are used for modeling and the test one is used to check the validation accuracy, the used function set in simulation is as follows:

Function Set$=(X^2, X^3, X^4, X^5, X^6, X^y, e^x, \log(x), \ln(x), \sqrt{x}, \sin(x), \cos(x), +, -, *, /)$

For terminal set also following set is already used in this research:

Terminal Set = (rand , 1, 2, 3, 4, 5, 6, 7, 8, 9,10 ,11 ,12 ,13 ,14 ,15 ,16)

Here our logic obeys the fact that if we assume X_t for price of tth day thus the next day price could be formulated as:

$$X_{t+1} = X_t + \delta_{t+1} \quad (6)$$

Where δ could be a positive or a negative value. On the other hand, if we consider X_1 for the first day, other days prices predictions will be calculated as:

$$X_2 = X_1 + \delta_1 \qquad (7)$$
$$X_3 = X_2 + \delta_2 = X_1 + (\delta_1 + \delta_2) \qquad (8)$$
$$X_4 = X_3 + \delta_3 = X_1 + (\delta_1 + \delta_2 + \delta_3) \qquad (9)$$

$$X_n = X_{n-1} + \delta_{n-1} = X_1 + (\delta_1 + \delta_2 + \dots + \delta_{n-1}) \qquad (10)$$

In (10) we assume $(\delta_1 + \delta_2 + \dots + \delta_{n-1}) = \delta$, therefore the ith prices would be determined through the following function:

$$F(X_i) = X1 + \delta \quad (11)$$

Here, we try to find $F(X_i)$ through Genetic Programming and determine the δ function.

4. RESULTS AND DISCUSSIONS:

After designing the model, simulations are conducted in Genetic Programming approach; also there were 3 different types of neural networks and neuro-fuzzy hybrid system which were presented by us before [9]. In the following simulations, around 247 patterns are considered in training sets and the remaining 100 samples are used for test set, so that the simulation is validated for the next 100 days, but in just 1 step prediction. All the results are gathered in 1 picture for doing a comparison.

Here, after simulation through G.P. with specific adjustment and setting, the following formula tree is generated (Fig. 4-1) and also the result for 1 step prediction is shown respectively in fig 4-2:
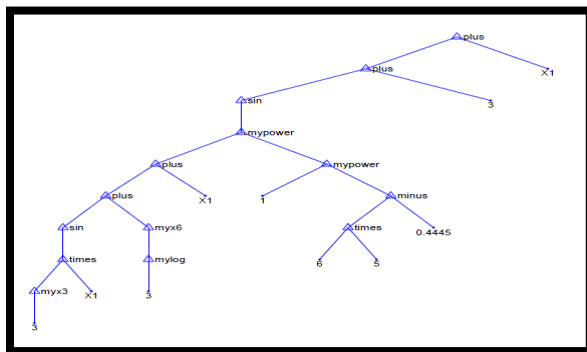
Figure 4-1: The Generated formula for finding the et days prices of PET with specific settings.
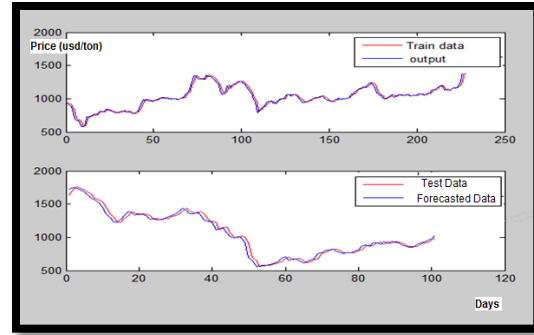
Figure 4-2: Forecasting using Genetic programming

The probability function for Mutation and Cross Over in simulation are not fixed and they have been adjusted dynamically and adaptively to obtain the best possible result (see fig. 4-3).
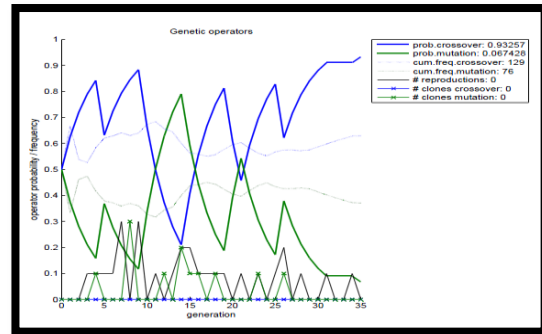
Figure 4-3 : operation probability and frequency is adjusted dynamically.

In the first step , the simulations were conducted for 1 step prediction and all the results were gathered and shown in figure 4-4 .It's obviously clear that the first two pure neural networks results cannot satisfy the research question, but the LoLiMoT and G.P. are competing to present the best possible answers.
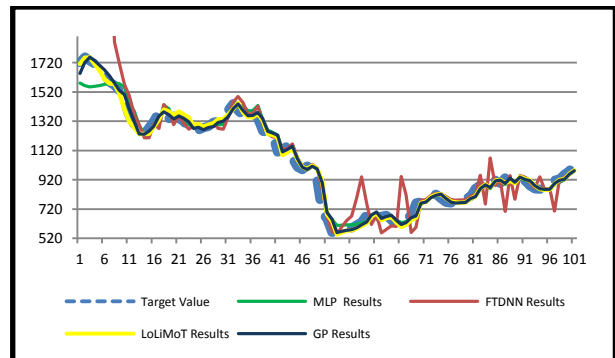
Figure 4 -4: All 4 simulations results for 100 days prediction with 1 step prediction in 1 figure

Generally, to check the error volume in such a problem, researchers use Normalized Mean square Error which is defined as follows:

$$\mathbf{NMSE} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i*=1}^{n} y_i^2}(12)$$

Based on the above formula, the error rate of each model for 1 step prediction in LoLiMoT and G.P. are as follows:

MODEL NAME	MLP	FTDNN	LoLiMoT	G.P
NMSE RATIO	0.002831	0.07701	0.001411	0.001367

Table 4 -1: NMSE ratio for around 100 day's prediction with 1 step prediction for LoLiMoT and G.P.

Based on desirability which was defined previously, just the answers of these two models are desirable. In the next step, just these two models were tested for 5, 10 and 15 steps prediction. Figure 4-4 shows the results for 15 steps prediction for the last two models which already had better estimation .As it was mentioned in table 4-2, the error ratio for AHIS model including Neuro-fuzzy is absolutely better than other 2 models; However, G.P. results are even better in 1 step prediction (although the results are too close to each other). It means that for short-term predictions the G.P. wins the race. It seems that the number of patterns in this phase is not completely enough, but in this situation the results for LoLiMoT are better and ultimately the results are considerable. It means that for long-time prediction LoLiMoT answers are closer to our purpose. Table 4-3 demonstrates the results of 15 steps prediction for the next 15 days.
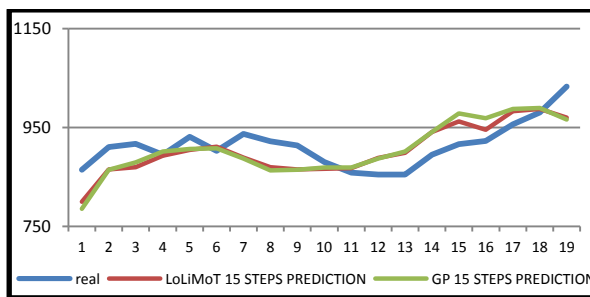
Figure 4 -4: selected simulation result for 19 days prediction with 15 step prediction for G.P and LoLiMoT

Test Day	Real Data	F.V. LoLiMoT	F.V. G.P.	Err. LoLiMoT	Err. G.P.
1	864.92	799.97	785.97	64.95	78.95
2	910.88	865.40	864.87	45.48	46.01
3	916.98	869.86	879.26	47.12	37.72
4	894.93	892.91	901.23	2.01	6.31
5	931.56	904.55	906.56	27.01	25.00
6	902.55	911.41	907.88	8.87	5.34
7	937.13	888.91	887.26	48.22	49.87
8	922.53	869.65	863.59	52.88	58.93
9	913.90	865.49	864.55	48.41	49.35
10	879.91	866.81	869.35	13.10	10.55
12	858.56	868.18	869.36	9.63	10.81
13	854.77	888.47	887.27	33.7	32.50
14	855.12	898.74	901.59	43.63	46.48
15	895.05	940.89	941.26	45.84	46.21

Table 4 -3: Forecasted value and Error volume for LoLiMoT with 15 step prediction

Finally, based on the formula which was mentioned in (12) the NMSE ratio would be as follows:

MODEL NAME	LoLiMoT	G.P.
NMSE RATIO	0.001859	0.002262

Table 4 -3: NMSE ratio for around 19 days prediction with 15 step prediction for both the LoLiMoT and G.P. answers

5. CONCLUSION:

In the introduction we asked "How to forecast PET chips prices for short time and long time?" This research follows our previous research which proposed a hybrid neuro-fuzzy system for predicting long-term forecasting in specific economic item. Previously, we showed that AHIS which includes LoLiMoT (a hybrid neuro-fuzzy model) provides a relevant answer to this question. Here, a G.P. based model has been tested and the answers for 1 step predictions improved the results of our previous research; however, still the results of AHIS including LoLiMoT are better in long-term forecasting. Here, the theoretical interest is to propose a new model that extends the Efficient Market Hypothesis. On the managerial Interest side, this model could be embedded in a Decision Support System (DSS). Our experience in that field indicates that such tools could be very useful for real decision-makers on PET market.

This communication has some Limitations. It seems that by increasing the number of testing samples and the range of training samples and events (especially in long-term prediction), the system would be more stable and the answers would be far more accurate. The last limitation is that, all other models which have the potential for better answers are not yet applied; Models such as using the Markov model and the combination of HMM with a neuro-fuzzy system. Therefore, for further researches it's strongly offered to researchers to find a model, combining the Markov Model with Neural Networks.

REFERENCES:

[1] N. Masterakis, al., Recent Advances in Data Networks, Communications, and Computers, WSEAS Press, Bucharest, Romania, 2008.

[2] R.S.T. Lee, J.N.K. Liu, Norn Predictor – Stock Prediction using a Neural Oscillatory – based recurrent

network, International Journal of Computational Intelligence and Applications 1(2001) 431-459.

[3] Y.S. Abu-Mostafa, A. Atiya, Introduction to Financial Forecasting, Applied Intelligence, 6 (1996) 205-213.

[4] D. Zhang, J.J.P. Tsai, Machine Learning applications in Software Engineering, World Scientific Publishing, Singapore, 2005.

[5] J.W. Lee, P. Jonghun, O. Jangmin, L. Jongwoo, H. Euyseok, A Multiagent Approach to Q-Learning for Daily Stock Trading, Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 37 (2007) 864-877.

[6] Y. Perwej, A. Perwej, Prediction of the Bombay Stock Exchange (BSE) Market Returns Using Artificial Neural Network and Genetic Algorithm, Journal of Intelligent Learning Systems and Applications, 4 (2012) 108-119.

[7] M. Kaboudan, M. Conover, A Three-Step Combined Genetic Programming And Neural Networks Method Of Forecasting The S&P/Case-Shiller Home Price Index, International Journal of Computational Intelligence and Applications, 12 (2013) 1-16.

[8] B. Krose, P. Van der Smagt, An Introduction to Neural Networks, 8th ed., University of Amsterdam, Amsterdam, 1996.

[9] M.S. Fazli, Prévoir le prix des polyéthylène terephthalate (PET) en utilisant un

système intelligent, in, University of Montesquieu Bordeaux IV, Bordeaux, 2012.

[10] J.R. Koza, Genetic programming II: automatic discovery of reusable programs, MIT Press, 1994.

[11] J.R. Koza, Genetic programming: on the programming of computers by means of natural selection, MIT Press, 1992.

[12] J.R. Koza, D. Andre, F.H. Bennett, M.A. Keane, Genetic Programming III: Darwinian Invention and Problem Solving, Morgan Kaufmann Publishers Inc., 1999.

[13] M.S. Fazli, J.F. Lebraty,A Comparative Study on Forecasting Polyester Chips Prices, Using Different Hybrid Intelligent Systems, Accepted for publishing in IJCNN 2013 Conference Proceeding and IEEExplore, 2013.

APPENDIX I: LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Networks
DSS	Decision Support System
EM	Efficient Market Hypotheses
FA	Fundamental Analysis
FDY	Fully Drawn Yarn
FTDNN	Focused Time Delay Neural
GP	Genetic Programing
ICIS	Integrated chemical information system
JSE	Johannesburg Stock Exchange
LOLIMOT	Locally Linear Model Tree
MEG	Mono Ethylene Glycol
MLP	Multi-Layer Perceptron
PET	Poly Ethylene Terephtelate
POY	Partially Oriented Yarn
PTA	Purified Terephthalic Acid
PX	Paraxylene
TA	Technical Analysis

SESSION

XIII TECHNICAL SESSION ON APPLICATIONS OF ADVANCED AI TECHNIQUES TO INFORMATION MANAGEMENT FOR SOLVING COMPANY-RELATED PROBLEMS

Chair(s)

Dr. David de la Fuente
Dr. Jose A. Olivas

Analysis and comparison of Candlestick patterns applied to S&P 500

J. Quintairos[1]**, R. Rosillo**[1]**, D. de la Fuente**[1]**, N. García-Fernández**[1]
[1]Department of Business Management, University of Oviedo, Gijón, Spain

Abstract - *This document provides an overview of the Candlestick analysis. Candlestick patterns are used to design trading strategies. In this paper a trading strategy is implemented by each Candlestick pattern using S&P500 index. The data covers the period from January 2004 to September 2012. The results explain that Hanging Man pattern achieves the best results if it is compared with other Candlestick patterns and Buy and Hold strategy.*

Keywords: Technical analysis, quantitative trading strategies, Candlestick patterns.

1   Introduction

Candlestick analysis was created in Japan to forecast the rice grain and satisfactory results were obtained. It is spreading among investors and most experienced professional analysts with unprecedented success. Candlestick analysis is simple, clear and, accurate. Candlestick patterns are easy to manage and all kinds of investors can use them.

The main feature of Candlestick charts is the use of the open and close session to form a range (a rectangle). The range is hollow if close session is higher than open session and the range is filling with black color if close session is lower than open session.

Nisson, 2001 [1] relies on the Candlestick technique, arguing it is an indicator for predicting market values. Marshall, 2006 [2] shows that the use of candlestick trading strategies does not provide profits for the Dow Jones Industrial Average stocks over the 1992-2002 period. Using the same approach to examine the Japanese market on a list of 100 Tokyo stock market also failed to demonstrate the effectiveness of the Candlestick technique.

However, Goo, Chen and Chang, 2007 [3] found that investors can gain an average return of about 10% through buying and holding stocks for 10 days when observing "Bullish Harami" pattern. Zhu Min, 2011 [4] mentioned that the "Bearish Harami" pattern does work on Taiwan 69 electronic securities through buying on bullish patterns and selling in bearish patterns.

Candlestick technique is simpler against Fibonacci ratios and Elliott Wave. Also it proves high effective if it is applied properly because it allows a psychological reading of the session. This feature can be applied in a quantitative algorithm.

Therefore, this kind of analysis highlights the following aspects: simplicity of the application against highly sophisticated techniques such as the principle of Elliot Wave, reading psychological ability of each day, and good results are achieved if it is properly applied.

The rest of the paper is structured as follows. In Section 2, the research goals are explained. Section 3 explains the trading algorithm developed. Section 4 shows the empirical results of the different candlestick patterns. Finally, Section 5 provides some concluding remarks.

2   Research goals

The research compares different candlestick patterns using S&P500 index. The amount of Candlestick patterns is really huge, so we apply five patterns widely used by investors: Hanging Man, Shooting Star, Inverted Hammer, Bearish Harami and Bearish Engulfing.

The generated profits by each Candlestick pattern are calculated using the dataset. After that we consider the hit ratio obtained and the achieved points of the index to determine the success of each Candlestick pattern.

The results are compared with the Buy and Hold strategy.

3   Data and methodology

3.1   Data

The dataset used in this study was obtained for free from Casa Kishoo database [5]. This data has all necessary information in the prices of continuous market companies. The dataset of S&P500 is originated from January 2004 to September 2012.

As seen in Brock, 1992 [6] the results are shown in different periods and globally to analyze better the strategy. By this way, it is possible to analyze the results in bullish movements and bearish movements.

3.2   Candlestick patterns

A Candlestick pattern is a chart representation formed by open, close, high and low price.

Candlestick patterns can prove how strongly they are depending on the length of the body.

The selected patterns for this study are explained below:

3.2.1 Hanging Man

It has a small body, white or black, whose closure is near the opening. Its lower shadow is at least twice the size of the real body. It has not got or almost not upper shadow. Unconditionally appears in an uptrend.

If Hanging Man occurs shortly after the opening, i.e. there is a sell-off that causes prices to fall sharply to gradually close near the opening (black real body) or above it (white real body), means that bullish forces are still in force and recovers, but with somewhat fragile, because it cannot make new highs above the previous day.
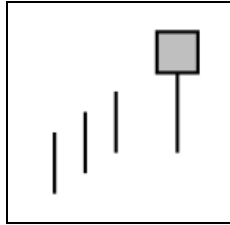
Fig. 1: Hanging Man pattern.

3.2.2 Shooting Star

It has a small body, white or black, whose closure is near the opening. It has an upper shadow usually three times the actual body. Unconditionally appears in an uptrend. It has not got or almost not lower shadow.

If Shooting Star occurs, it means that the value opens strongly on previous day's closing. Always in a strong uptrend, but reached a point downtrends strongly imposes, losing thrust bullish force closing from the lowest point.
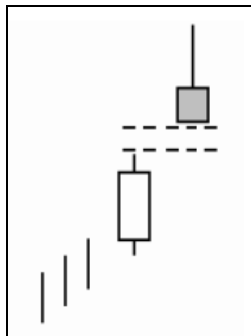
Fig. 2: Shooting Star pattern.

3.2.3 Inverted Hammer

It has a small body, white or black, in which its closure is near the opening. It has an upper shadow is at least twice the actual body size. It has not got or almost not lower shadow. Unconditionally appears after a downtrend.

If Inverted Hammer occurs, it means that after the downtrend, the market opened near its low. Then prices change direction and formed a bullish rally. However, bullish forces cannot sustain the rise during the day and finally turn prices to close near the low of the day.
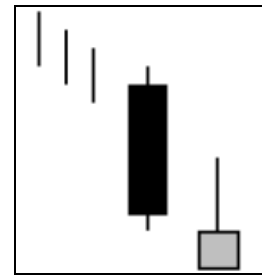
Fig. 3: Inverted Hammer pattern.

3.2.4 Bearish Harami

This pattern is a formation of a long candle, white or black, followed by a small candle the next day, white or black, within the previous candle, after an uptrend.

If Bearish Harami occurs, it means is a continuation pattern, rather than change. That is, the market it is in a "pause" to continue with an upward trend. This happens most of the time, but it is not a sure thing, since Harami shows that after an uptrend, prices are restrained at the time to follow their progress.
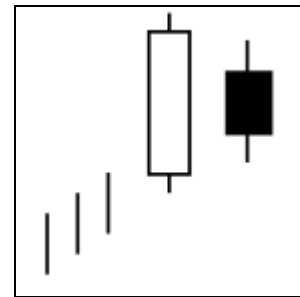
Fig. 4: Bearish Harami pattern.

3.2.5 Bearish Engulfing

This pattern is a formation of a candle with white body and another candle with black body that completely covers the white real body of the previous day, with a prior trend necessarily bullish.

If Bearish Engulfing occurs, it means that after a clear uptrend appears a small white real body candle indicating that bullish forces are losing strength. The next day, there are still bullish forces, so it opens above the white real body, but the bearish forces are catching the bullish ones, closing below the white real body of the previous day.

Fig. 5: Bearish Engulfing pattern.

For more information regarding the appearance Candlestick chart patterns, it is recommended to consult Morris, 1995 [7].

3.3   Identification of candlestick patterns

To identify patterns, an application from Matlab captures the open, close, high and low values. Then, checks if these values can be assigned to any of the above patterns and it stores them.

Whereas "n" is the current day and "n-1" is the previous day for each of the patterns, here there are the respective checks:

3.3.1   Hanging Man

IF [open(n)-close(n)]*2 ≤ close(n)-low(n)

3.3.2   Shooting Star

IF [close(n)-open(n)]*2 ≤ high(n)-close(n)
AND [close(n)-open(n)]*2 > open(n)-low(n)
AND close(n-1) > open(n-1)

3.3.3   Inverted Hammer

IF [close(n)-open(n)]*2 c high(n)-close(n)
AND [close(n)-open(n)]*2 > open(n)-low(n)
AND close(n-1) < open(n-1)

3.3.4   Bearish Harami

IF close(n) < open(n)
AND close(n-1) > open(n-1)
AND open(n) < close(n-1)
AND close(n) > open (n-1)
AND [close(n-1) - open(n-1)] > [open(n)-close(n)]*2

3.3.5   Bearish Engulfing

IF close(n) < open(n)
AND close(n-1) > open(n-1)
AND open(n) ≤ close(n-1)
AND close(n) > open (n-1)

3.4   Trading strategies

Next section (Empirical Results) summarizes the terms B-S-1 and B-S-2. The objective of this analysis is, for each pattern, to determine when a pattern has to advise to buy and sell: close of next day (B-S-1) or close of 2 next days (B-S-2). The analysis applies both actions and develops a statistical profit in both situations, applying to all patterns. A variant to this research would it be to alter this lag such as a strategy "B-S-4", to buy and sell within four next days. Given the nature of the investigation, an analysis has been applied in short term, so we only consider the results obtained by the strategies B-S-1 and B-S-2.

Both strategies will be compared with the "buy and hold" strategy, which involves buying shares on the first day (January 1, 2004) and sell only the last day (August 31, 2012).

Generally, Morris, 1995 [7] argues that Candlestick analysis technique is considered a short-range, defining the maximum period as 10 days evaluable. Therefore we have chosen two strategies belonging to that period.

4   Empirical results

As mentioned in section 3.4, for the input dataset, it would be attempted to assign one of five patterns available. Once assigned a pattern, the profit is calculated if the share is bought and sold the next day (B-S-1) or if the share is bought and sold the next two days (B-S-2).

4.1   Comparing strategies

Fig. 6, on the last page, contains the summary obtained by applying Candlestick Analysis.

Regardless of the type of strategy chosen (B-S-2 or B-S-1), it can be seen that, by far, the pattern "Hanging Man" is the one that provides a higher profit if is applied an action of buying and selling when is detected. The other patterns are similar: "Inverted Hammer" provides positive results (no investment would be lost), but the profit is very small.

The reason for this difference in the pattern "Hanging Man" is noticeable when we compare the hit ratio of each pattern, as shown in Table 1 on the last page.

As can be seen, the hit ratio of each of the patterns, regardless of the strategy of buying and selling always reaches a value of approximately 50%. However, the number of times the "Hanging Man" pattern is assigned is notably higher (this is because the activation conditions of "Hanging Man" pattern are easier to be fulfilled than the rest). Due to this greater frequency, there are more decisions of buy and sale.

4.2   Comparison against the "Buy and Hold" strategy

The following Table 2 are the results compared to the strategy "Buy and Hold":

Profit	B-S-2	B-S-1	Buy and Hold
Hanging Man	1077,27	288,20	287,16
Shooting Star	-58,10	-11,75	287,16
Inverted Hammer	127,29	84,55	287,16
Bearish Harami	5,24	-101,37	287,16
Bearish Engulfing	7,64	-1,21	287,16

Table 2: Comparison against the "Buy and Hold" strategy.

The results show a marked inefficiency of the selected patterns, with the exception of the "Hanging Man" pattern. However, it can be seen that the difference of "buy and hold" strategy with B-S-1 is virtually zero, being B-S-2 the optimal strategy to apply, which reports a profit five times higher.

5   Conclusions

Using Candlestick Analysis, we could compare the results with Cihan et al (2009) [8], Lee et al (1999) [9] and Zhu Min (2011) [4]. The fact that the "Hanging Man" pattern offers better results may be influenced because Candlestick analysis is affected by the strength which a given pattern occurs. It would be interesting to apply more complex patterns to analyze their relevance.

This pattern is really relevant for this dataset. However, it is possible that from other different dataset the prediction does not want to be correct due to the variation in the strength of the pattern [4].

The profit change based on the strategy (B-S-1 or BS-2) is because Candlestick Analysis is under the influence of the days to predict, settling in the future 10 days maximum. In [4] it has been more successful in making predictions using a range of 5 days instead of 10; this is an aspect to consider.

It is recommended to combine Candlestick Analysis with other algorithms as expert systems or genetic algorithms to address this inaccuracy [8].

6   Acknowledgements

7   References

[1] Nisson, S. (2001). Japanese candlestick charting techniques: A contemporary guide to the ancient.

[2] Marshall, B.R., M.R. Young & L.C. Rose (2006). Candlestick Technical Trading Strategies: Can They Create Value for Investors? Journal of Banking & Finance, 30, 2303-2323.

[3] Goo, Y., Chen, D. & Chang, Y. (2007). The Application of Japanese Candlestick Trading Strategies in Taiwan. Investment Management and Financial Innovations, 4, 49-71.

[4] Zhu, Min. (2011). Is Candlestick Technical Analysis Effective in Trend Reversal Prediction?, Proceedings of China-Canada workshop on financial engineering and enterprise risk management.

[5] http://www.tradingcasakishoo.com

[6] Brock, W., Lakonishok, J. and LeBaron, B. (1992) Simple technical trading rules and the stochastic properties of stock returns, Journal of Finance, 47, 1731–64.

[7] Morris, G., 1995. Candlestick Charting Explained: Timeless Techniques for Trading Stocks and Futures, seconded. McGraw-Hill Trade, New York.

[8] Kamo, Takenori and Dagli, Cihan (2009). Hybrid approach to the Japanese candlestick method for financial forecasting, Expert Systems with applications.

[9] Lee, KH and Jo, GS (1999). Expert system for predicting stock market timing using a candlestick chart, Expert Systems with applications.
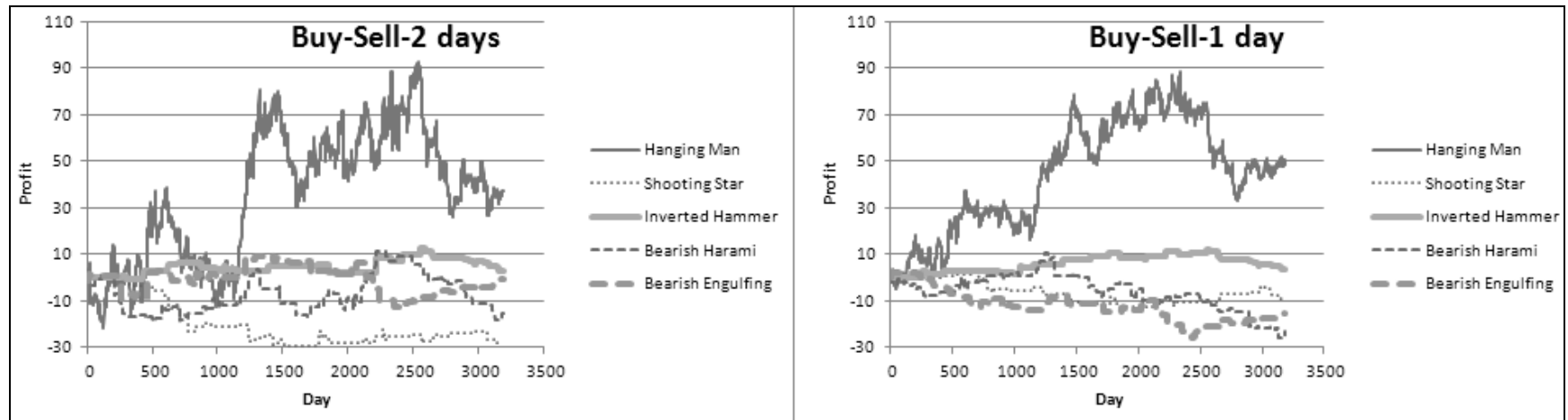
Fig. 6: Comparison of B-S-2 and B-S-1 strategies.

HIT RATIO	B-S-2				B-S-1			
	Success	Error	% Success	% Error	Success	Error	% Success	% Error
Hanging Man	15197	15720	49,15%	50,85%	15394	15371	50,04%	49,96%
Shooting Star	456	458	49,89%	50,11%	470	444	51,42%	48,58%
Inverted Hammer	367	354	50,90%	49,10%	363	356	50,49%	49,51%
Bearish Harami	972	1022	48,75%	51,25%	958	1035	48,07%	51,93%
Bearish Engulfing	693	743	48,26%	51,74%	710	723	49,55%	50,45%

Table 1: Hit ratio from Candlestick patterns for B-S-2 y B-S-1 strategies.

THE QUALITY OF E-SERVICE IN B2C WEBSITES: AN APPROXIMATION WITH FUZZY DECISSION SUPPORT SYSTEMS

Castro A.[1], Puente J.[1], Gómez A.[1], Fernández I.[1], Ponte, B.[1]
[1]Business Administration Department, University of Oviedo, Gijón, Asturias, Spain.

Abstract - *The present paper proposes a model to evaluate B2C websites. It has been considered a set of dimensions that have influence in the evaluation according to the literature review and a study of reliability. Once validated the model, it makes use of fuzzy inference systems in order to eliminate the uncertainty associated with the decision-making process.*
As a result of this research, a model capable of designing a simple and intuitive knowledge base is obtained (experts and/or users) in the evaluation of B2C websites, thus obtaining a further optimization of the results obtained.

Keywords: Fuzzy Inference Systems, E-service, Quality Service, B2C websites evaluation.

1   Introduction

The growth of online shopping activities has been particularly rapid, presenting business opportunities, opening up a vast new territory, which business enterprises are eager to explore and resulting in more consumers and companies choosing the Internet as means of purchase and sale (Shih et al. 2002).

Among the different types of E-commerce, Business to Customer (B2C) technology covers those websites and transactions through which organizations sell goods and services to customers directly over the Internet. Efficiency and cost effectiveness of this technology have already transformed the web into a global environment for business. Nevertheless, consumers find many difficulties in the online purchasing processes that have to do with searching and browsing websites, evaluating and comparing products or with the payment process among others. All these inconveniences affect the satisfaction of customers with the B2C websites (Kuo et al.,2011).

The present paper proposes an exhaustive analysis of the variable influence-teas in the design and evaluation of B2C web sites. Once validated the proposed model is a methodology based on the fuzzy set theory. This methodology allows to improve the interpretation of the base of knowledge induced in the system of decision, improving

this way the behaviour of the model opposite to other simpler systems.

2   Literature Review

E-service quality analysis methods are supported by questionnaires with Likert-type scale. These questionnaires are filled into by potential customers. Next, Table 1 summarises the most commonly used methods, according to the reviewed literature, to evaluate the quality of the online service.

Table 1: **Relevant dimensions for evaluate the quality of e-service in B2C websites.**

RELEVANT DIMENSIONS FOR EVALUATE THE QUALITY OF E-SERVICE IN B2C WEBSITES	
Design	**Definition**
	Web design is an activity that consists in planning, modelling and implementing websites
	Authors
	Rangathan y Ganapathy (2001), Alpar (2001), Melián y Padrón (2005), Brown y Jayakody (2008), Lin (2009), Oppenheim y Ward (2010), Stefani y Xenos (2011).
Information	**Definition**
	Information, in general terms, is defined as an organised set of processed data that form a message that improves the level of knowledge of the receptors of the message.
	Authors
	Chang y Sangjae (2002), DeLone y McLean (2003), Evanshitzki et al. (2004), Melián y Padrón (2005), Cao et al. (2006), Meziane y Nefti (2006), Nielsen (2006), Joia y Barbosa (2008), Bauernfeind et al. (2009), Lin (2009), Sun y Lin (2009), Oppenheim y Ward (2010).
Warranty	**Definition**
	Warranty usually refers to the action that a person or company performs so as to assert what has previously arranged. By offering a Warranty, it is intended to supply greater security to the fulfiling of an obligation or the payment of a debt, according to the arranged.
	Authors
	Santos (2003), Melián y Padrón (2005), Otim y Varum (2006).
Offer	**Definition**
	In general terms an offer is a proposal made to execute or fulfil a series of requirements.
	Authors
	Yang et al. (2000), Melián y Padrón (2005), Cao et al. (2006).
Personalisation	**Definition**
	Personalisation is the adaption of the products or services to the necessities of customers, in such a way that a customer feels exclusive, unique and different.
	Authors
	Melián y Padrón (2005), Koutsabasis et al (2008) Adjei y Clark (2009), Bauernfeind et al (2009), Oppenheim y Ward (2010).
Payment Security	**Definition**
	Payment Security is defined as a protection Warranty regarding made transactions and used means of payment during the process of purchase.
	Authors
	Madu y Madu (2002), Francis y White (2002), Otim y Varum (2006), Venkataiahgari et al. (2006), Oppenheim y Ward (2010).
Privacy	**Definition**
	Privacy in the service is defined as the protection given to the data provided by the customer regarding record, treatment and preservation of those data. In electronic commerce, privacy refers only to the customers' personal data.
	Authors
	Ranganathan y Ganapathy (2001), Zeithaml et al. (2002), Oppenheim y Ward (2010), Stefani y Xenos (2011).
Trust	**Definition**
	Trust is defined as the expectations that are set on something or someone. Regarding electronic commerce, it refers to the hope that online customers have that the purchase made in a web portal will be performed according to their expectations.
	Authors
	Gefen (2000), Kaplan y Nieschwietz (2003), Oppenheim y Ward (2010), Hong y Cho (2011).

3   Proposed Model

The model that is proposed to evaluate B2C platforms stems from a Delphi Study and through in-depth interviews performed in Spain by experts on Market Online. The study of the most relevant variables describes in part 2 has been the starting point. Additionally, a pre-test has been made to know the feeling of the customers.

To make the B2C evaluation system more easily understandable, it has been structured into three subsystems. Every subsystem's output dependent variable is function of a maximum of three input variables. Figure 1 depicts this structure.

A partial evaluation will be obtained from every subsystem. Later, the partial results obtained are combined, thus achieving the final evaluation of the B2C websites.
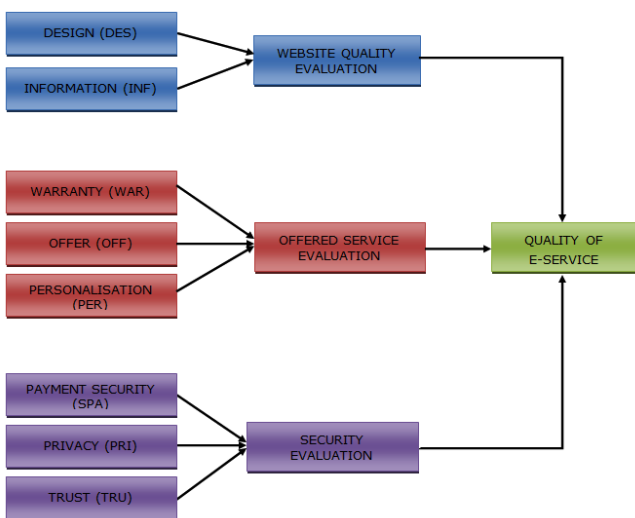
Figure 1: Model to evaluate the quality of e-service in B2C websites.

The information has been gathered by means of personal surveys filled in by Internet regular purchasers. The scales to evaluate websites underwent a pre-test performed by 405 regular Internet customers aiming to put to the test the proper comprehension of the questions and add, remove or modify the composition of the final questionnaire. The target universe has been defined as the set of individuals that are older than 18, that are regular buyers on the Internet and that live in Spain. The sampling error was ±2,42 % with a trust level of 95%.

To determine the use of the B2C websites evaluation model, it has been evaluated the one-dimensionality and reliability and validity of the scales of measure. During the first phase of validation, the variables that have been studied with all the items present in the questionnaire have been considered. This exploratory analysis made apparent the necessity of removing some of the items to optimise the composition of the scales. To that end, it has been used the statistical suite SPSS v.19. As a result of this pre-test, it has been performed an exploratory factor analysis of principal components with Varimax rotation and a reliability analysis by means of Cronbach's alpha. Table 5 shows the results obtained in this study.

Table 2: **Results obtained through Exploratory Analysis and Cronbach´s α.**

	DES	INF	WAR	OFF	PER	PAY	PRI	TRU
Explicative Information Percent	59,3	68,4	65,7	54,4	52,5	65,1	90,6	82,1
Cronbach´s α	0,80	0,90	0,90	0,80	0,60	0,80	0,90	0,79

Cronbach's alpha coefficient is an indicator used to determine the reliability of the scales. As seen in the previous table, all proposed variables exceed the threshold value 0,6 recommended in exploratory studies (Hair et al. 1999). Besides, in all cases, the proposed variables show a cumulative percentage of variance upper 50%. This is why the results obtained in this pre-test allow the validity of the proposed model to be confirmed.

4   Methodology

The methodology used to in this research is based on fuzzy decision support systems. These are based on the theory of fuzzy sets developed by Zadeh (1965), which allows you to incorporate the term of uncertainty associated with the decision-making process which is better approximate to the reality (Martin y Sanz, 2005).

In fuzzy inference systems, variables, linguistic, allow to process both qualitative and quantitative information since they take as values labels associated to concepts of common language. This contrasts with the traditional numeric variables whose values are numbers (Ponce, 2011).

In our case, to assess the quality of perceived e-service - by users or experts- presents a high level of subjectivity. For this reason, the use of linguistic labels will allow a better adaptation of the assessment criteria given in the evaluation of B2C websites. It will also be tested the design of a fuzzy system in the field of B2C e-commerce, and specifically in the evaluation of websites, allowing a better interpretation of the basis of knowledge inserted in the decision systems, thus improving the results to the output of the proposed model.

4.1 Fuzzy Decision Support Systems (FDSS)

In this case, for use to a fuzzy decision support system will be utilised aided by the application Matlab fuzzy logic toolbox ® version 2.0. (Mamdani and Gaines, 1981).

So as to make the inference, it is required a knowledge base built up from an expert's know-how, able to explain the operation of the system by means of a set of linguistic rules made up with the input and output variables (other options would allow to constitute the base of rules from reasoning systems based on cases). So, in the first place, it is necessary to define the range and form of the labels in which the dominions of the system variables are partitioned in order to define the variables in a fuzzy way. Mamdami type fuzzy decision systems, supported by all the aforementioned, allow to make inferences through a process that consists of five stages (Chen & Klein, 1997): fuzzyfication, application of logical operators in every rule's antecedent, implication to every rule's consequent, aggregation of all rules' consequents and defuzzyfication of the final aggregate.

Additionally, it will be established the adequate sequence of fuzzy subsystems that are part of the model and the evaluation knowledge will be aggregated as conditional rules for each qualification subsystem (in such a way that allows to intuitively assign linguistic labels to all the variables (Martínez-López et al., 2009)). Figure 2 represents the labels used in input and output variables.

Figure 2: Labels of input and output variables.

Later, the four rule-bases that are to be used are defined for each evaluation subsystem by using the expert's knowledge: Web Quality, Offered Service, Service Security and Final Evaluation (See Table 7).

Table 3: **Bases of rules for each partial and final subsystem.**

Labels: EL – Extremely low; VL – Very low; L – Low; M – Medium; H – High; VH – Very High; EH – Extremely High

For instance, the shaded cell in the table related to security expresses the rule: If the score of security is medium

and privacy is low and confidence is high, then the final evaluation of the security is medium.

Once a fuzzy subsystem has been designed, the evaluation of a website can be inferred from it as a function of the crisp values given to the input variables. What is more, it is easy and intuitive to analyse the congruence of the obtained evaluations by using the inference maps supplied by each model's subsystem. In these maps, the scores of the input variables are represented by the height of the surface in each point. For, example, the Figure 3 shows the evaluation of the offered service as a function of two input variables (Offer and Personalisation), while keeping constant the other input variable which is not shown in the graphic (Warranty).

Figure 3: Map of solutions of the Offered Service subsystem with low warranty (left graphic) and high warranty (right graphic).

The analysis of the influence of those input variables on the Offered Service evaluation shows that in case the customer perceives that the offered products have a short warranty period, the final evaluation is penalised so that the maximum values are around 5,5. On the opposite, if the variable [Warranty] has medium or high values, the evaluation of the offered service can reach values over 9 points (if the scores of the rest of the variables are regarded as high).

Therefore, establishing a warranty system in agreement with customers' necessities and wishes may grant competitive advantages to the company over those that do not establish a warranty policy.

5 Conclusions

In these paper carried out an analysis of the main variables used in the evaluation of websites in the textile sector. It has conducted a comprehensive literature review, along with a Delphi method and a pre-test. Once considered the appropriateness of the different variables have been a market study to regular users in the search/purchase of items of fashion through the Internet, thus the validation of the proposed model.

Once the proposed model has been validated, it has analyzed the uncertainty associated with the processes of evaluation of this type of websites. As a result, a model, based on fuzzy inference systems, has been able to improve the interpretation of the induced in decision system, knowledge base thus improving behaviour compared to other simpler systems.

As future lines of research is intended to merge with neural network fuzzy inference systems in order to achieve a base of knowledge as dynamic as possible.

6   References

[1]   Adjei M, Clark M (2010) "Relationship marketing in a B2C context: the moderating role of personality traits". Journal of Retailing and Consumer Services. 17, (73-79). 2010.

[2]   Alpar P. "Satisfaction with a web site: It´s measurement, factors and correlates". Working Paper No. 99/01. Philipps-Univesität Marburg, Institute für Wirtschaftsinformatik, 2001.

[3]   Bauernfeind U, Mayr T, Zins A. **"**A conceptual model for quality dimensions for B2C recommender systems". ECIS Proceedings, (1007-1018). 2006.

[4]   Brown I, y Jayakody R. "B2C e-Commerce success: a test and validation of a revised conceptual model". Electronic Journal Information Systems Evaluation. 11, (167-184). 2008.

[5]   Cao M, Zhang Q, Seydel J. "B2C e-commerce web site quality: an empirical examination". Industrial Management & Data Systems, 105, (645-661). 2006.

[6]   Chang L, Sangaje L. "A cognitive map simulation approach to adjusting the design factors of the electronic commerce web sites". Expert Systems with Applications, 24, (1-11). 2002.

[7]   Chen, CB, Klein CM. "An efficient approach to solving Fuzzy MADM Problems". Fuzzy Sets and Systems. 88, (51–67). 1997.

[8]   Evanschitzy H. "E-satisfaction: a re-examination". Journal of Retailing. 80, (239-247). 2004.

[9]   Francis J, White L. "PIRQUAL: A escale for measuring customer expectations and perceptions of quality in Internet retailing". In K.Evans and L. Scheer (Eds), 2002. Marketing educator's conference: Marketing theory and applications. 13, (263-270). 2002.

[10] Gefen D. "E-commerce: The role of familiarity and trust". Omega, 28, (725-737). 2000.

[11] Hair JF, Anderson R, Tatham R, Black W. "Multivariate Analysis". Prentice Hall. Fifth Edition. Madrid. 1999.

[12] Hong IB, Cho H. "The impact of consumer trust on attiutdinal loyalty and purchase intentions in B2C marketplaces: Intermediary vs seller trust". International Journal of Information Management. 31, (469-479). 2011.

[13] Joia LA, Barbosa de Oliveira LC "Development and testing of an e-commerce web site evaluation model". Journal of Electronic Commerce in Organizations. 6, (37-53), 2008.

[14] Kaplan SE, Nieschwietz RJ. "A Web assurance services model of trust for B2C e-commerce". International Journal of Accounting Information Systems. 4, (95-114). 2003.

[15] Koutsabasis P, Stravraskis M, Viorres N, Darzentas JS, Spyrou T, Jarzentas J. "A descriptive reference framework for the personalization of e-business applications". Electron Commerce Res. 8, (173-192). 2008.

[16] Kuo HM, Chen CW, Chen CW. "A study of merchandise information and interface design on B2C websites". Journal of Marine Science and Technology. 19, (15-25), 2011

[17] Lin HF. "An application of fuzzy AHP for evaluating course website quality". Computers & Education. 10. 2009.

[18] Madu CN, Madu AA. "Dimensions of e-quality". InternationalJournal of Quality & Reliability Management. 19, (246-258). 2002.

[19] Mamdani EH, Gaines BR. "Fuzzy reasoning and its applications", NY. Academic Press. 1981.

[20] Martin B, Sanz A, García A. "Redes Neuronales y Sistemas Borrosos". 3º Ed.. RA-MA. 2005.

[21] Melián L, y Padrón V. "Quality in e-commerce B2C: perceptions and expectations". Conferencia IADIS Ibero-Americana. 2005.

[22] Meziane F, Nefti S. "Evaluating E-commerce trust fuzzy logic". School of Computing, Science and Engineering. University of Salford. United Kingdom. 2006.

[23] Nielsen J, Loranger H. "Prioritizing Web Usability". New Riders. Berkeley. 2006.

[24] Oppenheim C, Ward L. "Evaluation of web sites for B2C e-commerce". Aslib Proceedings. 58, (237-260), 2010.

[25] Otim S, Varum G. "An empirical study on web-based services and customer loyalty". European Journal of Information Systems. 15, (527-541). 2006.

[26] Ponce, P. "Artificial Intelligence with engeniering applications". Marcombo. Barcelona. 2011.

[27] Ranganathan C, Ganapathy S. "Key dimensions of business-to-consumer web sites". Information& Management. 39, (457-465). 2001.

[28] Santos J. "E-service quality: a model of virtual service quality dimensions". Managing Service Quality. 13, (233-246). 2003.

[29] Shih, T., Chiu C, Hsu H, Lin F. "An integrated framework for recommendation systems in e-commerce". Industrial Management and Data System, 102, (417-431). 2002.

[30] Stefani A, Xenos M. "Weight-modeling of B2C system quality". Computer Standards & Interfaces. 33:411-421. 2011.

[31] Sun C, Lin G. "Using fuzzy TOPSIS method for evaluating the competitive advantages of shopping websites". Expert Systems with Applications. 36, (11764-11771). 2009.

[32] Venkataiahgari AK, Atwood JW, Debbabi M. "A survey of secure B2C commerce for multicast services". IEEE CCECE/CCGEI. Concordia University. Montreal. 2006.

[33] Yang Z, Peterson R, Huang L. "Taking the pulse of Internet pharmacies". Marketing Health Services. 21, (4-10), 2000.

[34] Zadeh, Lofti A. "Fuzzy Sets". Information and control. 83, (338-353). 1965.

[35] Zeithaml V, Parasuraman A, Malhotra A (2002) Service quality delivery through web sites: A critical review of extant knowledge. Journal of the Academy of Marketing Science. 30, (362-410). 2002.

Real-time Scheduling of Flexible Manufacturing Systems using Support Vector Machines and Neural Networks

P. Priore[1], R. Pino[1], J. Parreño[1], J. Lozano[1] and M. Monterrey[1]
[1]EPI de Gijón, Campus de Viesques, 33203 Gijón, Asturias, Spain

Abstract - *Dispatching rules are usually applied to schedule jobs in Flexible Manufacturing Systems (FMSs) dynamically. Despite their frequent use, one of the drawbacks that they display is that the state the manufacturing system is in dictates the level of performance of the rule. As no rule is better than all the other rules for all system states, it would highly desirable to know which rule is the most appropriate for each given condition, and to this end this paper proposes a scheduling approach that employs Support Vector Machines (SVMs) and backpropagation neural networks. Using these latter techniques, and by analysing the earlier performance of the system, "scheduling knowledge" is obtained whereby the right dispatching rule at each particular moment can be determined. A module that generates new control attributes is also designed in order to improve the "scheduling knowledge" that is obtained. Simulation results show that the proposed approach leads to significant performance improvements over existing dispatching rules.*

Keywords: Scheduling, Neural Networks, SVMs, FMS, Simulation

1 Introduction

One of the most commonly applied solutions to the scheduling problem in FMSs involves using dispatching rules, which have been evaluated for performance by many researchers (see for example, [12], [21], [22]). Almost all the above studies point to the fact that rule performance depends on the criteria that are chosen, and the system's configuration and conditions (utilisation level of the system, relative loading, due date tightness, and so on). It would thus be interesting to be able to change dispatching rules at the right moment dynamically.

The literature describes two basic approaches to modify dispatching rules. The first approach is to select a rule at the appropriate moment by simulating a set of pre-established dispatching rules and opting for the one that provides the best performance (see for example, [7], [8], [9], [23]). The second approach, involving artificial intelligence, requires a set of earlier system simulations (training examples) to determine what the best rule is for each possible system state. A machine learning algorithm [11] is trained to acquire knowledge through these training examples, and this knowledge is then used to make intelligent decisions in real time (see for example, [16], [19], [20]). Aytug [1] and Priore [15] provide a review in which machine learning is applied to solving scheduling problems.

Nevertheless, there are hardly any studies in the literature that compare the different types of machine learning algorithms used in scheduling problems. This paper therefore presents a scheduling approach that uses and compares SVMs and neural networks. To improve the manufacturing system's performance, a new approach is also proposed whereby new control attributes that are arithmetical combinations of the original attributes can be determined.

The rest of this paper is organised as follows. Machine learning algorithms used in this paper are first described. An approach to scheduling jobs that employs machine learning is then presented. This is followed by the experimental study, which describes a new approach to determine new control attributes from the original ones. The two machine learning algorithms used are also compared. Finally, the proposed scheduling approach is compared with the alternative of using a combination of dispatching rules constantly. A summary of the results obtained concludes the paper.

2 Neural Networks and Support Vector Machines

"Backpropagation neural networks", or multilayer perceptron [18], which will be applied in this work, figure amongst those networks that are most well-know and most widely used as pattern classifiers or function approximators ([5], [10]). The backpropagation training algorithm is used in this type of neural networks. This algorithm calculates the most adequate connection weights and thresholds so that the difference between the network output and the desired one is minimised.

Support vector machines [4] were originally designed for binary classification. Let (x_1, y_1), (x_2, y_2), … , (x_n, y_n) be a group of data belonging to Class 1 or Class 2, where $x_i \in R^N$ and the associated labels be $y_i=1$ for Class 1and -1 for Class 2 $(i=1, … , n)$. The formulation of SVMs is as follows:

$$Min \quad \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i$$

subject to the constraints:

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i \quad i = 1,...,n$$
$$\xi_i \geq 0 \quad i = 1,...,n$$

(1)

where w is the weight vector; C is the penalty weight; ξ_i are non-negative slack variables; b is a scalar, and x_i are mapped into a higher dimensional space by a non-linear mapping function ϕ. Mapping function ϕ needs to satisfy the following equation:

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

where $k(x_i, x_j)$ is called kernel function.

Minimizing $\frac{1}{2} w^T w$ implies that SVMs tries to maximise $\frac{2}{\|w\|}$, which represents the margin of separation between both classes. The data that satisfy the equality in Eq. (1) are called support vectors. Moreover, by adding a set of non-negative Lagrange multipliers α_i and β_i to generate the Lagrangian, the upper- mentioned constrained optimization problem can be worked out with the dual form shown below:

$$Max \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k(x_i, x_j)$$

subject to the constraints:

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$
$$0 \leq \alpha_i \leq C \quad i = 1,...,n$$

Having obtained the support vectors (SVs), the decision function for an unseen data (x) is as follows:

$$y = sign \left\{ \sum_{SVs} \alpha_i y_i k(x, x_i) + b \right\}$$

3   Scheduling using Neural Networks and Support Vector Machines

Two contrasting features need to be fulfilled for a real-time scheduling system that dynamically modifies dispatching rules to work properly [13]:

1. Rule selection must take into account a variety of information about the manufacturing system in real time.
2. Rule selection must be completed fast enough for real operations not to be delayed.

One way of doing this is to employ some class of knowledge about the relationship between the manufacturing system's state and the rule to be applied at that moment. However, one of the most difficult problems is precisely how this knowledge is to be acquired. Machine learning algorithms, such as SVMs or neural networks, are used to do this. However, the training examples and the learning algorithm must be adequate for this knowledge to be useful. Moreover, in generating the training examples, the attributes selected are crucial to the performance of the scheduling system [3].

Figure 1 shows a scheduling system that employs machine learning. The example generator creates different manufacturing system states via the simulation model and choose the best dispatching rule for each particular state. The machine learning algorithm employs the examples to generate the knowledge required to make future scheduling decisions. The real time control system using the 'scheduling knowledge', the manufacturing system's state and performance, choose the best dispatching rule for job scheduling. Further examples may possibly be needed in order to refine the knowledge about the manufacturing system depending on the performance of the latter.
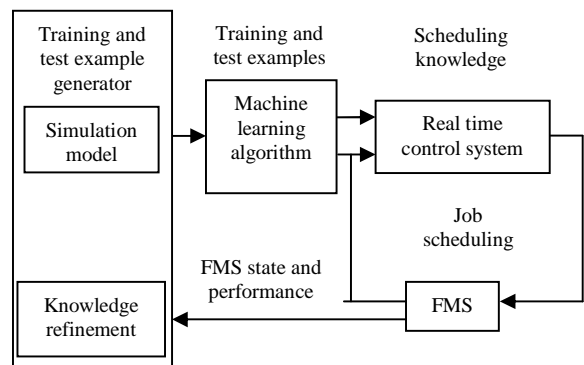
Figure 1. General overview of a knowledge-based scheduling system.

4 Experimental Study

4.1 The proposed FMS

The selected FMS consists of a loading station, an unloading station, four machining centres and a material handling system. Two types of decision are studied in the FMS proposed. The first is the selection by the machine of parts assigned to it using the following dispatching rules: SPT (Shortest Processing Time), EDD (Earliest Due Date), MDD (Modified Job Due Date), and SRPT (Shortest Remaining Processing Time). These rules were selected because of their fine performance in earlier studies (see for example, [14]). The second type of decision involves the selection of the machines by the parts, as an operation can be processed on different machines. The dispatching rules employed in this FMS are: SPT (Shortest Processing Time), NINQ (Shortest Queue), WINQ (Work in Queue), and LUS (Lowest Utilised Station).

4.2 Generating training and test examples

The control attributes used to describe the manufacturing system state must first be defined in order to generate training and test examples. In this particular FMS these are: F, flow allowance factor which measures due date tightness [2]; NAMO: number of alternative machines for an operation; MU: mean utilisation of the manufacturing system; U_i: utilisation of machine i; WIP: mean number of parts in the system; RBM: ratio of the utilisation of the bottleneck machine to the mean utilisation of the manufacturing system; RSDU: ratio of the standard deviation of individual machine utilisations to mean utilization.

The training and test examples needed for the learning stage are obtained by simulation using the WITNESS programme. The following suppositions were made to do this: (1) Jobs arrive at the system following a Poisson distribution; (2) Processing times for each operation are sampled from an exponential distribution with a mean of one; (3) The actual number of operations of a job is a random variable, equally distributed among the integers from one to four; (4) The probability of assigning an operation to a machine depends on the parameters PO_i (percentage of operations assigned to machine i). These percentages fluctuate between 10% and 40%. It is also assumed that the first two machines have a greater workload; (5) The number of alternative machines for an operation varies between one and four; (6) The job arrival rate varies so that the overall use of the system ranges between 55% and 95%; (7) The value of factor F fluctuates between one and ten.

As mean tardiness and mean flow time in the system are the most widely used criteria to measure system performance in all manufacturing systems, they are also employed in this study. In all, 1100 different control attribute combinations were randomly generated, and 100 of these were used as test examples. For each combination of attributes, mean tardiness

and mean flow time values resulting from the use of each of the dispatching rules in isolation were calculated. Sixteen simulations were actually needed to generate a training or test example, as there are four rules for each of the decisions to be taken.

4.3 The application of neural networks

Backpropagation neural networks are particularly used to solve classification problems such as the one being considered in this work. Table I provides a summary of the results obtained using different-sized sets of training examples for the criteria of mean tardiness and mean flow time. Generally, it can be seen that as the number of training examples increases, test example error (examples that have not previously been dealt with) decreases considerably. Table I also shows that test error fluctuates between 16% and 12% upwards of 400 examples for the criterion of mean tardiness. Furthermore, for the criterion of mean flow time, test error is observed to oscillate between 7% and 4% upwards of 500 examples. Errors for this latter criterion are lower due to there being five dispatching rule combinations (SPT+SPT, SPT+NINQ, SPT+WINQ, MDD+WINQ, SRPT+WINQ) that are really used. In contrast, twelve combinations are used for the criterion of mean tardiness.

Table I. Test error using neural networks for the criteria of mean tardiness (MT) and mean flow time (MFT).

Number of examples	Test error (MT)	Test error (MFT)
200	21%	9%
300	17%	10%
400	16%	10%
500	15%	7%
600	13%	5%
700	14%	6%
800	16%	6%
900	15%	5%
1000	15%	4%
1100	12%	4%

The ideal configuration of the neural network used for the criterion of mean tardiness was found to have 11 input nodes (one for each control attribute), 16 nodes in the hidden layer, and 12 nodes in the output layer (one for each dispatching rule combination). Similarly, the ideal configuration of the neural network employed for the criterion of mean flow time was found to have 11 input nodes, 16 nodes in the hidden layer, and 5 nodes in the output layer.

4.4 The application of support vector machines

The scheduling problem is essentially a multi-class classification problem as several dispatching rule

combinations are employed in the FMS. This study uses the one-against-one method to extend the binary SVMs to generate the multi-class scheduler since this method is more suitable for practical use than other methods [6]. In the same way, in this study, the radial basis function (RBF) and the polynomial function have been used as kernel functions. After several preliminary tests, it has been decided to make use of the RBF Kernel since it is the one that shows a better performance. Furthermore, by employing the grid search technique on the examples, the best performance for the SVMs is obtained when C=1,000 and σ=10. Table II provides a summary of the results obtained for the criteria of mean tardiness and mean flow time. Generally, it can be seen that as the number of examples increases, test example error decreases considerably. Table II also shows that test error fluctuates between 11% and 10% upwards of 700 examples for the criterion of mean tardiness. Furthermore, for the criterion of mean flow time, test error drops to 1% upwards of 700 examples.

Table II. Test error using SVMs for the criteria of mean tardiness (MT) and mean flow time (MFT).

Number of examples	Test error (MT)	Test error (MFT)
200	16%	6%
300	15%	5%
400	15%	2%
500	16%	2%
600	12%	2%
700	11%	1%
800	11%	1%
900	11%	1%
1000	10%	1%
1100	10%	1%

4.5   Generating new control attributes

On occasions, it is necessary to obtain arithmetical combinations of the original attributes to improve the scheduling knowledge. But in many cases these combinations are not known beforehand, and are only found in very simple manufacturing systems after close examinations of their simulation results. For these reasons, a module was designed which automatically selects the 'useful' combinations of the original attributes by using simulation data which originally provided test and training examples. To do this, the basic arithmetic operators considered are adding, subtracting, multiplying and dividing. The pseudo-code for the generator of the new control attributes is as follows:

1. Determination of the combinations of the present attributes.
2. Generation of new training and test examples in the light of earlier combinations.

3. Selection of the 'useful' combinations, which are in the decision tree and in the set of decision rules generated by C4.5 [17].
4. If the new decision tree and/or the set of decision rules has fewer classification errors, go back to step one. If not, stop the algorithm.

However, the decision to continue may also be taken at step four because, even though error may not be improved by the present iteration, it may well be improved during later iteration(s).

The proposed module rendered the following 'useful' control attribute combinations for the criterion of mean tardiness: U1+U2, U1+U4, U2+U3, U1-U2, U2-U4, U3-U4 and U3/U4. Table III shows the results obtained for the criterion of mean tardiness when the SVMs and the generator module of new control attributes were applied. It can be seen from the results that test error oscillates between 10% and 8% from 600 training examples upwards, and that the lowest test error was achieved with 1000 and 1100 examples. The proposed module is then applied for the criterion of mean flow time, and the following combinations of attributes were determined to be 'useful': U1-U2, U3-U4, U1/U2 and U2/U3. The Table shows how test error drops to 0% from 700 examples upwards. If these results are compared with those in Table II, an improvement can be seen to exist. Only sets of 600 examples or more were used, as lower errors are obtained upwards of this training set size.

Table III. Test error using SVMs and the generator module of new control attributes for the criteria of mean tardiness (MT) and mean flow time (MFT).

Number of examples	Test error (MT)	Test error (MFT)
600	10%	1%
700	9%	0%
800	9%	0%
900	9%	0%
1000	8%	0%
1100	8%	0%

Test error using backpropagation neural networks and the new attributes generated was likewise calculated. Results are shown in Table IV, where it is again clear that classification error drops compared to the alternative of using the original control attributes. For the criterion of mean tardiness, the backpropagation neural network gives a 12% test error, compared to the 8% error of the SVMs. Furthermore, for the criterion of mean flow time, the SVMs are seen to give zero test error, whereas the backpropagation neural network generates a bigger test error (2%). Finally, mention should be made of the fact that the ideal neural network for the criterion of mean tardiness has 18, 15 and 12 neurons in the input,

hidden and output layers respectively, whilst for the criterion of mean flow time the ideal neural network has 15, 10 and 5 neurons.

Table IV. Test error using neural networks and the generator module of new control attributes for the criteria of mean tardiness (MT) and mean flow time (MFT).

Number of examples	Test error (MT)	Test error (MFT)
600	15%	4%
700	14%	4%
800	12%	4%
900	14%	2%
1000	12%	2%
1100	12%	2%

4.6 Learning-based scheduling

To select the best combination of dispatching rules according to the FMS's state in real time we must implement the scheduling knowledge in the FMS simulation model. Selecting the monitoring period is another key question because the frequency used to test the control attributes determines the performance of the manufacturing system. To do this, multiples of the average total processing time for a job, which in our particular case are 2.5, 5, 10 and 20 time units, are taken (see for example, [8], [9], [23]). In view of the results in the previous section, 1000 examples were used for both performance criteria. Five independent replicas of 100,000 time units were carried out.

Table V. Mean tardiness (MT) and mean flow time (MFT) for the proposed strategies.

Strategy used	MT	MFT	Strategy used	MT	MFT
SPT+SPT	5.4916	2,4041	**MDD+NINQ**	**1.1450**	1,2495
SPT+NINQ	1.2220	**1,0438**	**MDD+WINQ**	**1.1566**	1,2546
SPT+WINQ	1.2011	**1,0415**	MDD+LUS	2.5326	1,8537
SPT+LUS	2.5920	1,5187	SRPT+SPT	6.1452	2,6549
EDD+SPT	4.7207	2,6106	SRPT+NINQ	1.4174	1,1410
EDD+NINQ	1.6802	1,3909	SRPT+WINQ	1.4089	1,1419
EDD+WINQ	1.6885	1,3948	SRPT+LUS	3.0247	1,7094
EDD+LUS	3.2958	2,0493	**SVMs**	**1.0000**	**1.0000**
MDD+SPT	4.7620	2,6750	Neural Network	1,0298	1.0055

Table V summarises the results obtained. Mean tardiness and mean flow time values in the Table are the average of the five replicas. Readability has been improved by showing values in the Table that are relative to the lowest mean tardiness and mean flow time obtained (these are assigned a value of one). The monitoring period chosen was 2.5 time units. Table V shows that the best alternative is to employ a knowledge-based strategy and that the SVMs generate lower mean tardiness values than neural network. The combinations

MDD+NINQ and MDD+WINQ are the best of the strategies that use a fixed combination of dispatching rules, but their mean tardiness values are higher than the neural network alternative by between 14.50% and 15.66%.

Moreover, the SVMs give better results than the neural network for the criterion of mean flow time. Table V also shows that the combinations SPT+NINQ and SPT+WINQ generate the least mean flow time from amongst the strategies that apply a fixed combination of rules. However, mean flow time values are greater than the SVMs alternative by between 4.15% and 4.38%. Finally, the SVMs-based system is compared with the other strategies by using ANOVA. The conclusion is that this scheduling system stands out above the other strategies with a significance level of less than 0.05.

5 Conclusions

An approach for scheduling using SVMs and neural networks is proposed in this study. A generator module of new control attributes is also incorporated, and this reduces test error obtained with the machine learning algorithms leading to better performance of the manufacturing system. The SVMs-based scheduling system is shown to provide the lowest mean tardiness and mean flow time values. Future research might focus on using more decision types for the proposed FMS. However, the more decision types that are used, the more simulations are needed to generate the training and test examples. A simulator could usefully be incorporated to decide which rule to apply when the "scheduling knowledge" provides two or more theoretically correct dispatching rules. Finally, a knowledge base refinement module could also be added, which would automatically modify the knowledge base when major changes in the manufacturing system come about.

6 References

[1] Aytug, H., Bhattacharyya, S., Koehler, G.J. and Snowdon, J.L. (1994), "A review of machine learning in scheduling", *IEEE Transactions on Engineering Management*, Vol. 41, pp. 165-171.

[2] Baker, K.R. (1984), "Sequencing rules and due-date assignments in a job shop", *Management Science*, Vol. 30, pp. 1093-1103.

[3] Chen, C.C. and Yih, Y. (1996), "Identifying attributes for knowledge-based development in dynamic scheduling environments", *International Journal of Production Research*, Vol. 34, pp. 1739-1755.

[4] Cortes, C. and Vapnik, V. (1995), "Support-vector network", *Machine Learning,* Vol. 20, pp. 273-297.

[5] Freeman, J.A. and Skapura, D.M. (1991), *Neural Networks: Algorithms, Applications, and Programming Techniques*, Addison Wesley, Reading, MA.

[6] Hsu, C.W. and Lin, C.J. (2002), "A comparison of methods for multi-class support vector machines", *IEEE Transactions on Neural Networks,* Vol.13, pp. 415-425.

[7] Ishii, N. and Talavage, J. (1991), "A transient-based real-time scheduling algorithm in FMS", *International Journal of Production Research*, Vol. 29, pp. 2501-2520.

[8] Jeong, K.-C. and Kim, Y.-D. (1998), "A real-time scheduling mechanism for a flexible manufacturing system: using simulation and dispatching rules", *International Journal of Production Research*, Vol. 36, pp. 2609-2626.

[9] Kim, M.H. and Kim, Y.-D. (1994), "Simulation-based real-time scheduling in a flexible manufacturing system", *Journal of Manufacturing Systems*, Vol. 13, pp. 85-93.

[10] Lippman, R.P. (1987), "An introduction to computing with Neural Networks", *IEEE ASSP Magazine*, Vol. 3, pp. 4-22.

[11] Michalski, R.S., Carbonell, J.G. and Mitchell, T.M. (1983), *Machine Learning. An Artificial Intelligence Approach,* Tioga Press, Palo Alto, CA.

[12] Montazeri, M. and Wassenhove, L.N.V. (1990), "Analysis of scheduling rules for an FMS", *International Journal of Production Research*, Vol. 28, pp. 785-802.

[13] Nakasuka, S. and Yoshida, T. (1992), "Dynamic scheduling system utilizing machine learning as a knowledge acquisition tool", *International Journal of Production Research*, Vol. 30, pp. 411-431.

[14] O'keefe, R.M. and Kasirajan, T. (1992), "Interaction between dispatching and next station selection rules in a dedicated flexible manufacturing system", *International Journal of Production Research*, Vol. 30, pp. 1753-1772.

[15] Priore, P., de la Fuente, D., Gómez, A. and Puente, J. (2001), "A review of machine learning in dynamic scheduling of flexible manufacturing systems", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 15, pp. 251-263.

[16] Priore, P., de la Fuente, D., Pino, R. and Puente, J. (2001), "Learning-based scheduling of flexible manufacturing systems using case-based reasoning", *Applied Artificial Intelligence,* Vol 15, pp. 949-963

[17] Quinlan, J.R. (1993), *C4.5: Programs for Machine Learning,* Morgan Kaufmann Publishers, San Mateo, CA.

[18] Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986), "Learning representations by back-propagating errors", *Nature*, Vol. 323, pp. 533-536.

[19] Shiue, Y.-R., Guh, R.-S. and Tseng, T.-Y. (2009), "GA-based learning bias selection mechanism for real-time scheduling systems", *Expert Systems with Applications,* Vol. 36, pp. 11451-11460.

[20] Shiue, Y.-R., Guh, R.-S. and Lee, K.-C. (2011), "Study of SOM-based intelligent multi-controller for real-time scheduling", *Applied Soft Computing,* Vol. 11, pp. 4569-4580.

[21] Stecke, K.E. and Solberg, J. (1981), "Loading and control policies for a flexible manufacturing system", *International Journal of Production Research*, Vol. 19, pp. 481-490.

[22] Tang, L.-L., Yih, Y. and Liu, C.-Y. (1993), "A study on decision rules of a scheduling model in an FMS", *Computers in Industry*, Vol. 22, pp. 1-13.

[23] Wu, S.-Y.D. and Wysk, R.A. (1989), "An application of discrete-event simulation to on-line control and scheduling in flexible manufacturing", *International Journal of Production Research*, Vol. 27, pp. 1603-1623.

Improving Educational Methods by Means of an Intelligent Virtual Agent

Celia G. Rospide[1], Cristina Puente[1], Jose A. Olivas[2]

[1] Advanced Technical Faculty of Engineering – ICAI, Comillas Pontifical University, Madrid, Spain

[2] Information Technologies and Systems Dept, University of Castilla-La Mancha Ciudad Real, Spain

Abstract *–In this paper we explore the role that an educational chatbot could provide improving the learning process of a student. We present a virtual agent, as part of an e-Learning platform capable of guiding and answering questions related to a studied subject.*

Keywords: Virtual agent, Flexible answer, e-Learning.

1 Introduction

The arrival of a new educational model, added to the introduction of new technologies as part of the educational model, provide new ways of learning processes that makes the process more creative and flexible for the student.

In particular, the use of virtual agents as software to guide users and answer his questions is not new, and is widely applied through the Internet in pages to sell products, like Anna´s from Ikea, as seen in figure 1, and other type of pages like call centers:

Fig. 1 Ikea's chatbot.

The construction of a chatbot able to emulate a flexible conversation is highly complex. Nowadays, the so called virtual robot has not been developed in its totality, and the 'perfect chatbot'still does not exist. For this reason, there are several enterprises with good reputations in the computer market, like IBM, working on these types of projects to improve their performance [1].

The idea of a virtual agent to answer questions is not new. An intelligent virtual agent is an entity capable of perceiving its environment and processing those perceptions in a rational way [2]. To be considered as AI systems, they have to fulfil the following characteristics [3]:

- Learn new problems and improve the range of solutions.

- Real time adaptation.

- Analysis of a situation in terms of behaviour, evaluating the possibilities of success and failure.

- Evolve through interaction with the environment.

- Manage great amounts of data in order to choose the best solution.

One of the greatest problems in the implementation of a virtual agent is the design of its 'personality', as it has to be original and intelligent. However, the behavior of an agent many times is not what is expected, especially when there are many restrictions or when the problem is not defined precisely.

Among the diversity of virtual agents, we can find the group of Intelligent Tutoring Systems (ITS). These type of agents were created in the eighties with the idea of sharing knowledge using an intelligent way to guide the student in their learning process. So a intelligent tutor can be defined as [4] " *a software system which employs AI techniques to represent knowledge, and interacts with the students to transfer it*" or as [5] "*systems to model teaching, learning,*

communication and knowledge domain of the specialist and the understanding of the student in that domain".

Recent research is focused on finding an alternative to a human tutor. However, most of the tutoring systems do not acquire the expected level of rationality due to the difficulty involved with modelling human behavior, beyond the application of the most advanced programming techniques.

The goal of this project, is to guide the student through the learning of a given subject. So, taking these premises as a basis, this chatbot is an ITS, able to distinguish among several user profiles (according to the ones stored into a database), to serve as support in the learning process of a student in a given subject. The system is composed of a hardware and software architecture which is explained in the following sections, as well as the interaction among the software modules that make understanding possible and the composition of a flexible answer. Finally, we will conclude with some guidelines for future work as well as some conclusions about the proposed work.

2   Creating a user profile

The goal of this development is to serve as a useful educational tool for the student. So that, we have to create a virtual agent flexible for each one of the students. This chatbot is divided in several modules, as we will see in the following sections, but there is one module of vital importance which will be the core of its success and the main difference with other software for the same purpose. This module is the profile module, which will elaborate an specific set of exercises according to the student grades. If the student has failed, the system, at the beginning of the program will recommend the student to review the parts where he looks to be weaker, as seen in figure 2:
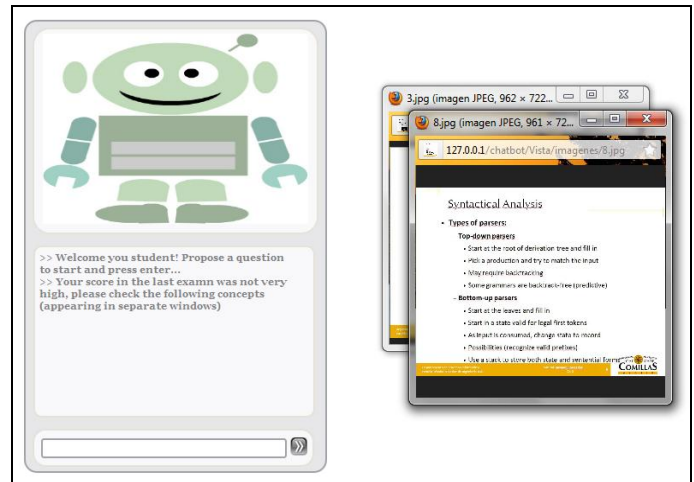
Fig. 2 Screenshot of the agent when the student fails an examn.

The platform used to develop this educational bot was a web environment. The hardware used was typical client-server architecture, where the client is a browser like Mozilla or Explorer, and the server is an Apache with some other additions like a lexical analyzer.

The software is composed of several modules written in different programming languages (like Php, Javascript, C, Lex), which interact among themselves to understand the user's question and propose a suitable answer. The software model proposed to carry out this task is a Model-View-Controller, where the model interacts with a MySQL database to extract information, the View part is the interface, written in HTML5 plus Javascript, and the Model contains the interaction with the lexical analyzer plus the specific modules written in Php. In a graphic form, figure 3 represents the interaction of all these components:

.
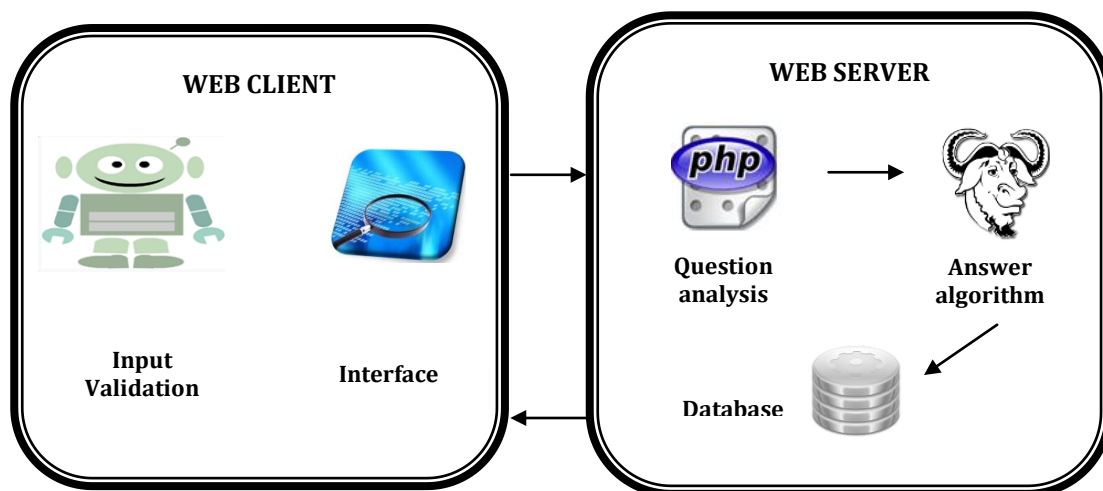
Fig. 3 Architecture of the virtual agent.

Before the questioning process, there is a screen validation to control the access to the system and to match the student with his profile. This way, by means of a search into the database, if the credentials are correct, the agent will be able to recommend the student to have a look at any theme in particular if he has failed, as seen in figure 2.

Fig.4. Validation screen to select the student's profile.

If on the other hand, the student is a good one and belongs to a high profile (marked as HIGH in the database), the system will recognize if, and will not recommend him to study anything in particular, as seen in figure 5:
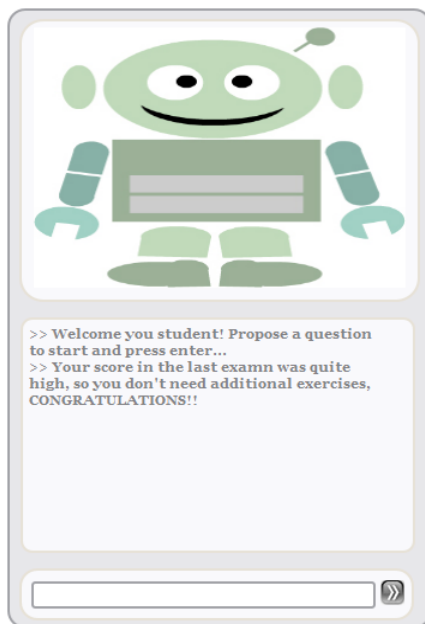
Fig.5. Validation screen to select the student's profile.

The client part is composed of two parts, the interface, and another algorithm to validate the question. The interface is composed of those graphical elements which serve as input and output for information, and has three elements, an image which varies according to the answer, a text area where the dialog between user and the machine is displayed, and an input textbox where the user must write his query.
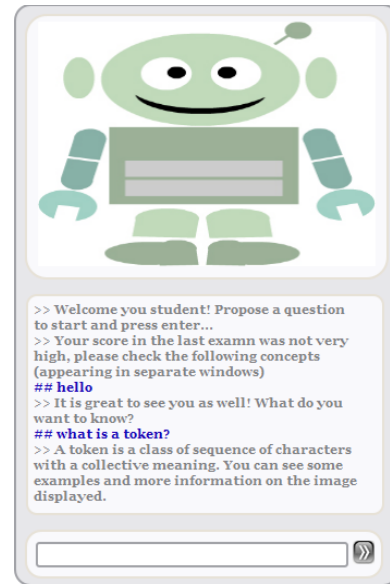
Fig.6. Client interface.

The algorithm used to validate a question was programmed using AJAX, and is in charge of sending the query to the server and waiting for an answer to be added at the end of the text area.

To achieve a greater level of realism, we have pictured several images describing the emotions of the robot. So that, if the agent knows the answer, he will be happy, or if he does not know, he will be sad. This illusion improves the addiction of the student to the platform, creating a better atmosphere to propose questions. All the possible states of the bot are described in figure 7:
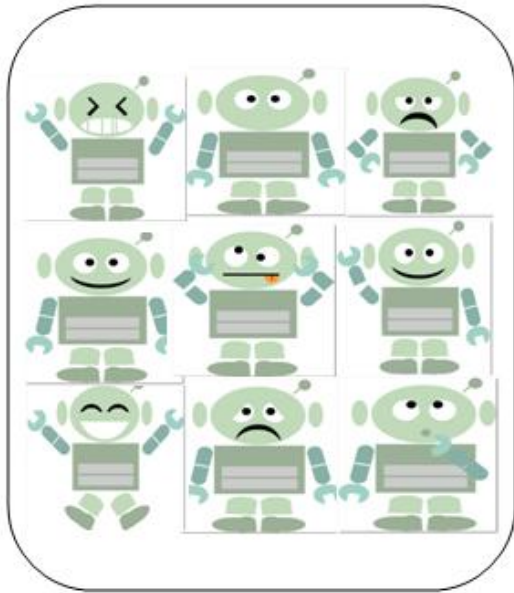
Fig.7. Agent's moods.

On the other hand, the web server is divided in three main parts, the query analysis module, the answer algorithm, programmed using the lexical analyzer FLEX[1], and a terminology database built through an ontology (hand-made), with the terms used in each part of the subject and the platform Wordnet Similarities[2] [6].

3   Query Analysis Module

The query analysis module is in charge of selecting and analyse those keywords that may appear in the question proposed by the user. It is responsible as well of produce a suitable answer, and is composed of other four sub-modules: bad words, welcome, general information and complex answer (in fig.3 represented as answer algorithm by means of FLEX morphological analyzer).

When a question is proposed, the system runs the different modules to chain a suitable answer according to the input words. In this way, the first module that evaluates the query is the bad words' one, because if it is found one of these words, the question will be rejected and the student will be warmed to be polite or he will be kicked out of the system.

This algorithm detects if there are any bad words within the input received. We have implemented a procedure that checks the query words against a database table with a complete list of these words. The result is as seen in figure 8, where the user proposes three questions with some of these words mixed in:
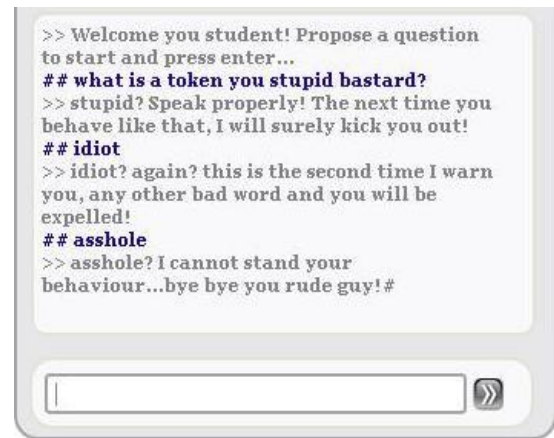
Fig. 8 Example of bad words output (grey text)

If there are no bad words in the query, the input query will access the welcome module. This algorithm notices if the user is being nice or polite to it, separating the technical query (if one exists) from the welcoming greeting. In this case the answer will be a composition of answers. As in the previous case, the number of times that the user says hello to the bot is controlled, in order to focus the conversation. If the user repeats a greeting message many times, the system will encourage him to ask more specific questions, without being expelled in this case, as seen in the following figure:
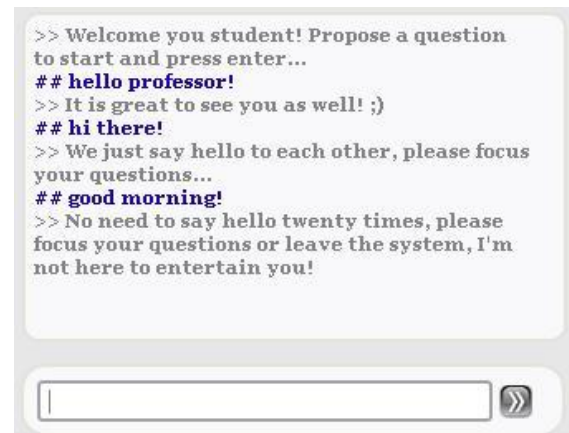
Fig. 9 Example of welcoming output (grey text)

One of the more frequent uses of a chatbot is to answer FAQs. We have included a set (through the database) of questions related to the subject such as the day of the exam, the hour, class, name of the professors and contact information, and many other useful pieces of information.

This third module, is entirely written in Php, and by matching the query keywords against those stored in the database is able to answer a wide range of questions proposed in a different way. This algorithm also evaluates the question pronouns, like how, what, when, etc., for instance, if the user begins his

---

[1] http://www.gnu.org.

[2] wn-similarity.sourceforge.net/.

question with 'when', the system will know that he is asking for time. Added to the textual answer, in some cases the system has additional stored information, like schedules, etc. so this information will be displayed along with the answer in a new window:
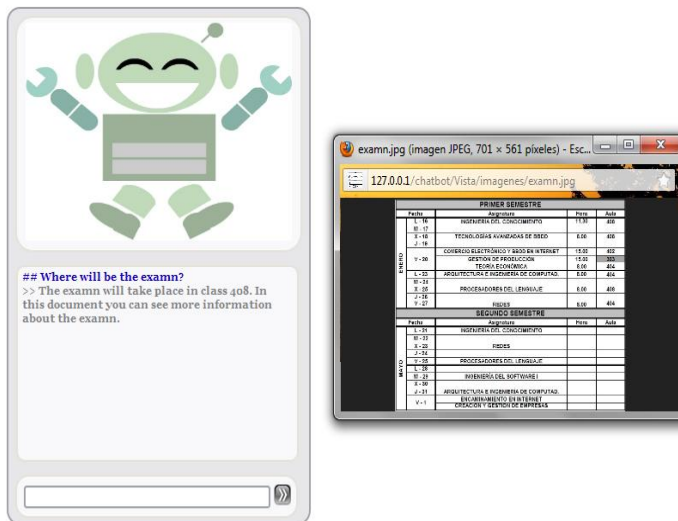
Fig. 10 Example of general information output

If the query is more specific and related to the subject's lessons, then it will be processed by a lexical analyzer, composed of an automaton which will process the input through a set of states.

4   Query Analysis Module

This module is the most important one. Is the one that makes a difference with other chatbots. The query analysis algorithm was developed using the morphological analyzer FLEX plus the C programming language. This program is able to detect and locate certain lexical patterns within a text by means of an automaton.

For our specific purpose, this automaton is composed of six states which will be able to recognize:

- Concepts related to the studied subject

- Terms and keywords specifying what the user wants to know about those concepts (like definitions, how it works, and so on).

- Interrogative pronouns.

- Words and expressions commonly used to quit the system or say goodbye.

- Words and expressions used to express gratitude.

- Clauses used to lead the conversation in a friendly way.

The automaton created in FLEX has a [Pattern]-[Action] structure, so if the automaton detects any of the defined patterns, the associated action, written in C code, will be dispatched.

The defined states control the structure of the input query, so in the first place an interrogative pronoun should appear, followed by a word or set of words which will determine what the user wants to know specifically about a concept related with the subject, followed by a concept related with the subject. The query "*What is a token?*" follows this structure, but not all queries have to, for example "*How does a lexical analyzer work?*" where the concept related to the subject is not at the end of the query. To take these problems into account, we have created six states which control the possible structures of a query.

In order to ascertain the answer to the proposed question, we have classified the types of questions on the basis of their type and priority:

- Low priority: those not related with the subject's contents.

- High priority: those containing keywords or relevant patterns.

- Guiding questions: when there is not a clear answer to the proposed question, the system will answer back with another question to check if the prediction was correct. Otherwise it will encourage the student to propose the question in a different way.

This way, if the input is correct, the algorithm will recognize the type of question and a set of keywords, and will look up the related information in the database .

The database algorithm will compare the type of question first, and with the possible results will match the input keywords with those contained in each record. The 'priority' of the matches will be taken; so many times a record that has two keywords with high priority matched will be selected instead of another with three keywords with lower priority. To do so, we have used a scoring system. The record that after the matching obtains the highest number of points is selected and its answer displayed. If this record has addition information associated, it will be displayed as well, as in the following example, where the user asks "What is a token?" and the bot gives him the definition along with the slide that contains more information related:
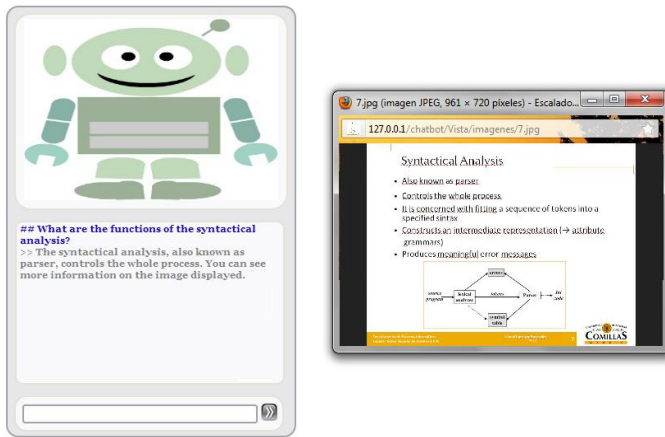
Fig. 11 Example of general information output

Many times the users are not specific with their questions, so the system will try to guess what they mean to say. The prediction algorithm is in charge of this part. The function of this process is to add flexibility and realism to the communication. So we have implemented the following considerations:

- Create the illusion of listening by introducing substrings of the input query in the answer.

- Admit its ignorance about certain issues.

- Control the date (if it is close to the exam date) to offer review exercises, for example, or make comments about how late it is, or say "*Good night*" when leaving the system.

- Redirect the conversation when the user asks for questions which have little or nothing to do with the subject.

- Introduce friendly substring in the answer like "*friend*", "*colleague*", "*dude*", etc.

- Calculate the length of the conversation, and if it is too short, encourage him to solve more problems.

5   Conclusions

This work proposes a new version of a virtual agent adapted to a student profile to help and guide him in the study of a given subject.

As future work, we are implementing this prototype in a real environment to evaluate how useful it is for a student and if it can be improved to make the study process easier. So that we would like to obtain data related to use statistics to have a real vision of the student's use.

6   Acknoledgements

7   References

[1] R. Russell. Language Use, Personality and True Conversational Interfaces. Artificial Intelligence and Computer Science. University of Edinburgh, 2002.

[2] B. Hayes-Roth. An Architecture for Adaptive Intelligent Systems, Artificial Intelligence, Vol. 72, pp. 329-365, Elsevier, 1995.

[3] N. Kasabov. Advanced Neuro-Fuzzy Engineering for Building Intelligent Adaptive Information Systems in Fuzzy Systems Design: Social and Engineering Applications. L.Reznik, V.Dimitrov and J.Kacprzyk (eds) Heidelberg, Physica-Verlag, 1998.

[4] K. Vanlehn, Student Modelling. Foundations of Intelligent Tutoring Systems, pp. 55-78, Hillsdale. N.J. Lawrence Erlbaum Associates, 1998.

[5] B. Wolf , Context Dependent Planning in a Machine Tutor. Ph.D. Dissertation, University of Massachusetts, Amherst, Massachusetts, 1984.

[6] T.Pedersen, WordNet::Similarity - Measuring the Relatedness of Concepts. Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-04), pp. 1024-1025, July 25-29, San Jose CA, 2004.

Using Fuzzy Prototypes as a Nucleation Event Classification Method

Francisco P. Romero[1], Maria T. Baeza[1], Jose A. Olivas[2] and Emilio Martínez-Torres[2]

[1]School of Industrial Engineering, University of Castilla La Mancha, Toledo, Spain
[2]Department of Informations Systems and Technologies, University of Castilla La Mancha, Ciudad Real, Spain
[3]Department of Physical Chemistry, University of Castilla La Mancha, Ciudad Real, Spain

Abstract—*Although the topic of atmospheric aerosols is not a well known by general population and it is only well know by experts, their effects could increase the consequences of climate change Moreover, the mechanisms of atmospheric aerosol formation and their evolution are not well established nowadays. An application of Zadeh's prototype theory in the knowledge discovery process is presented in this paper, and as a practical example, a method for classifying and analyzing data collected in a meteorological station is defined. This classification and analysis will allow to study and consequently to improve both our knowledge of mechanism of atmospheric aerosol formation and improve current atmospheric models*

Keywords: Fuzzy Prototypes, Nucleation Events, Atmospheric Chemistry

1. Introduction

Atmospheric aerosols have been studied for decades but they are still a not well understood phenomenon. In fact, according to the inform written by the expert commission of IPPC [1] the highest uncertainties in understanding climate change are associated with atmospheric aerosols. As a consequence our climate change predictions are limited.

Some aerosols, for example black carbon and mineral dust with high content in iron, they behave in a similar way to greenhouse gasses, being able to increase global warming. In contrast, other type of aerosols can reflect and/or disperse sun light, producing a decrease of the amount of energy that reach Earth's surface and consequently producing a cooling effect. Additionally, the smog in large polluted cities is caused because these particles disperse light.

At high concentrations, aerosols are dangerous pollutants for our health. It is thought that the exposition at high concentrations of aerosols during short periods of time doe not produce important effect in human being. However, at the long term, they can impact cardiopulmonary health, increase mortality and they reduce life expectancy [2].

Aerosols can be directly emitted to the atmosphere (primary aerosols) or they can be produced though photooxidation reactions in the atmosphere itself (secondary aerosols) from other pollutants. The mechanisms of the formation of these secondary atmospheric aerosol are not well established even when extensive measurements campaigns and long-term measurements have been taken to try to improve our understanding of this phenomenon [3]. From these studies it has been concluded that Regional new particle formation (RPNF) appears with great regularity during the day time and very rarely, if ever, during the night. In order to observe particle formation and subsequent growth up to 100 nm the air masses have to remain the same for a long time (up to 12h). For this reason one can analyze the continuous data a day at a time and classify days as RNPF event days and non-event days. To understand the factors that contribute to have RNPF even days and not-event days can help us to understand the mechanism of particle formation. Consequently, it is necessary to develop techniques to classify the current continuous datasets that allows us to correlate this information with different meteorological and chemical factors.

There are several previous works about classification methods for continuous datasets of aerosol measurements [4], [5]. Previous work includes the manual classification method developed by [6], based on visual observations and group decision making, and automatic methods like the developed by [7]. However, these models are not as successful for every station, because there are different environmental factors and/or differences in the amount of pollution between the measurement stations, moreover during the measurement period, there were some missing/bad quality data.

In consequence, there is a need of new computational methods that allow getting an automatic classification of continuous datasets of particle size distributions that get over the inconvenience of previous methods. Soft Computing [8] represents a synergy between fuzzy systems, machine learning and evolutionary computation, aimed at understanding and modeling complex phenomena and searching solutions to hard problems that can be, moreover, ill defined. Soft computing is also developing special methods that are very powerful even in solution of classical problems. This work is mainly focused on applications of fuzzy logic and machine learning algorithms to solve the problem of nucleation events detection classification that cannot be easily solved using other, conventional approaches.

The rest of this extended abstract is organized as follows: Section 2 highlights the previous research in the related area and Section 3 describe the methodology used for knowledge discovery and classification used to solve this problem. Finally, we offer concluding remarks and describe future directions of our research work.

2. Related Work

Atmospheric aerosol particle formation is frequently observed throughout the atmosphere, but the mechanism of aerosol formation is not well known yet. The climatic importance of newly formed aerosols propels the scientific community to explain the nucleation events discovering the factors that initiating nucleation.

Most studies on ambient nucleation events have investigated only physical or chemical mechanisms of nucleation and have been based on only a pre-established set of parameters. Moreover, the classification of days is carried out manually, by visual inspection, and determined by the judgment of the panel of researchers [6]. This method is very powerful but the decisions made by a panel of experts can appear to be arbitrary and highly variable, with no obvious explanation for other research groups.

To address this problem, some works introduce statistical and data mining methods to explain new particle formation. In [4] the data extracted from $Hyytiäalä$ station in Finland is analyzed using clustering and classification methods. They infer that there are two key parameters, relative humidity and preexisting aerosol particle surface (condensation sink), capable in explaining 88% of the nucleation events; but this model had some difficulties in separating non-event days from unclassified days. In order to solve this problem [7] develop and new automatic classification method based in techniques of computational intelligence like self-organizing map (SOM) and decision tree to solve this problem over the same dataset. With this method a strict and bivalent classification is obtained between "days with events" and "days without events" without any distinction more complex for example considering the intensity of the nucleation events, time during the day when nucleation events happen, etc. At the same time, it has a large percentage of days without classification that needs to be reduced to get optimal results. Moreover, the proposed models are not as successful for data recorded in different stations which may be due to different environmental factors and/or differences in the amount of pollution between the measurement stations.

In [5] Discriminant Analysis (DA) with non-parametric Epanechnikov kernel is used to find factors that classify the days as nucleation "event days" or "nonevent days". They used two different methods to test the good-ness of fit of the models: resubstitution, where the computed model is fitted to the same dataset from which it was estimated, and cross-validation, where the model is fitted to a different dataset than the one used in the estimation. The model still explained almost 88% of the events in resubstitution but it also gave a large number of false events (i.e. predict a nonevent day to be an event day), which increased the total classification error to 22%.

3. Fuzzy Prototypical Knowledge Discovery Process

The Fuzzy Prototypical Knowledge Discovery [9] (FPKD) is an extension of the classic KDD [10] process that presents as novelties the incorporation of knowledge in different points by means of the user or the expert decisions and a result prepared to generate some conceptual prototypes called Fuzzy Deformable Prototypes, based on the idea of Fuzzy Prototypical Categories.

Fuzzy Deformable Prototypes (FDP) [9] is a technique of knowledge representation based on Zadeh's Prototypes Theory [11]. Zadeh's idea suggests a concept that encompasses a set of prototypes, which represent the high, medium or low compatibility of the samples with the concept. Also FDP is based on the approach Deformable Prototypes of Bremermann [12], in order to deform the most similar prototypes to a new case until describe it completely.

The use of fuzzy logic let us get these results in a more understandable and useful way for their later use in the classification process. We can evaluate new elements from such prototypes, establish predictions for real situations and also make decisions from these predictions. Some other techniques, such as fuzzy clustering and aggregation functions, are also used, making easier the generation of structured, significant and easily updatable models.

The stages of the FPKD process applied to nucleation events classification are described in the following subsections (see also the top part of figure 1):
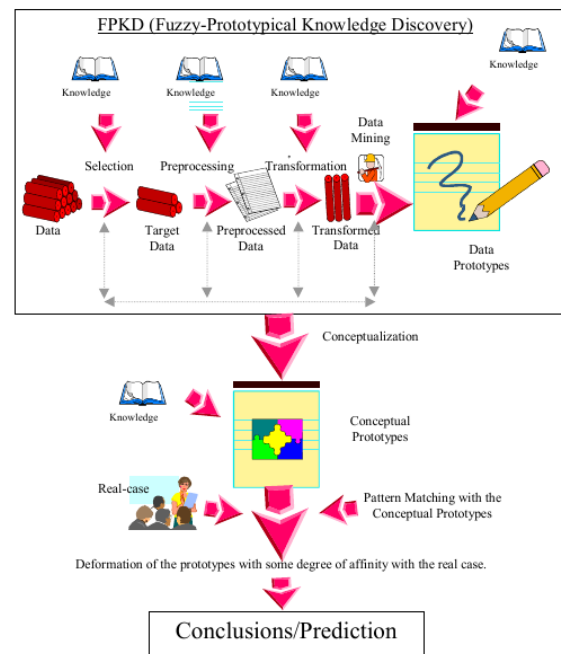
Fig. 1: Fuzzy Prototypical Knowledge Discovery Process

3.1 Selection

The measurements used in this study were performed from 1996 to 2003 at the SMEAR II station which is situated in $Hyytiäalä$ a Forestry Field Station of the University of Helsinki between Tampere and $Jyväaskylä$ in southern Finland. Aerosol size distribution was measured at a lower-(3-10 nm) and upper-size range (10-500 nm) every 10 min. This data has been widely used in the literature [4], [7]

3.2 Pre-processing

The aerosol-particle-size distributions are measured using some specific instrumentation [13]. Several issues may affect the accuracy of data collected from this instrument. The "noise" that is due to measurement error can mask changes that may, in fact, affect the classification results.

Filters are mainly used for removing noise from data specifically in a bi-dimensional distribution of data (either high and low values, smoothing, detecting edges, etc.). In this case we have used the median filter with a filter window of 3×3 size. This filter has two main advantages over other filters:

- The median is a more robust average than the mean and so a single very unrepresentative value in a neighborhood will not affect the median value significantly.
- The median filter does not create new unrealistic values when the filter straddles an "edge" in a 2D data distribution.

3.3 Transformation

In this step, it is necessary reduce the number of features used to represent the problem space. In this case, the data source includes 144 measures x 38 different sizes for each day, i.e., a day consists on a total amount of 5472 values. Many of theses values are not significant for the classification process.

According to Dalmaso [6] only the values related to particles with size below 25nm are useful to detect nucleation events. Also, nucleation events occur in a narrow window of time (from 1 to 3 hours), therefore, we can extract from each day the window of time with a maximum particles intensity (maximum amount of particles in a window of time), or with the highest growth in number of particles, called differential intensity. Using only a window of time it's not possible to distinguish between days with several nucleation events and days with only one event, i.e., the day is always represented by the most intense nucleation event in the day (a 12 x 11 matrix)

The next step is to normalize the data. Before running data mining algorithms, it is beneficial to rescale each feature dimension of the observation set with whitening. Each feature is divided by its standard deviation across all observations to give it unit variance. In this case, the normalization process avoid too distinguish between days with intense events and days with normal events.

The transformed dataset can then be used for nucleation events classification using data mining algorithms.

3.4 Data Mining

Clustering methods attempt to find natural groups of data, according to the similarities among the samples. Typically the "similarity" concept is defined as the distance between a data matrix (x_{ij}) and the cluster prototype (center). In this work, each vector consists of the measures values for each day obtained in the previous process, and a clustering process is carried out in order to find relationships among the days and to extract prototypes. The characteristics of this prototypes are not known beforehand; they are chosen randomly and updated at the same time as the partitioning of the data is made. In crisp clustering, when a data partition is build, a single sample (x_{ij}) belongs to only one cluster. The fuzzy clustering extends this notion to express the membership degree of an observation (matrix x_{ij}) to all classes. Fuzzy clustering methods estimate the membership degree (or adequacy), $m_i \in [0, 1]$ of each data point i to a cluster (or class) j.

A well defined and known fuzzy clustering algorithm is the Fuzzy C-means [14]. Fuzzy c-means algorithm applies two procedures for the determination of the cluster centers and for the assignment of the data points to these centers using the Euclidean distance. This process is repeated iteratively till the centers are stabilized. When the iteration cycles are terminated the fuzzy clustering algorithm outputs a matrix of final cluster centers where each row provides the center matrix and final fuzzy partition matrix and values of objective function for each iteration. The clustering process stops when the objective function improvement between two consecutive iterations is less than the minimum amount of improvement specified, with the accuracy off 0.99. The cluster centers are used for the determination of its corresponding linguistic which of course describes the proper classification of a day (Event, No Event, Undefined).

3.5 Knowledge Representation

The goal of this step is to represent the prototypes. Every cluster represents a day prototype extracted from the data and each prototype is described by a fuzzy deformable prototype [9] that finally will be represented by a type 1 fuzzy set (Fig. 2). The established classification scheme labels a day as a particle-formation event, a nonevent, or, if ambiguous, undefined. Therefore, we consider three possible prototypes.

3.6 Classification

Days measures are classified by maximum membership method. The partition matrix and the fuzzy numbers give the extent to which each day belongs to different prototypes. From the partition matrix, the optimal cluster to which the day maximum belongs to is selected. For each day, the row number of largest element in each column of the partition
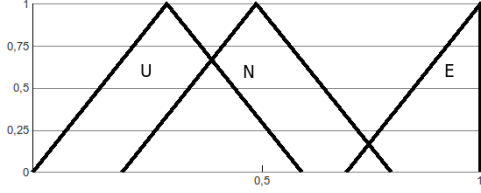
Fig. 2: Fuzzy Numbers. (U) Undefined, (N) No Event, (E) Event

matrix is found and the day is assigned the cluster center value, corresponding to that row. In this way, a X value is obtained, and from X, we obtain the degrees of membership to the prototypes represented by means of the fuzzy numbers. The definition of the prototype with the highest value of membership will be chosen as the most suitable for the analyzed day.

4. Experiment

In this section, the performance of the proposed classification method is evaluated in terms of the classification accuracy.

To assess the usability of this approach a software prototype, named ANuCla (**A**utomatic **Nu**cleation **Cla**ssification), has been developed in Python[1]. The tool provides an interface in order to monitor the whole knowledge discovery process. Once preprocessed the data is stored in a SQLite database[2]. A screenshot of the software tool and its componentes is shown in Fig. 3.
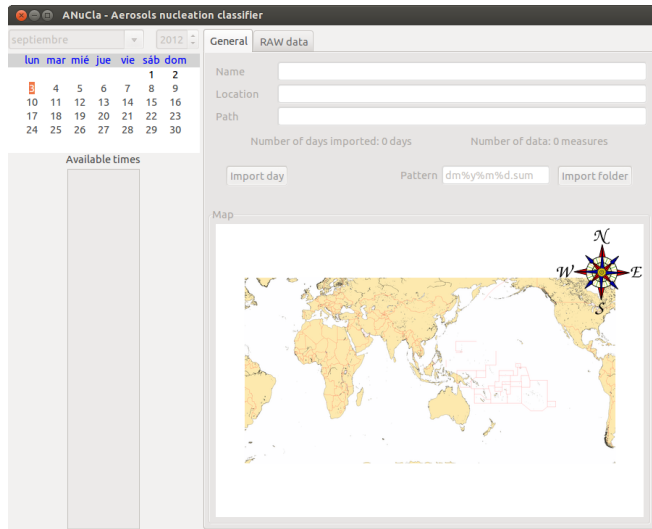
Fig. 3: Screenshot of the software tool

4.1 Data Set

Measurements used in this work were performed during the years 1996-2006 at the SMEAR II station, which is located in the $Hyyti\ddot{a}\ddot{a}l\ddot{a}$ Forestry Field Station of the University of Helsinki in southern Finland [13]. The core of the dataset used in this study are the total particle size distributions measured and the charged particle size distribution normalized according to its size ($\frac{dN}{dlogD_p}, cm^{-3}$). The undefined values (NaN) have been converted to 0.0. During the measurement period the measurements with the instruments were continuous, therefore, the raw datasets obtained display very sparse time series, several years of measurements every 10 min, with a large number of missing values and classes that are partially unbalanced (see Table 1).

Table 1: Data Set Features

Category	Number of Elements
With Events	973
Without Events	1477
Undefined	1557
Total	4007
No Data	58
Bad Data	235
Partially Bad	98

4.2 Perfomance Measures

As performance measures, we followed the standard definition of recall, precision, and F measure (the harmonic mean between precision and recall) [15]. For the evaluation of performance average across categories, we used the micro-averaging method.

Before formalizing these concepts we need to introduce the following notations. Given a set of categories $\Delta = \{\zeta_1, \ldots, \zeta_n\}$ and a set of days $\mathcal{D} = \{d_1, \ldots, d_m\}$ to be classified, we denote the set formed by the days $d_i \in \mathcal{D}$ classified as pertaining to a category $\zeta \in \Delta$ by C_ζ. Also, we denote the set of days $d_i \in \mathcal{D}$ assigned by an expert to the category $\zeta \in \Delta$ as E_ζ. Then, the precision (P_ζ), recall (R_ζ) and their F measure (F_ζ), w.r.t. a category ζ, are calculated by the following formulas.

$$P_\zeta = \frac{|C_\zeta \bigcap E_\zeta|}{|C_\zeta|} \quad (1) \qquad R_\zeta = \frac{|C_\zeta \bigcap E_\zeta|}{|E_\zeta|} \quad (2)$$

$$F_\zeta = \frac{2 * P_\zeta * R_\zeta}{P_\zeta + R_\zeta} \quad (3)$$

Observe that for a set S, as usual, $|S|$ denotes the cardinality of S. Therefore, $|C_\zeta|$ denotes the number of days which are classified as corresponding to category ζ by the classifier; analogously, $|E_\zeta|$ denotes the number of days whose assigned category is ζ, according to an expert pronouncement, and $|C_\zeta \bigcap E_\zeta|$ is the number of days which are correctly classified as pertaining to a category ζ by the classification method.

These preceding concepts can be generalized to a set of categories Δ. In this case, the precision (P), recall (R) and their F measure (F) w.r.t. Δ are calculated by the following formulas.

$$P = \frac{\sum_{\zeta \in \Delta} |C_\zeta \bigcap E_\zeta|}{\sum_{\zeta \in \Delta} |C_\zeta|} \quad (4) \qquad R = \frac{\sum_{\zeta \in \Delta} |C_\zeta \bigcap E_\zeta|}{\sum_{\zeta \in \Delta} |E_\zeta|} \quad (5)$$

$$F = \frac{2 * P * R}{P + R} \quad (6)$$

4.3 Results

Considering all the data over the performance of the proposed methods was 78% of F-Measure. Therefore, we can conclude the method performed well compared to traditional classification method (see Table 2 and Table 3) and similar than the self-organizing maps based approach proposed in [7]. As is in [7] occurs, the biggest disagreement between methods was found in the class of undefined days. This class is actually a merge of undefined days and outlier days, and there are not feature to describe this class.

Table 2: New Classification compared to the Manual Method

Proposal	Manual		
	Events	Non event	Undefined
Events	83%	6%	4%
Non event	10%	79%	21%
Undefined	7%	17%	75%

Table 3: Precision, Recall and F-Measure Results

Categories	Precision	Recall	F-Measure
Events	89%	83%	86%
Non event	71%	77%	74%
Undefined	76%	75%	75%

5. Conclusions and Future Work

In this article an application of Fuzzy Prototypical Knowledge Discovery for nucleation events classification clustering has been presented. The proposed fuzzy method showed to be an effective alternative to manual nucleation event classification. However there is drawback in this proposal when it is necessary to distinguish between days with more than a nucleation event, or between days with intense nucleation events. In the future we plan to use Gaussian fitting algorithms for feature reduction in order to improve the day representation.

Future investigations and experiments should consider the applications of this classification method on data provided from other meteorologic stations.

Acknowledgment

References

[1] R. H. Moss, J. A. Edmonds, K. A. Hibbard, M. R. Manning, S. K. Rose, D. P. van Vuuren, T. R. Carter, S. Emori, M. Kainuma, T. Kram, G. A. Meehl, J. F. B. Mitchell, N. Nakicenovic, K. Riahi, S. J. Smith, R. J. Stouffer, A. M. Thomson, J. P. Weyant, and T. J. Wilbanks, "The next generation of scenarios for climate change research and assessment," *Nature*, vol. 463, no. 7282, pp. 747–756, 2010.

[2] W. H. Organization, "Global health risks: mortality and burden of disease attributable to selected major risks." *World Health Organization*, vol. 2011, no. 9, p. 62, 2009.

[3] S. Solomon, D. Qin, M. Manning, Z. Chen, M. Marquis, K. B. Averyt, M. Tignor, and H. L. Miller, Eds., *Climate Change 2007 - The Physical Science Basis: Working Group I Contribution to the Fourth Assessment Report of the IPCC*. Cambridge, UK and New York, NY, USA: Cambridge University Press, Sept. 2007.

[4] S. Hyvönen, H. Junninen, L. Laakso, M. D. Maso, T. Grönholm, B. Bonn, P. Keronen, P. Aalto, V. Hiltunen, T. Pohja, S. Launiainen, P. Hari, H. Mannila, and M. Kulmala, "A look at aerosol formation using data mining techniques," *Atmospheric Chemistry and Physics*, vol. 5, pp. 3345–3356, 2005.

[5] S. Mikkonen, K. E. J. Lehtinen, A. Hamed, J. Joutsensaari, M. C. Facchini, and A. Laaksonen, "Using discriminant analysis as a nucleation event classification method," *Atmospheric Chemistry and Physics*, vol. 6, no. 12, pp. 5549–5557, 2006. [Online]. Available: http://www.atmos-chem-phys.net/6/5549/2006

[6] M. D. Maso, M. Kulmala, I. Riipinen, R. Wagner, T. Hussein, P. P. Aalto, and K. E. J. Lehtinen, "Formation and growth of fresh atmospheric aerosols: eight years of aerosol size distribution data from SMEAR II, Hyytiälä, Finland," *Boreal Env. Res.*, vol. 10, pp. 323–336, 2005.

[7] H. Junninen, I. Riipinen, M. Maso, and M. Kulmala, "An Algorithm for Automatic Classification of Two-dimensional Aerosol Data," in *Nucleation and Atmospheric Aerosols*, C. O'Dowd and P. Wagner, Eds. Springer Netherlands, 2007, pp. 957–961.

[8] L. A. Zadeh, "Fuzzy Logic, Neural Networks, and Soft Computing," *Commun. ACM*, vol. 37, no. 3, pp. 77–84, 1994.

[9] J. A. Olivas, "Applications of Fuzzy Prototypes: Forest Fire Prediction and Medical Diagnosis," in *Proceedings of the International Conference on Artificial Intelligence, IC-AI '99, June 28 - July 1, 1999, Las Vegas, Nevada, USA, Volume 2*, H. R. Arabnia, Ed. CSREA Press, 1999, pp. 662–668.

[10] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "The KDD Process for Extracting Useful Knowledge from Volumes of Data," *Commun. ACM*, vol. 39, no. 11, pp. 27–34, 1996.

[11] L. A. Zadeh, "A note on prototype set theory and fuzzy sets," *Cognition*, vol. 12, pp. 291–297, 1982.

[12] H. Bremerman, "Pattern recognition by deformable prototypes," *Structural stability, the theory of catastrophes, and applications in the sciences. Springer Notes in Math*, vol. 25, pp. 15–57, 1976.

[13] P. Hari and M. Kulmala, "Station for measuring ecosystem-atmosphere relations," *Boreal Environ. Res*, vol. 10, pp. 315–322, 2005.

[14] J. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984. [Online]. Available: http://dx.doi.org/10.1016/0098-3004(84)90020-7

[15] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, no. 4, pp. 427–437, July 2009. [Online]. Available: http://dx.doi.org/10.1016/j.ipm.2009.03.002

Correction Factor Optimization in User Profile Similarities

R. Seisdedos

Advanced Technical Faculty of Engineering ICAI
Pontifical Comillas University
Madrid, Spain
robertoseisdedos@upcomillas.es

C. Puente

Advanced Technical Faculty of Engineering ICAI
Pontifical Comillas University
Madrid, Spain
cpuente2@upcomillas.es

J. A. Olivas

Information Technologies and Systems Dept
University of Castilla-La Mancha
Ciudad Real, Spain
joseangel.olivas@uclm.es

Abstract — **The focus of this paper is aimed towards the field of the recommender systems. It will strive to propose a combination of some of the methods used to calculate the similarity among users adding some variations and tendency analysis. The final aim is to obtain more efficient similarity calculations and thus more accurate recommendations provided to the user. We have tested these methods in a personal recommender system of sport trips that we have developed. This system recommends the best matching offers to the user based on his characteristics and search preferences.**

Keywords: Fuzzy Logic, Correction factor, Similarity matrix, Fuzzy similarity degree

I.   INTRODUCTION

Ever since a new way of communication through the Internet appeared with the so-known social networks such as facebook, twitter and others, the usage of user profile based information has expanded widely. This type of information marked a turning point in different fields. It has been used to help prediction or decision systems to make predictions, recommender systems (such as Netflix or Pandora) to provide more accurate results to the users searches by means of analyzing not only the tendencies of the user searching, but also of similar users. Additionally it has been used as the basis to run different types of analysis and surveys by clustering and classifying the users by their similarity using some other clustering algorithms.

This type of information may vary completely among different applications due to the need of the specific pieces of information that each application needs. Despite the differences of the information, the calculation of the similarity has been done by different methods from the Jaccard index, the cosine-based similarity over expert systems [2] to frameworks that calculate the similarity of users using their location, like the *socialmatching* framework of Terveen et al. [1].

The idea that is to be shown in this paper expanded from the developing of a personal recommender system of sport trips. This idea occurred to us when facing the problem of calculating fuzzy similarities using already fuzzy characteristics of the users. The basic method corresponds to a variation of the Jaccard coefficient.

This recommender system bases its functionality on both, the information the user has provided the system explicitly such his age, sport mastership, etc. and the information the system creates with every use of itself. This means that the system that is being developed is a hybrid recommender system [3]. The pieces of information on which this system bases its functionality are the age, the number of sports the user likes, the budget the user wants to spend on a round trip to practice the selected sport, as well as the past tendencies the user registered while using the system and the distance.

The distance factor that has been used throughout the developing of the application is the so-called haversine distance, which could be interpreted as the distance between two points on a spherical surface.
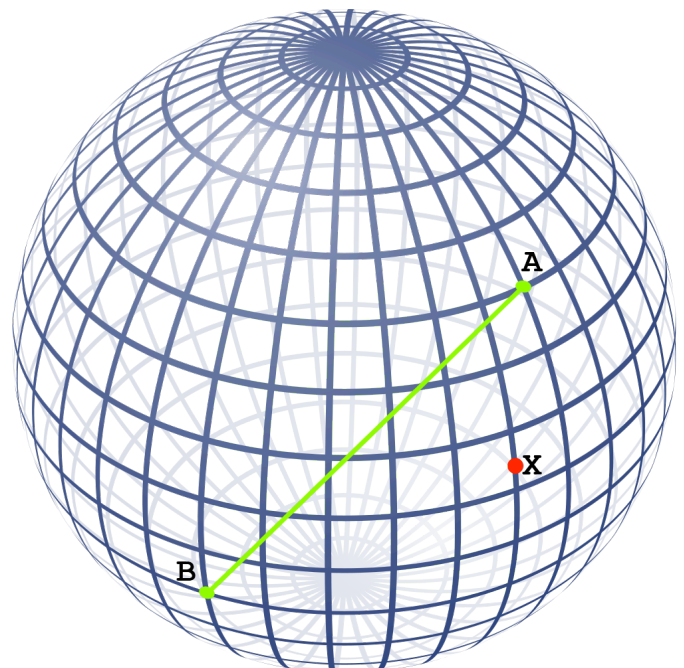
Figure 1: Haversine distance between two points

The calculation of this value incurs in the use of the coordinates of the user's location (latitude and longitude) as well as spherical trigonometry by using the following formula: [4]

$$\Delta\sigma = 2 * \arcsin\left(\sqrt{\sin^2\left(\frac{lat_a - lat_b}{2}\right) + \cos(lat_a) * \cos(lat_b) * \sin^2\left(\frac{long_a - long_b}{2}\right)}\right)$$

Equation 1: Haversine formula to calculate the spherical distance

II. USER REPRESENTATION BASICS

Some of the methods named above, such as the Jaccard index or the cosine-based distance, treat all this information in Euclidian form, meaning that each piece of information might be taken out of its general meaning and use it to locate it in coordinated system of n dimensions, where n is the number of properties the system needs the user to declare. Thus the user would be represented as a point in the corresponding Euclidian space. For instance, in Figure 2, the user A has been represented as a point in a 2 dimensional space with the coordinates (X_a, Y_a).
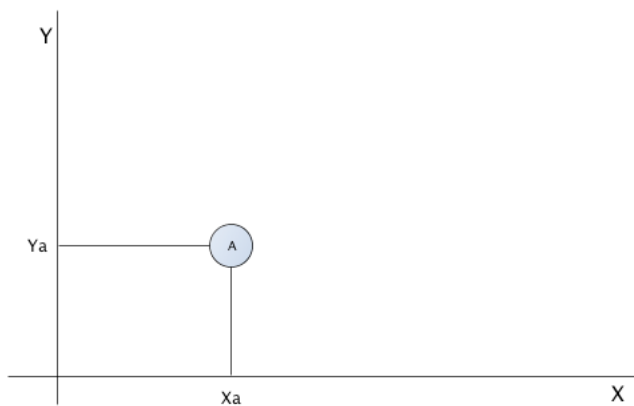
Figure 2: 2-dimension representation of a user.

If there is more than just one point in the Euclidian space it is possible to calculate the vector that links both of these points, and also the length of the vector, which is known as the modulo. This is represented in Figure 3, where the red arrow stands for the vector that unites both of the users.

III. USER SIMILARITY ALGORITHM

Using the basis of the user representation in an Euclidian space of 5 dimensions, one for each characteristic, the users are represented, thereafter it is possible to calculate the vector-distance among them, as well as the cosine-based distance.

However, as each one of the characteristics of the user are typed differently, they all have to be categorized between the values 0 and 1 to be able to determine via a more standard and objective method how similar or dissimilar the two users, just

as in fuzzy sets, where 1 stands for sets that are completely similar and 0 for dissimilar sets.

Once the differentiation vector has been calculated using the basic vector calculation and categorizing all of the coordinates within the values 0 and 1, the final similarity degree is calculated. To categorize each coordinate, the maximum value has to be set depending on the variable magnitude.

In this case, the basic idea to calculate it has been taken from the Jaccard index. To obtain the Jaccard similarity index, the number of sets both elements have in common is divided by the total number of sets that result from the union of both elements. This can be translated by the formula:

$$JS = \frac{|A \cap B|}{|A \cup B|} = \frac{S_{11}}{S_{10} + S_{01} + S_{11}}$$

Equation 2: Jaccard similarity index

Taking this idea as a starting point the final degree is calculated. The difference between the Jaccard index and the method that has been used, is that the two elements that are to be analyzed are similar in all the different sets but in in different degrees thus being fuzzy.Whereas, on the other hand, the Jaccard distance calculates the similarity over binary sets where the elements are similar or not.

So taking this into consideration, the method to calculate the final simililarity degree would have been a simple average calculus that would always result in the value 1. This is why, some alterations have been proposed to calculate this value.

To finally be able to retrive this derived information from the users, a weighed average is used to alter the weight of each one of the characteristics of the differentiation vector in the final result of the similarity degree in the system. The final calculation would be then obtained using the following formula:

$$GS(A, B) = \frac{\sum_{i=1}^{5} X_{i, \overrightarrow{v_{(A,B)}}} * P_{i, \overrightarrow{v_{(A,B)}}}}{\sum_{i=1}^{5} P_{i, \overrightarrow{v_{(A,B)}}}}$$

Equation 3: Final similarity degree formula

These weights or correction factors have a minimum value of 0 and a maximum value of 1 and can be set as default and use always the same or use predictive methods such as neural networks to estimate the best values for each of the different characteristics. They can also be categorized depending on the uncategorized value of the coordinates of the vector.
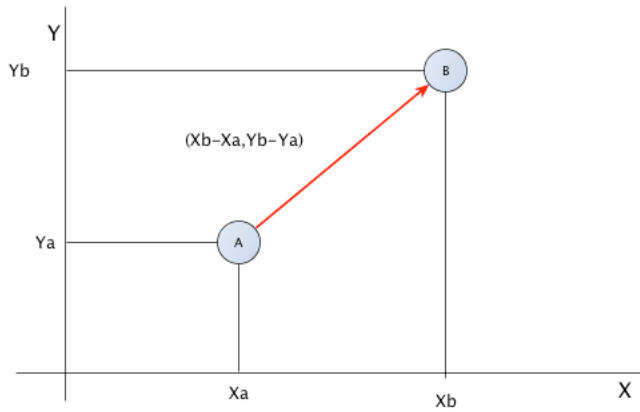
Figure 3: Vector between two users.

This is the basic algorithm to calculate the final similarity degree the algorithm calculates the upper triangle of the similarity matrix.

This value is used afterwards to add more efficiency in the recommendations of the system to the user.

III. CORRECTION FACTOR OPTIMIZATION

As described above, the similarity degree between users is calculated using a weighed value or correction factor to determine how important a characteristic is in the current state of the system. These factors can be used as an invariable default value. However this approach may lead to default ratings over time instead of being able to improve the result efficiency as time goes by.

These factors can be adapted to each one of the characteristics while calculating how important these factors are towards the similarity of every pair of users and the characteristic that is being altered.

The following algorithm will explain how this approach may be implemented in a system.

Step 1: Calculate the similarity degree as described above with the default value set for the system. In this case we use as the default correction factor the value 1.

Step 2: Now it is time to calculate the new correction factor to optimize the similarity degree calculation by averaging each one of the characteristics by user's average similarity in them (we will call this value Rogue Correction Factor (RCF)).

The average will be calculated considering the previous similarity degree and the uncategorized similarities in each of the characteristics for each user towards the others and the average uncategorized similarity of that user towards the rest in the characteristic that is being evaluated. This means that, in the end, the new correction factor will be calculated for each pair of two users, and the formula will be like:

$$RCF_c = \frac{SD(U_i, U_j) * C(U_i, U_j)}{\sum_{k=0}^{N} C(U_i, U_k)}$$

Equation 4: Category fuzzy value calculation

Step 3: After completing step 2, there will be a rogue correction factor for each user in each characteristic because every user has different similarity values towards the rest of users each characteristic.

Due to the fact that the correction factors that will be applied to calculate the updated similarity degree between two users have to be symmetric, (meaning that it has to be the same to calculate the similarity degree between $user_a$ and $user_b$ and vice versa), the final correction factors are obtained by averaging the rogue factors of the users in question for each characteristic.

$$CF_{c,U_{i,j}} = \frac{RCF_{c,U_{i,j}} + RCF_{c,U_{j,i}}}{2}$$

Equation 5: Final Correction Factor calculation

Step 4: After the final correction factor matrixes have been calculated, the new similarity degree is calculated in the same way it was calculated the first time. However now the correction factor over each characteristic may vary for each pair of users, adapting the fuzzy degree of similarity for each user.

IV. PRESENTATION OF THE EXAMPLE

In this paper, we are going to show how this algorithm works calculating the similarity degree among 5 different users. The description of the different characteristics that are going to be used goes as follows:

- Latitude and longitude: used to calculate the position of a person on the Earth's surface, and also to calculate the distance to other users. To categorize afterwards the values, the maximum value that is used is the distance between two antipodes.
- Age: this characteristic is taken into account because the older the user is the bigger its budget may be. Here the maximum difference in age will be 40 years.
- Budget: even though this characteristic is somehow already considered when considering the age, it is used to add more differentiation to users of similar age. The maximum value for this characteristic is more subjective regarding the average ages of the users and depending on to whom the application is

targeted. In this case we used as the maximum value 1500 €.

- Number of sports: that the user likes/practices. Here the maximum value is the maximum number of sports the application gives recommendations for minus 1. Thus he value will be 8.
- Past tendencies: referring to the number of times the user has accepted an offer of this system, being the maximum value the number of times the user that has used the system the most. This value is expected to vary along with the use of the system. However in this case we will use as the maximum value 5.

The different user's values on each and every one of these characteristics are:

	Lat	Long	Age	Budg	Sprt	Tnd
Usr1	40.4537	-3.8062	22	360	3	5
Usr2	41.3850	2.17340	22	500	2	0
Usr3	43.3621	-8.40916	35	900	4	3
Usr4	37.3880	-5.98233	30	760	3	4
Usr5	40,4207	-3.68817	43	900	1	2

Table 1: User description

V. EXAMPLE DEVELOPMENT

First of all, it is necessary to calculate the basic similarity degree among the users in the fashion described above in **Step 1**. Obtaining the results showed in Table 2.

	User 1	User 2	User 3	User 4	User 5
User 1	*1*	*0,7512*	*0,7530*	*0,8628*	*0,6529*
User 2		*1*	*0,7027*	*0,0732*	*0,7316*
User 3			*1*	*0,8844*	*0,8399*
User 4				*1*	*0,7824*
User 5					*1*

Table 2: Basic user similarity degrees

Once the basic fuzzy degrees have been calculated, it is possible to calculate the rogue correction factors for each user and characteristic, as shown in **Step 2**. In this example, we will present how it is done only for a user in a characteristic (the distance) and then show the results of the remaining characteristics:

$$RCF_{D,U_1} = \frac{512,86 * 0,7512 + 499,53 * 0,7530 + 389,38 * 0,8628 + 10,64 * 0,6429}{512,86 + 499,53 + 389,38 + 10,64} = 0,7818$$

Equation 6: Rogue correction factor calculation for user 1 in the distance characteristic

The remaining results are shown in the following table:

	Dist	Age	Budget	Sports	Tend
User 1	*0,7819*	*0,7239*	*0,7466*	*0,7025*	*0,7349*
User 2	*0,5267*	*0,5973*	*0,5816*	*0,5923*	*0,5443*
User 3	*0,7878*	*0,7709*	*0,7514*	*0,7946*	*0,7626*
User 4	*0,5716*	*0,6494*	*0,6356*	*0,6306*	*0,4506*
User 5	*0,7841*	*0,7296*	*0,6988*	*0,7652*	*0,7283*

Table 3: Rogue correction factors for each user and characteristic

After these values have been calculated, it is possible to obtain the symmetric final correction factors by averaging the rogue values for each pair of users in every characteristic:, like in **Step 3**:

Dist	User 1	User 2	User 3	User 4	User 5
User 1	0	0,6543	0,7848	0,6767	0,7830
User 2		0	0,6573	0,5492	0,6554
User 3			0	0,6797	0,7859
User 4				0	0,6778
User 5					0

Table 4: Correction factor for the distance

Age	User 1	User 2	User 3	User 4	User 5
User 1	0	0,6606	0,7474	0,6867	0,7267
User 2		0	0,6841	0,6234	0,6634
User 3			0	0,7102	0,7503
User 4				0	0,6895
User 5					0

Table 5: Correction factor for the age

Budget	User 1	User 2	User 3	User 4	User 5
User 1	0	0,6641	0,7490	0,6911	0,7227
User 2		0	0,6665	0,6086	0,6402
User 3			0	0,6935	0,7251
User 4				0	0,6672
User 5					0

Table 6: Correction factor for the budget

Sports	User 1	User 2	User 3	User 4	User 5
User 1	0	0,6474	0,7486	0,6666	0,7339
User 2		0	0,6935	0,6115	0,6788
User 3			0	0,7126	0,7799
User 4				0	0,6979
User 5					0

Table 7: Correction factor for the sports

Tend.	User 1	User 2	User 3	User 4	User 5
User 1	0	0,6396	0,7488	0,5927	0,7316
User 2		0	0,6535	0,4975	0,6363
User 3			0	0,6066	0,7455
User 4				0	0,5895
User 5					0

Table 8: Correction factor for the tendencies

Finally, it is possible to calculate the final similarity degrees using this newly obtained information as the weights of each one of the characteristics and pair of users like: (**Step 4**)

$$SD(U_1, U_2) = \frac{0,6543 * 0,9744 + 0,6606 * 1 + 0,6641 * 0,875 + 0,6474 * 0,9067 + 0,6396 * 0}{0,6543 + 0,6606 + 0,6641 + 0,6474 + 0,6393}$$

$$= 0,7553$$

Equation 7: Final fuzzy degree calculation for the users 1 and 2

In the end the final fuzzy similarity degrees will be as follows:

	User 1	User 2	User 3	User 4	User 5
User 1	*1*	*0,7553*	*0,7542*	*0,8632*	*0,6580*
User 2		*1*	*0,7034*	*0,7483*	*0,7328*
User 3			*1*	*0,8861*	*0,8385*
User 4				*1*	*0,7863*
User 5					*1*

Table 9: Final fuzzy similarity degrees

VI. CONCLUSIONS AND FURTHER WORK

This approach has high computational requirements because it needs to calculate several different matrixes depending on how many different characteristics are being evaluated. After all the matrixes have been obtained they are then combined to obtain the final values for the correction factors, and afterwards the final fuzzy degree can be calculated.

This is why in our system this procedure is done offline in batch mode to be able to give the most accurate results, and thus improve the recommendations over the searches of a user.

To recommend the users regarding their characteristics we calculated the similarity of a user to the different offers that have been generated and the tendencies of users similar to him, ranking the results in the very end in descending order of similarity.

However this approach may be implemented in any user-profile based recommender system to improve the recommendations that are not only based on the user characteristics but also on the tendencies of similar users to him.

This approach could also be modified to add clustering algorithms, such as spectral clusters [8], to classify the users first, and then calculate the average similarities between the clusters and take that similarity degree into consideration when obtaining the Rogue Correction Factors and the final fuzzy degrees.

REFERENCES

[1] Terveen, L.G., McDonald, D.E.: Social matching: A framework and research agenda. ACM Trans. Comput. –Hum.Interact,401-434 (2005)

[2] Ehrlich, K., Lin, C.Y., Griffiths-Fisher, V.: Searching for experts in the enterprise: combining text and social network analysis. In: International ACM SIGGROUP Conference on Supporting Group Work, pp. 117–126 (2007)

[3] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering 17 (6) (2005) 734–749.

[4] Isaac Todhunter: Spherical Trigonometry: For the Use of Colleges and Schools. Macmillan & Co. 1863

[5] L.A. Zadeh, Fu KS, Tanaka k, Shimura M, Calculus of fuzzy restrictions; Fuzzy Sets and their Applications to Cognitive and Decision Processes (1975) 1-39

[6] M.J. Lee, C.W. Chung A user similarity calculation based on the location for Social Network Services

[7] J. Serrano-Guerrero, E. Herrea-Viedma, J.A. Olivas, A. Cerezo, F.P. Romero, A google waved-based fuzzy recommender system to disseminate information in University Digital Libraries 2.0 (2011)

[8] A. N. Ng, M. I. Jordan, Y. Weiss, U.C. Berkeley, The Hebrew University, On spectral clustering: Analysis and Algorithm

SESSION

LEARNING AND APPLICATIONS - IN PARTICULAR, AS IT RELATES TO MAPREDUCE AND PHISHING DETECTION

Chair(s)

Dr. Fadi Thabtah

Gentle Early Detection Algorithm

Hussein Abdel-jaber[1], Mahmoud Abdeljaber[2], Hussein Abu Mansour[1], Malak El-Amir[1]

[1]Faculty of Computer Studies, Department of Information Technology and Computing, Arab Open University, Kingdom of Saudi Arabia, {habdeljaber, hmansour, melamir}@arabou.edu.sa

[2]Department of Computing, Philadelphia University, Jordan, mabdel@gmail.com

Abstract - *The developments in computer networks in recent days such as the internet have increased rapidly. Connections of these networks necessitate resources in order to send their data to their prospective destinations. Further, the connections require high speed router buffers which they route data in high speed. Congestion is one of the main issues that occur at the router buffer cause deterioration of the network performance, i.e. increasing average waiting time, decreasing throughput, etc. Gentle Random Early Detection (GRED) is one of the known congestion control algorithms proposed to detect congestion before the router buffer overflows. In fact, GRED improves the setting of the parameters for the maximum threshold position ($\max threshold$) at the router buffer and the maximum value for the packet dropping probability ($D_{\max}$). This paper presents an Adaptive GRED algorithm that detects congestion at router buffers in an preliminary stage, and enhances the parameters setting of the $\max threshold$ and the $D_{\max}$. During congestion, the simulation results reveal that the Adaptive GRED drops fewer packets than GRED, and it marginally offers better performance results than either GRED or BLUE-Linear analytical model with regard to mean queue length and average queuing delay when heavy congestion is existed.*

Keywords: Congestion Control, Gentle Random Early Detection (GRED), Adaptive GRED, Parameter Settings.

1. Introduction

With the rapid growth of computer network technologies, especially internet, congestion becomes one of the main issues that may affect the network's performance [13, 14]. Congestion occurs at router buffers of networks when the amount of arriving packets is larger than the available resources [12]. This can cause poor performance for computer networks [7] including the following:

- Increasing both the (mql) and the (D) for packets.
- Degrading the amount of packet that are passed through the router buffer successfully (T).

- Growing the P_L result.
- Increasing the packet dropping probability (D_p)

In order to improve the network performance, a group of congestion control algorithms called Active Queue Management (AQM) have been proposed [1, 2, 3, 4, 5, 6, 8, 9, 10, 11]. Examples of AQM algorithms are Random Early Detection (RED) [9], Gentle RED [11], Adaptive RED [10], Random Early Marking (REM) [5], Dynamic Random Early Drop (DRED) [6] and some discrete-time queue analytical models [1, 2, 3, 4] which are built based on some of AQM methods. For instance, the analytical models of DRED, GRED and BLUE were built by analysing a single queue node using a discrete time queues approach [15].

One of the most known AQM methods is RED. Though, the performance of RED may deteriorate in the following many situations. For example, at a certain time, the arrival rate could increase, and as a consequence the RED router buffer builds up and overflows. The congestion measure of RED (aql) could be smaller than the minimum threshold position at the router buffer ($\min threshold$). Consequently, no packets will be dropped even though the RED router buffer is overflowing. Another obstacles of RED is it's dependency on the input setting ($\min threshold$, $\max threshold$, queue weight (qw), $D_{\max}$) [9]. These parameters must set to certain values in order to derive a satisfactory performance [7, 10]. One possible solution to preventing RED's router buffer to build up quickly is to utilise the instantaneous queue length as a congestion detector rather than the aql. This gives the instantaneous queue length the opportunity to exceed the $\min threshold$ position, and to drop packets probabilistically before the router buffer overflows. DRED, BLUE and GRED algorithms are using the instantaneous queue length as a congestion detector. GRED algorithm was proposed primarily to deal with the abovementioned limitations of RED. Precisely, GRED improves the process of setting the $\max threshold$ and the $D_{\max}$ parameters, and stabilises the aql at a position named T_{aql}. The T_{aql} position is half way from the $\min threshold$ and $\max threshold$ locations. Thus preventing the router buffer from filling up and becoming larger than the $\max threshold$ position, and as a result fewer packets are dropped.

This paper presents some AQM algorithms such as GRED [11], Adaptive GRED [1] and BLUE-Linear analytical model [3] which aims to enhance the performance of GRED with reference to mql, D and P_L performance measures particularly during congestion. Unlike GRED, the D_{init} value of the Adaptive GRED varies from D_{max} to 0.5 as long as the aql varies from $\max threshold$ to $double \max threshold$, whereas the D_{init} value of GRED varies from D_{max} to 1.0 as long as the aql value varies from $\max threshold$ to $double \max threshold$. This enables Adaptive GRED of providing further enhancements in setting the input parameters, e.g. $\max threshold$ and D_{max}.

The organisation of this paper is as follows: Section 2 presents the GRED algorithm and Adaptive GRED is introduced in Section 3. Section 4 presents the BLUE-Linear analytical model. The simulation information of GRED, Adaptive GRED and BLUE-Linear is provided in Section 5. Section 6 compares the performance evaluation results of GRED, Adaptive GRED and BLUE-Linear, and finally, conclusions and future work are given in Section 7.

2. Gentle Random Early Detection (GRED) Algorithm

GRED has been proposed in [11] to deal with some of RED's issues [6, 10, 11] discussed in the previous section. The pseudocode of GRED is presented in Figure 1, and description (definition) of the parameters used in the pseudocode is defined as follows:

Definition 1: $current\_time$**:** the current time.

Definition 2: $idle\_time$**:** the beginning idle time at the router buffer.

Definition 3: n**:** the number of packets transmitted to the router buffer through an idle interval time.

Definition 4: C**:** a counter that represents the number of packets arrived at the router buffer and have not dropped since the last packet was dropped.

Definition 5: D_p **:** the packet dropping probability.

Definition 6: D_{init} **:** the initial packet dropping probability.

Definition 7: q_instantaneous**:** the instantaneous queue length.

Definition 8: qw : the queue weight.

Definition 9: D_{max} : the maximum value of D_{init}.

Definition 10: $q(time)$**:** the linear function for the time.

Definition 11: T_{aql} **:** target level for the aql.

Definition 12: $double \max threshold$ is set to $2 \times \max threshold$.

    Initialisation stage
    $C = -1$;
    $aql = 0.0$;
    for every arriving packet at a GRED router buffer:
    2.1 Calculate the aql for the arriving packet at the router buffer.

2.2 Examine the queue status at the router buffer, e.g. empty or not.
   if (the queue at the router buffer $==$ empty)
    {
      Compute n, where $n = q(current\_time - idle\_time)$;
$$aql = aql \times (1 - qw)^n ;$$
    }
  $else$
   $aql = aql \times (1 - qw) + qw \times q\_instantaneous$;

Determining the congestion status at the router buffer:

$if(aql < \min threshold)$
  {
    $D_p = 0.0$; // No packets have dropped
    Set $C = -1$;
  }

$elseif(\min threshold \le aql \&\& aql < \max threshold)$
  {
    $C = C + 1$;
      calculate the D_p value for the arriving packet as
      follows:
$$D_{init} = \frac{D_{max} \times (aql - \min threshold)}{(\max threshold - \min threshold)};$$
$$D_p = \frac{D_{init}}{(1 - C \times D_{init})};$$
  drop arriving packet probabilistically in terms of its (D_p) value;
      Set $C = 0$;
    }
$elseif(\max threshold \le aql \&\& aql < double \max threshold)$

  {
    $C = C + 1$;
  calculate the dropping probability (D_p) for the arrival packet as
      follows:
$$D_{init} = D_{max} + \frac{(1 - D_{max}) \times (aql - \max threshold)}{\max threshold};$$
$$D_p = \frac{D_{init}}{(1 - C \times D_{init})};$$
Mark/drop arriving packet probabilistically in terms of its (D_p) value;
      Set $C = 0$;
    }
   $else$ // $if(aql \ge double \max threshold)$
    {
      Mark/drop every arriving packet with $D_p = 1$;
      Set $C = 0$;
   }

When the GRED router buffer becomes empty
Set $idle\_time = current\_time$;

Figure 1: The pseudo code of GRED algorithm

2

Figure 1 shows that whenever a packet arrives at the router buffer, the aql value is calculated as in RED [9], which it is given in the following equation:

$$aql = aql \times (1 - qw) + q\_instantaneous \times qw \ldots\ldots (1)$$

If the aql value is smaller than the $\min threshold$ position, no packets will be dropped, and the D_p value is set to zero. On the other hand, if the aql value is larger than or equal to $double \max threshold$ position, a heavy congestion occurs, and subsequently every arriving packet will be dropped with $D_p = 1$. Finally, when the aql value is between $\min threshold$ and $double \max threshold$ positions, a congestion occurs, and the router buffer drops the arriving packets probabilistically ($0 < D_p < 1$). The result of D_{init} varies between D_{max} and 1 as long as the aql value varies between $\max threshold$ and $double \max threshold$ positions [11]. This variance produces further tune to the parameter settings of $\max threshold$ and D_{max}. The parameters of GRED, i.e. qw and $\min threshold$ have been set as in RED.

3. Adaptive GRED Algorithm

An adaptive GRED algorithm is proposed to improve the performance of GRED during congestion situations [1], i.e. deriving better quality results with reference to mql, D, and P_L performance measures. In addition, Adaptive GRED aims to enhance the parameter settings [1], e.g. $\max threshold$ and D_{max}, of GRED. The Adaptive GRED drops arriving packets incipiently during the congestion. The congestion measure of the Adaptive GRED is aql, and it's parameters are the similar to those of GRED. The calculation of the aql in Adaptive GRED is also similar to that of GRED. Therefore, the Adaptive GRED decides whether to drop every arriving packet as in the GRED algorithm (see Figures 1 and 2). Figure 2 shows the pseudocode of the Adaptive GRED.

1. Initialization stage
   $C = -1$;
   $aql = 0.0$;
2. For every arriving packet at an Adaptive GRED router buffer:
   2.1 Calculate the aql for the arriving packet at an Adaptive GRED router buffer.
   2.2 Examine the queue status at the router buffer whether empty or not.
   if (The queue at the Adaptive GRED router buffer == empty)
   {
   Compute n,
   where $n = q(current\_time - idle\_time)$;
   $$aql = aql \times (1 - qw)^n;$$
   }
   $else$

$$aql = aql \times (1 - qw) + qw \times q\_instantaneous;$$

Determine a congestion status at the Adaptive GRED router buffer:
$if (aql < \min threshold)$
{
$D_p = 0.0$; // No packets have dropped
Set $C = -1$;
}

$elseif (\min threshold \leq aql \,\&\&\, aql < \max threshold)$
{
$C = C + 1$
Calculate the D_p value for the arriving packet as follows:

$$D_{init} = \frac{D_{max} \times (aql - \min threshold)}{(\max threshold - \min threshold)};$$
$$D_p = \frac{D_{init}}{(1 - C \times D_{init})};$$

Mark/drop arriving packet probabilistically in terms of its (D_p) value;
Set $C = 0$;
}
$elseif (\max threshold \leq aql \,\&\&\, aql < double \max threshold)$

{
$C = C + 1$;
Calculate the marking/dropping probability (D_p) for the arrival packet as Follows:

$$D_{init} = D_{max} + \frac{\frac{(1 - D_{max})}{2} \times (aql - \max threshold)}{\max threshold};$$

$$D_p = \frac{D_{init}}{(1 - C \times D_{init})};$$

Mark/drop arriving packet probabilistically in terms of its (D_p) value;
Set $C = 0$;
}
$else$ // $if (aql \geq double \max threshold)$
{
Mark/drop every arriving packet with $D_p = 1$;
Set $C = 0$;
}

When the Adaptive GRED router buffer becomes empty
Set $idle\_time = current\_time$;

Figure 2: The pseudocode of Adaptive GRED method in detail.

3

The main difference between GRED and the Adaptive GRED is in calculating the D_{init} value. The way of computing D_{init} in GRED is shown in Figure 1. Whereas the Adaptive GRED computes the D_{init} according to the following equation:

$$D_{init} = D_{max} + \frac{(1 - D_{max})}{2} \times \frac{(aql - max\,threshold)}{max\,threshold} \quad\text{................. (2)}$$

To clarify, when aql value is between $max\,threshold$ and $double\,max\,threshold$ positions, the calculated D_{init} value of GRED varies from D_{max} value to 1.0 as the aql value varies from $max\,threshold$ to $double\,max\,threshold$ position. However, in the Adaptive GRED, the D_{init} value increases from D_{max} value to 0.5 as long as the aql value increases from $max\,threshold$ to $double\,max\,threshold$ position. This gives more improvements for the parameter settings of $max\,threshold$ and D_{max} than those of GRED.

Furthermore, the Adaptive GRED also offers better P_L performance as well as slightly better performance of mql and D than those of GRED when a high congestion has occurred. Lastly, the running time of the Adaptive GRED is the amount of time of all events. i.e. packet arrivals and departures.

4. BLUE-Linear Analytical Model

A discrete-time queue analytical model based on BLUE and discrete-time queues mechanism [15] is introduced in this section, which called BLUE-Linear analytical model [3]. This model aimed to identify congestion in an early stage, which means before the router buffer has overflowed [3]. BLUE-Linear model uses queue length (ql) as a congestion measure, and it relies upon a single threshold (th) placed at the router buffer. When ql is less than or equal to th value, no congestion is happened. Therefore, no packets are dropped. On the other hand, when ql value is above th value, congestion is arisen, thus the value of packet arrival probability (α) will be decreased linearly with aiming at controlling congestion, and also the packet dropping probability will be increased linearly from 0 to $\left(\frac{\alpha 1 - \alpha_i}{\alpha 1}\right)$ as the ql increases from $th+1$ to K. The increasing of α depends on the state of queue, and the following equation is used for computing α value at each queue state (α_i):

$\alpha_i = \alpha 1 - (1 + i - th)\frac{\alpha 1}{(1 + K - th)}$, if $i > th$. α_i depends on the queue state i. The following parameters are used in BLUE-Linear model: $\alpha 1$ and α_i denotes the packet arrival probability in a slot when the ql is less than or equal to th and larger than th position at the router buffer, respectively. β is

the packet departure probability in a slot. th, is a particular position at the router buffer. K is the finite capacity of the router buffer. The performance measures of BLUE-Linear model are provided as follows:

Mean queue length:

$$mql = P^{(1)}(1) = \frac{p_0}{(1-\beta)}\left[\frac{\gamma - \gamma^{th+1}[1 + th(1-\gamma)]}{(1-\gamma)^2} + \gamma^{th+1}(1-\alpha 1)\sum_{i=th+1}^{K}\prod_{j=th+1}^{i-1}(\gamma_j)\frac{i}{(1-\alpha_i)}\right],$$

where $\gamma_j = 1$ and $j \le th$ .... (3)

Throughput:

$$T = \beta\sum_{i=1}^{K}\Pi_i = \beta(1 - p_0) \text{ Packets/slot... (4)}$$

Average Queuing Delay:

$$D = \frac{mql}{T}\,slots = \frac{P^{(1)}(1)}{T}\,slots = \frac{\sum_{i=0}^{K}i \times p_i}{T}\,slots\text{............................. (5)}$$

Overflow Packet Loss Probability:

$$P_L = (1 - \beta)p_K \text{ .... (6)}$$

Packet Dropping Probability:

$$D_p = \sum_{i=th+1}^{K-1}\left(\frac{\alpha 1 - \alpha_i}{\alpha 1}\right) \times p_i \text{ .... (7)}$$

5. Simulation Information of GRED, Adaptive GRED and BLUE-Linear

It has been assumed that α denotes the packet arrival probability at the router buffer in a fixed time unit called slot [15], and β denotes the probability of packet departure from the router buffer in a slot. The packet arrivals can be modeled using a Bernoulli process and packet departures can be modeled using a geometrically distribution. The geometrically distributed means $\frac{1}{\alpha}$ and $\frac{1}{\beta}$ are used for the packet inter-arrival times and service times, respectively. BLUE-Linear is built and both GRED and Adaptive GRED are simulated based on discrete-time queue [4, 15] that uses a slot as a time unit. In each slot, packet arrival and/or departure may exist. The compared algorithms are simulated by applying them in a network consisting of a single router buffer node (see Figure 3), and also BLUE-Linear is built by analyzing this single router buffer node that shown in Figure 4. Packet arrival and departure are implemented in a single mode. The scheduling mode is First Come First Served (FCFS). The GRED and the Adaptive GRED simulations are implemented in Java, on i7 processor machine with 1.66 GHz, and on 4 GB RAM. In the same environment the results of BLUE-Linear are computed.

4
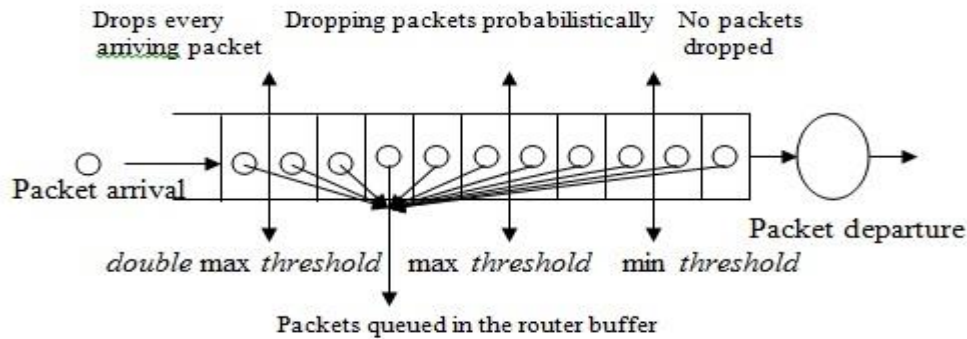
Figure 3: The single router buffer for GRED and Adaptive GRED.
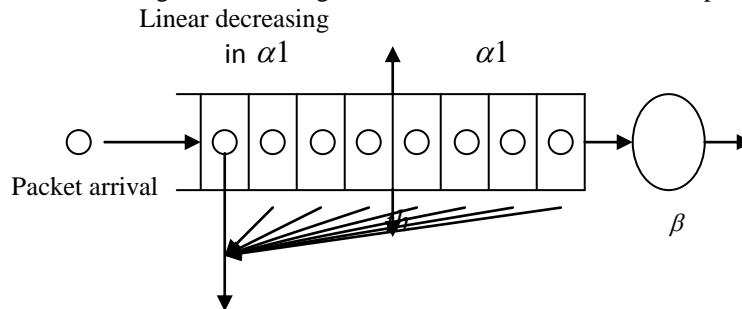
Packets queued in the router buffer.
Figure 4: The single queue node system for the BLUE-Linear model.

6. Performance Evaluation Results

This section compares GRED, Adaptive GRED and BLUE-Linear with reference to different performance measures , e.g. (mql, T, D, P_L, D_p), in order to determine which algorithm offers better performance. The parameters of compared methods are similar and have been set as follows: α and β were set to [0.18-0.93] and 0.5 respectively aiming to create noncongestion ($\alpha < \beta$) and congestion ($\alpha > \beta$) situations. The maximum number of queue node rooms (K) is finite and equal to 20, and this number can should generate accurate performance measure results at small queue sizes. $\min threshold$, $\max threshold$, D_{max} and qw have been set to 3, 9, 0.1 and 0.002, respectively as in RED [9]. $double \max threshold$ is set to 18 as in GRED [11]. th was set to $0.6 \times K = 12$. In the simulation, a large number of slots have been used (2000000) in order to generate a warming-up period. The warming-up period ends when the system reaches a steady state.

The performance measure results are evaluated by varying the values of α. Hence, the decision which method provides better performance is only given based on the values of α. After the system reaches the steady state, the evaluations of performance measures can be achieved. For each α, the simulations are run ten times in which in each run, the seed value for the random number generator is changed aiming to delete a prejudice in the performance measure results. Also, the performance measure result for each α represents the mean of ten run times for that value. All performance measure results versus α values are shown in Table 1 and Figures (5-6). Table 1 displays mql, T and D results of the GRED and the Adaptive GRED versus α. While the results of P_L and D_p versus α are illustrated in Figures 5 and 6, respectively.

Table 1 shows that the GRED and the Adaptive GRED give similar mql, T and D results when no or light congestion ($\alpha < \beta$) is occurring. Also, specifically, when α value is less than 0.48, mql and D results of GRED and Adaptive GRED are somewhat smaller than those of BLUE-Linear analytical model, therefore, their results are somewhat better than those of BLUE-Linear. On the other hand, if heavy congestion occurs ($\alpha > \beta$), the Adaptive GRED offers slightly better mql and D results than those of GRED. This is since the Adaptive GRED's router buffer drops more packets than that of GRED in these circumstances, i.e. $\alpha > \beta$ (see Figure 6). Moreover, when $\alpha > \beta$, GRED and Adaptive GRED offer better performance results with reference to mql

5

and D than those of BLUE-Linear. In case of existence of moderate congestion, i.e. $0.63 \geq \alpha \geq 0.48$, BLUE-Linear provides less mql and D results than their corresponded results for either GRED or Adaptive GRED. It is observed from Table 1 that the compared algorithms except BLUE-Linear generate similar T results regardless the congestion situation. However, at any situation BLUE-Linear present marginally higher T results than those of GRED or Adaptive GRED due to BLUE-Linear's router buffer serves marginally more packets than the router buffer of either GRED or Adaptive GRED. Figure 5 depicts that the Adaptive GRED loses fewer packets due to overflow (P_L) than GRED in congestion situations, and this is due to that it drops larger number of packets (D_p) than GRED (see Figure 6). This gives marginally smaller results of mql and D for Adaptive GRED than those of GRED. GRED's router buffer maintains it's aql at value lower than that of the Adaptive GRED when a heavy congestion has occurred. Therefore, both algorithms offer similar P_L and D_p results when no congestion or light congestion is present. Finally, BLUE-Linear loses and drops number of packets similar to those of GRED and Adaptive GRED when no congestion or light congestion is occurred. Whereas, if $\alpha > \beta$ BLUE-Linear loses fewer amount of packets than either GRED or Adaptive GRED (see Figure 5) since BLUE-Linear drops larger number of packets than either GRED or Adaptive GRED (see Figure 6).

Table 1: mql, T and D performance results of GRED, Adaptive GRED and BLUE-Linear.

α	GRED			Adaptive GRED			BLUE-Linear Analytical Model		
	mql	T	D	mql	T	D	mql	T	D
0.18	0.457	0.1787	2.5604	0.457	0.17866	2.5601	0.461	0.18	2.562
0.33	1.279	0.3277	3.903	1.279	0.3277	3.9038	1.36	0.32999	3.9397
0.48	6.1005	0.4689	13.009	6.088	0.4688	12.984	5.90455	0.4709	12.538
0.63	13.578	0.497	27.299	13.3645	0.4975	26.861	13.1948	0.49994	26.3926
0.78	14.7936	0.49885	29.6551	14.3547	0.49909	28.761	14.843	0.4999999	29.686
0.93	14.9456	0.499	29.9316	14.4275	0.4995	28.8811	15.69798	0.5	31.39597

Figure 5: P_L vs. α .

Figure 6: D_P vs. α .

6
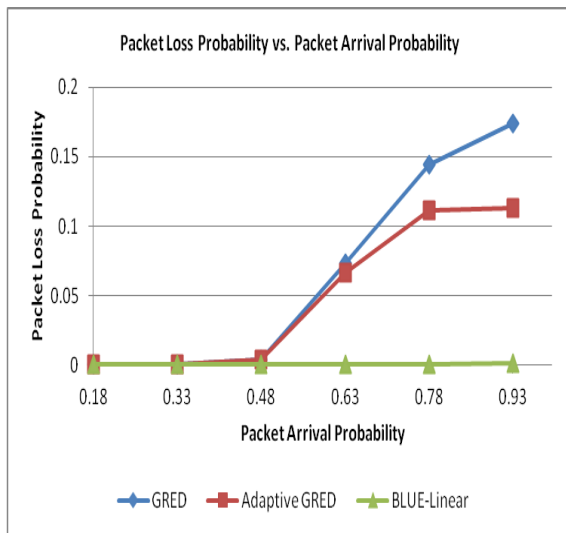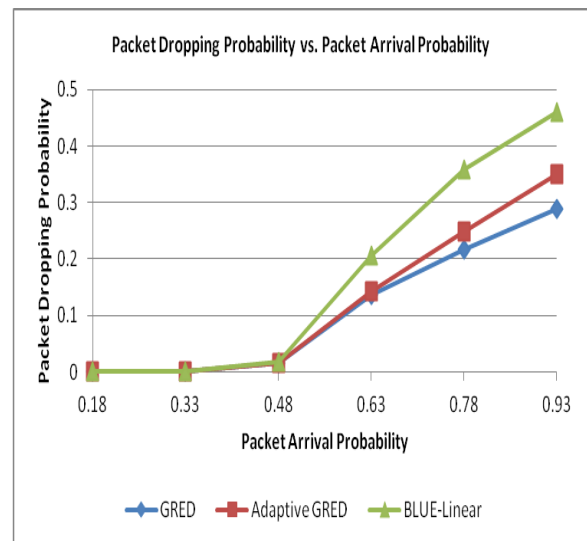
7. Conclusions and Future Work

An Adaptive GRED algorithm has been presented to improve the performance of GRED when congestion situation occurs in the networks router buffer. Adaptive GRED algorithm slightly provides better mql, P_L and D results than that of GRED. When $\alpha > \beta$, GRED and Adaptive GRED offered better performance results with reference to mql and D than those of BLUE-Linear. BLUE-Linear presented marginally better throughput performance than either GRED or Adaptive GRED whether congestion is occurred or not. Also, the Adaptive GRED improves the process of parameter settings of $\max threshold$ and $D_{\max}$. This is accomplished by varying the D_{init} value from the value of $D_{\max}$ from to 0.5 as long as the value of aql (its congestion measure) from $\max threshold$ value to $2 \times \max threshold$ value. A comparison between GRED and the Adaptive GRED was conducted, the results of the comparison are the following:

- GRED and Adaptive GRED offered similar mql, T, D, P_L and D_p results in no or light congestion situations.
- The Adaptive GRED marginally outperformed GRED with reference to mql and D results in congestion situations.
- The compared methods provide similar T results in congestion situations.
- The Adaptive GRED presented better P_L results than that of GRED, whereas GRED provided better D_p results than that of Adaptive GRED.

In near future we intend to use other arrival processes such as Markov-modulated Bernoulli process (MMBP) in the proposed Adaptive GRED method aiming to determine the performance in bursty traffic and long range dependence.

References

1. Ababneh J., Abdeljaber H., Thabtah F., Hadi W., and Badarneh E. "Derivation of Three Queue Nodes Discrete-Time Analytical Model Based on DRED Algorithm". The Seventh IEEE International Conference On Information Technology: New Generations (ITNG 2010). IEEE Computer Society, pp. 885-890, April 2010, Las Vegas, USA.

2. Abdeljaber H., Thabtah F., Woodward M. "Traffic Management for the Gentle Random Early Detection Using Discrete-Time Queueing," Proceedings of the International Business Information Management Conference (9th IBIMA), ISBN: 0-9753393-8-9, pp. 289-298, Marrakech, Morocco, January 2008.

3. Abdeljaber H., Woodward M., Thabtah F. "Performance Evaluation For DRED Discrete-Time Queueing Network Analytical Model," Proceedings of the Journal of Network And Computer Applications, Volume 31, Issue 4, pp. 750-770, Elsevier, November 2008.

4. Abdeljaber H., Woodward M., Thabtah F., Al-diabat M., "Modelling BLUE Active Queue Management Using Discrete-Time Queue," Proceedings of the 2007 International Conference of Information Security And Internet Engineering (ICISIE'07), pp. 568-573, London, U.K., July 2007.

5. Athuraliya S., Li V.H., Low S.H., and Yin, Q. "REM: Active Queue Management," IEEE Network, 15(3), pp. 48-53, May 2001.

6. Aweya J., Ouellette M., and Montuno D.Y. "A Control Theoretic Approach To Active Queue Management," Computer Networks, volume 36, issue 2-3, pp. 203-35, July 2001.

7. Braden R., Clark D., Crowcroft J., Davie B., Deering S., Estrin D., Floyd S., Jacobson V., Minshall G., Partridge C., Peterson L., Ramakrishnan K., Shenker S., wroclawski J., and Zhang L. "Recommendations On Queue Management And Congestion Avoidance in the Internet," RFC 2309, April 1998.

8. Feng W., Shin K.G., and kandlur D. "The Blue Active Queue Management Algorithms," IEEE/ACM Transactions on Networking, volume 10, issue 4, pp. 513-528, August 2002.

9. Floyd S., and Jacobson V. "Random Early Detection Gateways For Congestion Avoidance," IEEE/ACM Transactions on Networking, volume 1, issue 4, 1(4): pp. 397-413, August 1993.

10. Floyd S., Ramakrishnan G., and Shenker S. "Adaptive RED: An Algorithm For Increasing the Robustness of RED's Active Queue Management," Technical report, ICSI, August 1, 2001.

11. Floyd S. "Recommendations On Using the Gentle Variant of RED," May 2000, Available at http://www.aciri.org/floyd/red/gentle.html.

12. Tanenbaum A.S., Computer Networks, Fourth Edition, Prentice Hall PTR, 2002.

13. Thiruchelvi G., and Raja, J. "A Survey On Active Queue Management Mechanisms," International Journal of Computer Science and Network Security, volume 8, no. 12, December 2008.

14. Welzl M. "Network Congestion Control: Managing Internet Traffic," 282 pages, July 2005.

15. Woodward M. E. "Communication and Computer Networks: Modelling With Discrete-Time queues," Pentech Press, London, 1993.

7

Predicting Phishing Websites using Neural Network trained with Back-Propagation

Rami M. Mohammad[1]**, Fadi Thabtah**[2]**, and Lee McCluskey**[3]

[1]Computing and Engineering, University of Huddersfield, Huddersfield, UK, rami.mohammad@hud.ac.uk

[2]E-Business Department, CUD, Dubai, UAE, fadi@cud.ac.ae

[3]Computing and Engineering, University of Huddersfield, Huddersfield, UK, t.l.mccluskey@hud.ac.uk

Abstract — *Phishing is increasing dramatically with the development of modern technologies and the global worldwide computer networks. This results in the loss of customer's confidence in e-commerce and online banking, financial damages, and identity theft. Phishing is fraudulent effort aims to acquire sensitive information from users such as credit card credentials, and social security number. In this article, we propose a model for predicting phishing attacks based on Artificial Neural Network (ANN). A Feed Forward Neural Network trained by Back Propagation algorithm is developed to classify websites as phishing or legitimate. The suggested model shows high acceptance ability for noisy data, fault tolerance and high prediction accuracy with respect to false positive and false negative rates.*

Keywords: *Web Threat, Phishing, Information Security, Neural Network, Data Mining.*

1. Introduction

Internet facilitates reaching customers all over the globe without any market place restrictions and with effective use of e-commerce. As a result, the number of customers who rely on the Internet to perform procurements is increasing dramatically. Hundreds of millions of dollars are transferred through the internet every day. This number was tempting the fraudsters to carry out their fraudulent operations. Thus, internet-users were vulnerable to different types of web-threats. Hence, the suitability of the internet for commercial transactions becomes doubtful. Phishing is a form of web-threats that is defined as the art of mimicking a website of an authentic enterprise aiming to acquire private information [1]. Presumably, these websites have high visual similarities to the legitimate ones in an attempt to defraud the honest people. Social engineering and technical tricks are commonly combined together in order to start a phishing attack. Typically, a phishing attack starts by sending an e-mail that seems authentic to potential victims urging them to update or validate their information by following a URL link within the e-mail. Predicting and stopping phishing attack is a critical step toward protecting online transactions. Several approaches were proposed to mitigate these attacks. Anti-phishing measures may take several forms including legal, education

and technical solutions. Technical solution is the subject of our interest, particularly, heuristic-based approach. The most popular techniques in designing technical anti-phishing solutions include:

- *Blacklist Approach*: In which the requested URL is compared with those in that list. The downside of this approach is that the blacklist usually cannot cover all phishing websites since a newly created fraudulent website takes considerable time before it is being added to the list. The gap between launching and adding the suspicious website to the list may be enough for the phishers to achieve their goals.

- *Heuristic Approach*: Where several features related to website are collected to classify it as either phishy or legitimate. In contrast to the blacklist method, a heuristic-based solution can recognize freshly created fake websites in real-time.

The accuracy of the heuristic-based solution depends mainly on a set of discriminative criteria's picked out from the website. Hence, the way in which those features are processed plays an extensive role in classifying websites correctly. Therefore, an effective and fast retrieval method of information is essential for taking a good decision. Data mining is one of the techniques that can make use of the features extracted from the websites to find patterns as well as relationships among them [2]. Although plenty of applications offered for combating phishing websites, few of them make use of data mining techniques in distinguishing phishing websites from legitimate ones. Moreover, most of these suggested solutions are inapplicable, inaccurate and produce an unacceptable level of false positives rates, which means classifying legitimate website as phishy. Phishing detection is a type of classification tasks in data mining, which have been applied successfully in different domains, i.e. classification, clustering, etc. each instance in the testing dataset is assigned to one of predefined classes. Phishing is considered a binary classification problem because the target class has two possible values "Phishy" or "Legitimate". Neural Network (NN), which is the subject of our interest, is a computerized model of the human brains and nervous system. NN composed of interconnected processing units called (neurons) [3]. The links that connect the neurons to each other hold values that signify the relative importance of each input to a neuron and it

is called connections weights [3]. Connection weights are the crucial elements in any NN model. Connection weights are adjusted repeatedly during the training phase until reaching an acceptable solution. A trained neural network is considered as an expert in the field of information to which it is applied.

In this article, we try to answer the following research questions:

1- How NN can be trained to achieve an acceptable predictive performance.

2- What is the best NN architecture in predicting phishing websites?

This article structured as follows: Section II discusses related works and highlights different phishing detection methods presented in the literature. Section III describes the features used in our model. Section IV introduces traditional NN modelling techniques. In Sections V, VI, several experiments conducted. We conclude in Section VI.

2. RELATED WORK

Although a wide-range of anti-phishing solutions are offered, most of them are not talented to make a decision perfectly thus the false positive decisions rose intensely. In this section, we review current anti-phishing approaches as well as techniques utilized in developing solutions for phishing problem. One approach employed in [4]. is based on fuzzy data mining. The model works on multilayered approach i.e. each layer should have its own rules; however, it was not clear if the rules were established based on human experience, or extracted using an automated tool. Moreover, the authors classify the website as very-legitimate, legitimate, suspicious, phishy or very-phishy, but they did not clarify what is the fine line that separate one class from one another. Another method proposed in [5] suggested a new way to detect phishing websites by capturing abnormal behaviours demonstrated by these websites. Structured website consists of "W3C DOM" features. The authors have selected six structural features: (Abnormal URL, abnormal DNS record, abnormal anchors, Server form handler, abnormal cookies and abnormal certificate in SSL). Support-Vector-Machine classifier (Vapnik) is used to determine whether the website is phishy or not. The classification accuracy in this method was 84%, which is relatively considered low. However, this method snubs important features that can play a key role in determining the legitimacy of the website, which explains the low detection rate. One solution to improve this method could be by using security related features.

The method proposed in [6], suggested utilising "CANTINA" which is content-based technique to detect phishing websites using the term-frequency-inverse-document-frequency (TF-IDF) information retrieval measures [7]. TF-IDF often produces weights that assess the word importance to a document by counting its frequency. CANTINA works as follow:

1. Calculate the TF-IDF for a given webpage.

2. Take the five highest TF-IDF terms and add them to the URL to find the lexical signature.

3. The lexical signature is fed into a search engine.

If the N tops searching results having the current webpage, it is considered a legitimate webpage. If not, it is a phishy webpage. N was set to 30 in the experiments. If the search engine returns zero result, the website is labelled as phishy. However, a limitation of this method is that some legitimate websites consist of images and so extracting the TF-IDF terms may not be accurate in this case. Moreover, this method is delayed in querying through a search engine and thus the user may have started in the disclosure of his personal information. Lastly, this approach does not deal with hidden texts, which might be effective in detecting the type of the webpage. In 2010, a survey presented in [8] evaluated the performance of machine-learning-based-detection-methods including: "AdaBoost, Bagging, SVM, Classification and Regression Trees (CART), Logistic Regression (LR), Random Forests (RF), NN, Naive Bayes and Bayesian Additive Regression Trees (BART)". Results showed that 7 out of 9 of machine-learning-based-detection-methods outperformed CANTINA [9] in predicting phishing websites, those are: AdaBoost, Bagging, (LR), (RF), (NN), Naive Bayes and (BART)". Another study in [10] compared the predictive accuracy of several machine-learning strategies (LR), (CART), (BART), (SVM), (RF), and (NN) for predicting phishing emails. A dataset consist of 1171 phishing emails and 1718 legitimate emails are used within the comparative study. A set of 43 features were used to train and test the classifiers. The experiments showed that (RF) has the lowest error rate of 7.72%, followed by CART 08.13%.

3. PHISHING WEBSITES FEATURES

There are several features that distinguish phishing websites from legitimate ones. In our study, we used 18 features descried briefly hereunder:

1. IP address: Using IP address in the hostname part of the URL address means user can almost be sure someone is trying to steal his personal information.

2. Long URL: Phishers resort to hide the suspicious part of the URL, which may redirect the information submitted by the users or redirect the uploaded page to a suspicious domain.

3. URLs having "@" symbol: The "@" symbol leads the browser to ignore everything prior it and redirects the user to the link typed after it.

4. Prefix and Suffix in URLs: Phishers deceive users by reshaping the URL to look like legitimate ones. A technique used to do so is by adding prefix or suffix to the legitimate URL so users might not notice any difference.

5. Sub-domain(s) in URL: Another technique used by the phishers to deceive the users is by adding sub-domain(s) to the URL thus the users may believe that they are dealing with a credited website.

6. Misuse of HTTPs protocol: The existence of the HTTPs protocol every time sensitive information is being transferred reveals that the user certainly connected with an honest website. However, phishers may use a fake HTTPs protocol so that users might be deceived. In [11] a recommendation to

check whether the HTTPs protocol is offered by a trusted issuer such as "GeoTrust, GoDaddy".

7. Request URL: A webpage usually consists of a text and some objects such as images and videos. Typically, these objects are loaded to the webpage from the same domain where the webpage exists. If the objects are loaded from a domain different from the domain typed in the URL address then the webpage is potentially suspicious.

8. URL of Anchor: Similar to "Request URL" but for this feature the links within the webpage might refer to a domain different from the domain typed on the URL address bar. This feature is treated exactly as "Request URL".

9. Server Form Handler "SFH": Once the user submits his information, that information will be transferred to a server to be processed. Normally, the information is processed from the same domain where the webpage is being loaded. Phishers resort to make the server form handler either empty or the submitted information are transferred to different domains.

10. Abnormal URL: If the website identity does not match its record shown in the WHOIS database (http://who.is/) the website is classified as "Phishy". This feature is a binary feature.

11. Redirect Page: This feature is commonly used by phishers by hiding the real link which asking users to submit their information to a suspicious website.

12. Using Pop-up Window: It is unusual to find a legitimate website that asks users to submit their credentials through a popup window.

13. Hiding the Suspicious Links: Phishers resort to hide the suspicious link by showing a fake link on the status bar of the browser or by hiding the status bar itself.

14. DNS Record: If the DNS record is empty or not found the website is classified as "Phishy", otherwise it is classified as "Legitimate".

15. Website Traffic: Legitimate websites are of high traffic since they are visited regularly. Phishing websites often a short life thus their web traffic is either not exists ranked is below the limit that gives it the legitimate status.

16. Age of Domain: the website is considered "Legitimate" if the domain aged more than 2 years [11].

17. Disabling Right Click: Phishers use JavaScript to disable the right click function, so that users cannot view and save the source code.

18. Port number: We examine if there is a port number in the URL and check if the port belongs to the list of well-known HTTP ports such as 80, 8080, 21, 443, 70, and 1080. If the port number does not belong to the list, we flag it as a possibly phishing URL.

4. MODELLING NEURAL NETWORK

An Artificial Neural Network (ANN) is an information-processing model that is stimulated by how biological nervous systems process information. The key element of this model is the unique structure of the information-processing scheme. NN consist of a large number of highly interconnected processing elements "neurones", working in harmony to solve problems. ANNs, like human, learn by example. NNs, with their amazing ability to derive meaningful data from complex dataset, can be used to mine patterns that are too difficult to be noticed by humans [2]. A trained NN can be thought of as an "expert" in the domain it has been applied and can be used to predict class of new cases. Other advantages include [3]:

- Nonlinearity: NN is very effective technique in modelling classification problems where the output values are not directly related to its input.
- Adaptive: Neural network has the ability to adjust the weights based on the changes of its surrounding environments.
- Generalisation: NN is able to find the suitable output for the inputs that does not exist in the training data.
- Fault-tolerance: NN performance is not significantly affected under difficult circumstances such as losing connection between some neurons, noisy or missing data.
- Identical designing steps: The same principles, scheme and methodological steps are employed in designing ANN in all domains.

In our study, we used MATLAB to train our model. The NN performance is assessed by means of Mean Square Error (MSE). We show how NN can be structure to classify websites. MATLAB is a numerical computing environment and a programming language as well. The NN Toolbox is used to design, implement, visualize and simulate our NNs. MATLAB provides wide-ranging support for several NN paradigms, and graphical user interfaces (GUIs) supported by MATLAB enables the user to design NN in a very simple way. We developed Multi Layer Perceptron (MLP) model and calculated the resulting NN model performance by means of MSE. Fig. 1 shows the steps required to create an NN model. The MLP program starts by reading the training, validation and testing datasets, each dataset is stored in an Excel file. To read the datasets we used "xlsread" built-in function. Then, after loading the datasets, the training examples are randomized using the function "randperm.

Next, the input variables website features (Using_IP address, Long URL, URL having @ symbol … etc.) and output variable (website_class) are stated for both training and validation datasets.
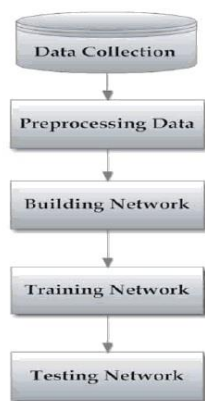
Figure 1 Steps to create an ANN models

MATLAB facilitate creating the MLP model using the "newff" built-in function, which creates a feed-forward back-propagation network. By using this function, we were able to specify the number of hidden-layers, the number of neurons in each layer, the transfer function, the training function, the weight/bias learning function and the performance function.

Once NN training is fully structured, the network performance has to be tested. Therefore, unseen dataset will be presented to the model to show its performance.
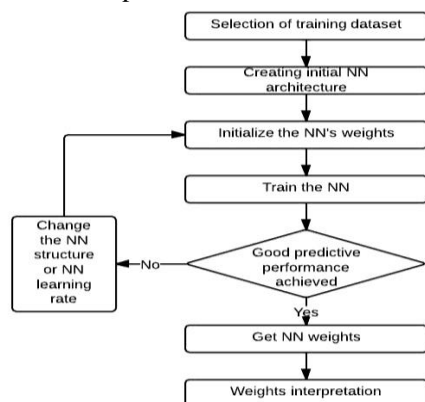
Figure 2 Phishing detection model

The phishing detection model is shown in Fig 2. The model starts by loading the training dataset, then we create the initial NN structure by means of number of layers, number of neurons in each layer and the learning parameters i.e. learning rate, momentum value and number of epochs. Once the NN structure is determined, the weights are initialized to small non zero values. The model is then trained until the maximum number of epochs or the desired error rate is achieved. The model is then tested on the testing dataset which is never being seen once before. If the predictive performance is acceptable then the NN is generated and the weights are produced. Otherwise, the NN structure is improved by changing the number of neurons in the hidden layer or by updating the network parameters i.e. learning rate and momentum value. In our model we adopted the pruning approach to specify the number of neurons in the hidden layer, since we started with a large number of neurons, and the progressively one or more

neurons removed during training until the desired performance is met.

5. EXPERIMENTAL METHODOLOGY

A dataset that consists of 1828 websites were used to extract the 18 features using our own tool. The dataset is composed of 859-legitimate website collected from yahoo directory (http://dir.yahoo.com/) and starting point directory (http://www.stpt.com/directory/), and 969-phishing website collected from Phishtank (http://www.phishtank.com/) and Millersmiles archives (http://www.millersmiles.co.uk/). The collected dataset holds categorical values i.e. "Legitimate", "Suspicious" and "Phishy". These values should be transformed to numerical values so that the neural network can perform its calculations thus we replaced the values 1,0 and -1 instead of "Legitimate", "Suspicious" and "Phishy" respectively. We are interested in obtaining a model with a good generalisation performance. However, most models are susceptible to overfitting, which means, while the error rate on the training dataset decreases during the training phase, the error rate on the unseen dataset (testing dataset) increases at some point. To overcome this problem, we used the "Hold-Out" validation technique, by dividing our dataset into training, validation and testing datasets. The examples in each dataset were selected randomly. We split our dataset to 15% for validation, 15% for testing and 70% for training. Training dataset is used to train the network and to adjust the weights of the network, while the testing dataset remains unseen and it is used to assess the predictive performance of the model. After training, we ran the network on the testing dataset. The error value on the testing dataset offers an unbiased approximation of the generalization error. There are several methods for supervised training of NNs. The backpropagation algorithm [3]is the most frequently used training method for ANNs. Backpropagation is usually implemented along with feed-forward NNs that have no feedback. The main idea in feed-forward NNs is to propagate the error through the hidden layers to update the weights of NN. The back-propagation

```
Initialize the weights vector
S = the training set fed to the network
Repeat
        For each "input-output" pair denoted by P in S
            In = input pattern in P
            Out = desired output
            Compute network output (netout)
            network error = Out – netout
        end For
    Find weight change for weights connecting hidden to output
    Find weight change for weights connecting input to hidden
    Update weights
Until reaching (a satisfactory network error value OR maximum iteration)
```

algorithm is described as the following pseudo code:

6. TRAINING TECHNIQUES

Determining the network architecture is one of the difficult tasks in constructing a NN model but one of the most essential steps. The NN architecture employed in this study is feed-forward with one hidden layer, which sometimes called multi-

layered perceptron. Problems that need more than one hidden layer are infrequently encountered. Determining the number of hidden layers is only a small problem. We must also decide the number of neurons in each hidden layer. Too few neurons in the hidden layers will cause under-fitting, and too many neurons can result in overfitting. Therefore, the number of hidden layers and the number of neurons in each hidden layers must be carefully determined. Sigmoid transferring function is used in our network. Table 1 summarises the predictive performance achieved in our experiments. The results showed that the best predictive performance was achieved when the number of neurons in the hidden layer was set to "2" and the learning rate was set to 0.7. Moreover, using more than one hidden layer does not improve the predictive performance on the model, thus a single hidden layer is enough to achieve a good predictive performance.

Table 1 Experimental results

Exp #	Number of layers	Number of neurons	Learning rate	Momentum	MSE
1	1	8	0.2	0.7	0.005200
2	1	8	0.6	0.7	0.002950
3	1	5	0.2	0.7	0.005490
4	1	4	0.2	0.7	0.005698
5	1	4	0.6	0.7	0.003956
6	1	4	0.4	0.5	0.005160
7	1	3	0.2	0.7	0.005902
8	1	3	0.4	0.4	0.005695
9	1	3	0.6	0.7	0.004658
10	1	2	0.2	0.7	0.005863
11	1	2	0.7	0.7	0.002234
12	2	3, 2	0.2	0.7	0.005599

7. CONCLUSION

The main goal of this paper was to develop an ANN model to classify websites as either "Phishy" or "Legitimate". Several NNs structures were studied to determine the NN parameters i.e. "number of hidden layers, number of hidden neurons, learning rate and momentum value"; that provide the best predictive accuracy. The selection of a suitable number of hidden neurons during constructing the NN showed to be crucial. One hidden layer was enough for the training and it achieved good performance. We created different networks aiming to lower the error. We assumed that the hidden neurons are 8, 5, 4, 3 and 2. The experimental results showed that the best performance is obtained when the number of hidden neurons was set to 2. Furthermore, the results indicated that the success of NN is impacted when the network parameters are changed i.e. "learning rate and momentum value".

Overall, we were able to show that NN is a good technique in predicting phishing websites. In near future we think of automating the process of building a NN in order to reduce the training time.

8. REFERENCES

[1] APWG, G. Aaron and R. Manning, "APWG Phishing Reports," APWG, 1 February 2013. [Online]. Available: http://www.antiphishing.org/resources/apwg-reports/. [Accessed 8 February 2013].

[2] I. H. Witten and E. Frank, "Data mining: practical machine learning tools and techniques with Java implementations," ACM, New York, NY, USA, March 2002.

[3] B. Widrow, M. and A. Lehr, "30 years of adaptive neural networks," *IEEE press,* vol. 78, no. 6, pp. 1415-1442, 1990.

[4] M. Aburrous, M. A. Hossain, K. Dahal and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert Systems with Applications: An International Journal,* pp. 7913-7921, December 2010.

[5] Y. Pan and X. Ding, "Anomaly Based Web Phishing Page Detection," in *In ACSAC '06: Proceedings of the 22nd Annual Computer Security Applications Conference.*, Washington, DC, Dec. 2006.

[6] Y. Zhang, J. Hong and L. Cranor, "CANTINA: A Content-Based Approach to Detect Phishing Web Sites," in *Proceedings of the 16th World Wide Web Conference*, Banff, Alberta, Canada, 2007.

[7] C. . D. Manning, P. Raghavan and H. Schütze , Introduction to Information Retrieval, Cambridge University Press, 2008.

[8] D. Miyamoto, H. Hazeyama and Y. Kadobayashi, "An Evaluation of Machine Learning-based Methods for Detection of Phishing Sites," *Australian Journal of Intelligent Information Processing Systems,* pp. 54-63, 2 10 2008.

[9] X. Guang, o. Jason, R. Carolyn P and C. Lorrie, "CANTINA+: A Feature-rich Machine Learning Framework for Detecting Phishing Web Sites," *ACM Transactions on Information and System Security,* pp. 1-28, 09 2011.

[10] S. Abu-Nimeh, D. Nappa, X. Wang and S. Nair, "A Comparison of Machine Learning Techniques for Phishing Detection," in *Proceeding eCrime '07 Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit* , New York, NY, USA , 2007.

[11] R. M. Mohammad, F. Thabtah and L. McCluskey, "An Assessment of Features Related to Phishing Websites using an Automated Technique," in *The 7th International Conference for Internet Technology and Secured Transactions (ICITST-2012)*, London, 2012.

Associative Classification Mining for Website Phishing Classification

[1]Neda Abdelhamid, [1]Aladdin Ayesh, [2]Fadi Thabtah
[1]Informatics Dept, De Montfort University, Leicester, LE1 9BH
Email: [1]p09050665@myemail.dmu.ac.uk
aayesh@dmu.ac.uk
[2]E-Business Dept, CUD, Dubai
[2]Email: fadi@cud.ac.ae

Abstract --_Website phishing is one of the crucial research topics for the internet community due to the massive number of online daily transactions. The process of predicting the phishing activity for a website is a typical classification problem in data mining where different website's features such as URL length, prefix and suffix, IP address, etc., are used to discover concealed correlations (knowledge) among these features that are useful for decision makers. In this article, an Associative classification (AC) data mining algorithm that uses association rule methods to build classification systems (classifiers) is developed and applied on the important problem of phishing classification. The proposed algorithm employs a classifier building method that discovers vital rules that possibly can be utilised to detect phishing activity based on a number of significant website's features. Experimental results using the proposed algorithms and three other rule based algorithms on real legitimate and fake websites collected from different sources have been conducted. The results reveal that our algorithm is highly competitive in classifying websites if contrasted with the other rule based classification algorithms with respective to accuracy rate. Further, our algorithm normally extracts smaller classifiers than other AC algorithm because of its novel rule evaluation method which reduces overfitting._

Keywords_: Associative Classification, Data Mining, Phishing Detection, WEB Security_

1. INTRODUCTION

Associative classification in data mining is about constructing classification systems (classifier) from an input data called the training data set aiming to predict the class value of unseen data called test data set accurately [1]. One distinguishing feature of AC algorithms is their ability to discover new hidden knowledge and then extract them as simple If-Then rules. In the last decade, different research studies on AC mining have resulted in the disseminations of various algorithms including CBA [2], CMAR [3], LCA [4], ADA [5]and others. These studies have revealed that AC is able to construct more accurate classifiers than rule based classification data mining approaches including rule induction and decision tree. Nevertheless, the numbers of rules discovered by the AC algorithms are normally huge which therefore limits its applicability sometimes in business domains. One primary reason of the large numbers of rules resulting from these AC algorithms is inherited from association rule since all correlations among the

attribute values and the class attribute are tested in the training phase and many rules are derived. One way to control the exponential growth in the number of rules is to develop rule filtering methods that minimise rules redundancy during building the classifier.

Rule evaluation sometimes called filtering or pruning usually occurs during building the classifier in AC mining. So once the complete set of rules are found in the training phase and sorted based on certain conditions (e.g. rule's confidence, support, body length, etc), the AC algorithm has to decide the way it should choose a subset of effective rules to represent the classifier. There are different ways used in AC to choose the classifier's rules. For instance, CBA [6] utilises the database coverage rule where rules that cover correctly a certain number of training cases are marked as accurate rules and the remaining rules get discarded. Lazy AC algorithms like L3G algorithms employ lazy pruning that stores primary and secondary rules in the classifier.

In this paper, we first treat the problem of generating large classifiers in AC by proposing a new rule evaluation method for removing useless and redundant rules during constructing the classifier. The new rule evaluation method is an enhancement of a current AC called Multiclass Associative Classification (MA) [7]. We have enhanced MAC rule pruning method and classification procedure in which rather than using one rule for prediction in the proposed algorithm we utilise group of rules prediction to enhance the accuracy rate. Further, in building the classifier we developed a rule evaluation method that increases the training coverage per rule in order to reduce the classifier size and thus end-user can control and understand the classifier easily. The proposed rule evaluation method ensures larger training data coverage per classifier rule by taking into account only the similarity of rule's body and the training case attribute values while building the classifier. Whereas other current AC algorithms like MCAR consider the class similarity between the candidate rule and the training data, and the attribute values in the candidate rule body and those belonging to the training data. The two enhancements have resulted in a new algorithm that we call Enhanced Multiclass Associative Classification (eMAC). So eMAC's rule evaluation method ensures less number of rules in the classifier.

We show the applicability of eMAC on a crucial domain related to web security named website phishing classification that normally criticised of having dense data because of the correlations among the website's features. Phishing is considered a form of web-threats that is defined

as the art of impersonating a website of an honest enterprise aiming to acquire private information such as usernames, password's and social security numbers [8]. Phishing websites are created by dishonest people to impersonate a webpage of genuine websites. Almost these websites have high visual similarities to the legitimate ones in an attempt to defraud the innocent people. Some of these websites designed to be almost similar to the genuine ones. Social engineering and technical tricks are commonly combined together in order start a phishing attack [8]. Phishing websites has become a serious problem not only because of the increased number of those websites but also due to the smart strategies used to design such websites, and thus even those having a good experience in the computer and internet might be deceived.

The process of detecting the type of website is a typical classification problem where different features like URL length, sub-domains, and adding prefix and suffix, etc, are utilised to learn important hidden knowledge among these features. This knowledge is in fact the classification system that in turn is used to automatically guess the phishing activities of the website when a user browses it. The phishing problem is considered a vital issue in ".COM" industry especially e-banking and e-commerce taking the number of online transactions involving payments.

This article deals with two problems

    1) Improvement of current AC algorithms particularly the generation of a large number of rules by proposing a new method that reduces the number of rules discovered without drastically impacting the predictive accuracy of the classifiers. In other words, and during constructing the classifier, we would like to minimise the number of rules derived by an AC algorithm. This can help decision makers especially in understanding, controlling and maintaining the final set of rules primarily when making a prediction decision.

    2) The applicability of AC mining on the website phishing problem to learn important hidden knowledge from the website's features correlations. These correlations will be extracted as "If-Then' rules in order to be used by end-user for the automatic classification of websites.

A number of fake and legitimate websites collected from known sources like Phishtank (www.phishtank.com) and millersmiles (www.millersmiles.co.uk) in the experimintation section to evaluate the performance of the proposed algorithm. Further, eMAC and three other AC and rule based algorithms have been contrasted with respect to different performance measures like classificaiton accuracy and numberof rules. More details are given in Section4.,

This article is structured as follows: Section 2 presents the phishing problem and related definitions to AC in data mining. The proposed algorithm and its main steps are explained in Section 3. Section 4 is devoted to experimentations and finally conclusions are given in Section 5.

2. THE PHISHING PROBLEM AND ASSOCIATIVE CLASSIFICATION MINING

Typically, a phishing attack starts by sending an e-mail that appears to be from an authentic organisation to victims urging them to update or validate their information by following a fake URL link within the e-mail body. E-mail remains the main spreading channel for phishing links since 65% of phishing attacks start by visiting a link received within an e-mail (Kaspersky Lab, 2013). Typically, two common approaches are used to detect phishing activities, i.e. blacklist and features methods [6]. In the black list approach the website URL is basically compared with those in the black list to identify whether it is legitimate or fake. On the other hand a more realistic approach which is based on extracting the website features and using a heuristic method to identify the phishing activities have been successfully utilised [9]. Unlike the blacklist approach, the features based approach distinguishes new created phishing in real-time [8]. The effectiveness of the features methods depends on selecting a set of significant features that could help in determining the phishy website [9].

Phishing detection for websites is a typical classification in data mining problem where the goal is to forecast the type of the website based on a number of features that can be stored in the training data set. For simplicity we can consider the website phishing detection a two class problem (binary classification) since the target class has only two possible values; "Phishy" or "Legitimate". Once a webpage is loaded on the browser a set of features will be extracted from the webpage. Those features have an influence in determining the type of the webpage. Website features like IP address, long URL, https and SSL are examples of important features that are used for learning knowledge. An AC data mining model will learn from the websites features important knowledge (correlations between the features values and the class attribute) to classify the webpage as either "Phishy" or "Legitimate".

We start formulating the phishing detection problem in AC data mining with definitions given in [4]. Let T denote the domain of the training data containing phishing features and C be a list of classes. Each training data $t \, \varepsilon \, T$ may be given a single class ck where ck ε C, and is represented as a pair (t, ck) where ck is connected with the data instance t in the training data. Let H denote the set of classifiers for $T \rightarrow C$ where each case $t \, \varepsilon \, T$ is given a classes and the goal is to find a classifier $h \, \varepsilon \, H$ that maximises the probability that h(t) = c for each test data. So, for the training data set T with m attributes A1, A2, … , Am and C is a set of classes,

Definition 1: An attribute value set (*AttValSet*) can be described as a set of disjoint attribute values contained in a training case, denoted $< (A_{i1}, a_{i1}), …, (A_{ik}, a_{ik})>$.

Definition 2: A rule *r* is of the form $< AttValSet, c>$, where $c \varepsilon C$ is the class.

Definition 3: The actual occurrence (*ActOccr*) of *r* in *T* is the number of cases in *T* that match *r's* antecedent.

Definition 4: The support count (*SuppCount*) of *r* is the number of cases in *T* that matches *r's* antecedent, and belong to a class c_i.

Definition 5: A rule *r* passes the user minimum support threshold (*minsupp*) if for *r*, the $SuppCount(r)/ \, |T| \geq minsupp$, where |T| is the number of cases in *T*.

Definition 6: A rule *r* passes the user minimum confidence threshold (*minconf*) if $SuppCount(r)/ActOccr(r) \geq minconf$.

Generally, an AC algorithm operates in three main phases. Firstly, it discovers all frequent attribute values which hold enough supports. Once all frequent attribute values are found, then it transforms the subset of which hold enough confidence values into rules. In other words, the algorithm finds and extracts rules that pass user defined thresholds denoted by minimum support (minsupp), and minimum confidence (minconf). In the second phase, rule pruning operates where only rules with high quality (confidence and support values) are selected to represent the classifier. Lastly, the classifier is utilised to forecast the class values on new unseen data.

3. THE PROPOSED ALGORITHM

The proposed algorithm utilises AC learning strategies to generate the rules. It comprises of three main steps: rules discovery, classifier building and class assignment procedure (prediction step). In the first step, it iterates over the input training data set in which the rules is found and extracted using minsupp and minconf thresholds. Then in the second step it tests the discovered rules on the training data set in order to select one subset to represent the classifier. The final step involves assigning classes to test data. The general description of the eMAC learning algorithm is depicted in Figure 1, and details are given in the next subsections.

We assume that the input attributes are categorical or continuous attributes. For continuous attributes any discretisation measure is employed before the training phase. Missing values attributes will be treated as other existing values in the data set.

3.1. RULE DISCOVERY

EMAC uses a training method that employs a simple intersection among ruleitems locations in the training data set (TIDs) to discover the rules. The TID of a ruleitem holds the row numbers that contain the attribute values and their corresponding class labels in the training data set. The proposed algorithm discovers the frequent ruleitem of size 1 (F1) after iterating over the training data set. Then, it intersects the TIDs of the disjoint ruleitems in F1 to discover the candidate ruleitems of size 2, and after determining F2 the possible remaining frequent ruleitems of size 3 are obtained from intersecting the TIDs of the disjoint ruleitems of F2, and so forth. The TIDs of a ruleitem comprises useful information that are utilised to

locate values easily in the training data set especially in computing the support and confidence for rules.

When frequent attribute values are identified, eMAC generates any of which as a rule when it passes the minconf threshold. Now, when an attribute value is connected with more than one class and became frequent, EMAC considers only the largest frequency class associated with the attribute value and ignores the other. In cases that the classes frequencies in the training data set when connected with the attribute value is similar the choice is random.

3.2. RULE RANKING METHOD

There are several different rule ranking formulas containing different criteria considered by scholars in AC. For instance, CBA algorithm [2] and its successors consider the rule's confidence and support as main criteria for rule favouring, CMAR [3] and MCAR [4] algorithms add on top of that the rule's length and the majority class count respectively when rules having identical confidence and support. On the other hand, lazy AC algorithms [10] place specific rule first (rules with large number of attribute values in their body) since they claim these rules are often more accurate. Though, this approach has been criticised of ending up with very large classifiers that are hard to be maintained, understood and updated.

We argue that the minority class frequency as a rule preference parameter should be employed rather than the majority class count as in MCAR when rules are having similar confidence, support and length. This is since the numbers of rules for the lower frequency class are normally smaller than that of the largest frequency class. Therefore, ranking rules with smaller frequency class higher gives them a better chance to survive during rule evaluation and be part of the classifier and resulting with more representation in the context of rules for each class with low frequency in the training data. We have favoured rules associated with less frequent class in rule ranking since such a class is not well represented by rules in the classifier and usually has less number of rules.

3.3. CLASSIFIER CONSTRUCTION

After rules are sorted a subset of which gets chosen to comprise the classifier. The classifier is built by eMAC as follows:

For each training case eMAC iterates over the set of discovered rules and selects the first rule that matches the

---

Input: Training data *D*, *minsupp* and *minconf* thresholds

Output: A classifier that comprises rules

Step One:
- Iterate over the training data set *D* with *n* columns to find all frequent ruleitems
- Convert any frequent ruleitem that passes *minconf* to a single label rule
- Sort the rules set according to Section 3.2.2

Step Two:
- Evaluate the complete set of rules discovered in step (1) on the training data set in order to remove redundant rules or rules that have no training data coverage

Step Three:
- Classify test cases

Fig. 1. The proposed algorithm

training case as a classifier rule. The same process is repeated until all training cases are utilised or all candidate rules have been evaluated. In cases when the training data has any uncovered data the default class rule will be formed. This rules will represents the majority class in remaining uncovered training data. Finally, eMAC outputs all marked rules to form the classifier. The remaining unmarked rules are discarded by the proposed algorithm since some higher ranked rules have covered their training cases during building the classifier and therefore these unmarked rules become redundant and useless.

The rule pruning of the proposed algorithm differs from other pruning procedure in AC such as CBA, CMAR, and CPAR in that it does not require the similarity of the class of both the evaluated rule and the training case as a condition of rule significance rather it only considers the matching between the rule body and the training case. This reduces overfitting the training data set since most of current AC algorithms mark the candidate rule as a classifier rule if its body matches the training case and has the same class as the training case. This may result in more accurate prediction on the training data set but not necessarily on new unseen test cases. We argue that the similarity test between the candidate rule class and the training case class has limited effect on the predictive power of the resulting classifiers during the prediction step. Lastly, one obvious advantage of the proposed rule evaluation method is that it ensures more data coverage per rule which consequently often leads to less number of rules in the classifier. This means end-user can control the classifier and understand it easily.

3.4. CLASSIFICATION OF TEST DATA

When a test case is about to classify, the prediction procedure of the EMAC algorithm works as follow:

It iterates over the set of the rules stored in the classifier, it highlights all rules that are contained in the test data (the rule's body matches some attribute values in the test data). If only one rule is applicable to the test data then the class of that rule is assigned to the test data. In cases where multiple rules are applicable to the test data, the algorithm categorises these rules into groups according to their classes, and counts the number of rules in each group. The class belonging to the group that has the largest number of rules gets assigned to the test data. In case that more than one group having the same number of rules, then the choice will be random. This method which utilises more than one rule to make the class assignment of test data have improved upon single rule prediction procedures such as that of CBA and MCAR that takes the class of the highest ranked rule in the classifier matching the test data to make

the prediction decision. Lastly, in cases when no rules in the classifier are applicable to the test case, the default class (Majority class in the training dataset) will be assigned to that case.

4. EXPERIMENTAL RESULTS

4.1. DATA AND PHISHING FEATURES

We have investigated a large number of different features contributing in the classification of the type of the websites that have been proposed in [8]. We selected nine effective features among them after applying Chi-square feature selection metric in WEKA against 1228 different websites. The dataset utilised in the experiments consists of 547 and 681 legitimate and fake websites respectively. It has been collected from yahoo directory (http://dir.yahoo.com/), starting point directory (http://www.stpt.com/directory), Phishtank (www.phishtank.com) and Millersmiles archives (http://www.millersmiles.co.uk/). Seven samples of the websites features data is shown in Table 1 where the class is either "1" (legitimate) or "0" (phishy). The "-1" value in the below table denotes "Suspicious" which can go either phishy or legitimate so the end-user is unsure about the feature's value. The features that we consider are described below,

1. Using IP address: Using IP address in the hostname part of the URL address means user can almost be sure someone is trying to steal his personal information.

2. Long URL: Phishers resort to hide the suspicious part of the URL, which may redirect the information submitted by the users or redirect the uploaded page to a suspicious domain.

3. Adding Prefix and Suffix to URL: Phishers try to deceive users by reshaping the URL to look like legitimate one. A technique used to do so is by adding prefix or suffix to the legitimate URL thus the user may not notice any difference.

4. Sub-domain(s) in URL: Another technique used by the phishers to deceive the users is by adding sub-domain(s) to the URL thus the users may believe that they are dealing with a credited website.

5. Misuse of HTTPs protocol: The existence of the HTTPs protocol every time sensitive information is being transferred reveals that the user certainly connected with an honest website. However, phishers may use a fake HTTPs protocol so that the users may be deceived.

6. Request URL: A webpage consists of a text and

Table 1 Sample of the websites features data

URL Anchor	Request URL	HTTPs	URL Length	Prefix Suffix	IP	Sub Domain	Sub-domain	Domain age	Class
-1	-1	1	-1	0	0	0	1	0	-1
1	1	-1	0	1	1	1	-1	-1	1
-1	0	1	-1	0	0	0	0	0	1
-1	1	1	1	1	0	1	0	0	-1
-1	1	1	0	1	0	1	0	0	1
-1	-1	-1	-1	0	0	0	0	-1	-1
1	-1	-1	0	0	1	1	-1	-1	-1

some objects such as images and videos. Typically, these objects are loaded to the webpage from the same domain where the webpage exists. If the objects are loaded from a domain different from the domain typed in the URL address bar the webpage is potentially suspicious.

7. URL of Anchor: Similar to "Request URL" but for this feature the links within the webpage might refer to a domain different from the domain typed on the URL address bar. This feature is treated exactly as "Request URL".

8. Website Traffic: Legitimate websites having high web traffic since they are visited regularly. Phishing websites often have short life thus their web traffic is either does not exist or its rank is less than the limit that gives it the legitimate status.

9. Age of Domain: The website is considered "Legitimate" if the domain aged more than 2 years. Otherwise, the website is considered "Phishy".

4.2. EXPERIMENTS RESULTS

Ten-fold cross-validation was utilised to evaluate the classification models and to produce error rates in the experiments.

Four dissimilar rule based classification algorithms which utilise a variety of rule learning methodologies have been considered for contrasting purposes with EMAC. These algorithms are CBA [2], PRISM [11], PART [11], and MCAR [4]. Our selection of the above classification algorithms is because firstly all these algorithms generate rules in the form of "If-Then" rules for fair comparison. Secondly, the chosen algorithms use different learning methodologies in discovering and producing the rules. The learning strategy exploited by CBA is based on Apriori association rule technique where frequent ruleitems are produced iteratively based on the minsupp threshold inputted by the end-user. On the other hand, MCAR uses vertical mining methodology to discover the rules. Mainly, it utilises ruleitem's locations in the training data set (tid-list) to perform tid-list intersections to compute the ruleitems's support and confidence which in turn are used to decide whether the ruleitem is a rule. Finally, PRISM is a covering algorithm that divides the data set into parts according to the available class labels and produces all rules for each class. For each class, it is starts with an empty rule and adds the highest expected accuracy for each possible

attribute value. It stops adding attribute values to the rule body when the candidate rule expected accuracy reaches 100% and at that point it generates the rules and removes all training data covered by the rules from the training data set. The algorithm repeats the same step until the data belonging to the selected class gets empty. Once this happens PRISM begins generating rules for another class and so forth. When the data in all parts are covered PRISM merges all rules derived for all class labels and forms the classifier. Lastly, PART algorithm is a combination of decision tree and rule induction algorithm that constructs partial decision trees.

The experiments were conducted on an I3 machine with 2.0 Ghz. The experiments of PRISM were carried out in Weka software [11]. For the AC algorithms (CBA, MCAR), CBA source code has been obtained from its prospective authors and (EMAC, MCAR) were implemented in Java.

Several researchers in AC, i.e. [2] [3] [4], have revealed that the minsupp threshold usually controls the numbers of rules generated. Thus, we have followed them in setting the support threshold to 1%-5% in the experiments of CBA, MCAR and the proposed algorithm. The confidence threshold, however, has less impact on the general performance of AC algorithms and we set it to 50% for CBA, MCAR and EMAC.

Figure 2 displays the classification accuracy of the compared algorithm on the nine phishing detection data set. It is obvious from the figure that the proposed algorithm is highly effective in predictive power when contrasted with other AC algorithms as well as rule based ones. Precisely, EMAC has outperformed PRISM and PART by 7.77% and 0.93% respectively. MCAR algorithm has slightly outperformed the proposed algorithm on the selected nine features data set by 0.21%. Though as we will see shortly, MCAR have produced 56 more rules in the classifier than EMAC, which is approximately 38% larger classifier to accomplish just 0.21% higher accuracy. We believe that there should be a trade-off between the number of rules produced and classification accuracy where one can accept smaller classifier in the exchange with slightly lower accuracy. One possible reason for the slight increase in the accuracy for MCAR over the proposed algorithm is the way it builds the classifier. In particular, MCAR evaluates each candidate rule derived in the learning phase on the training data set in which a rule is considered significant if it covers "correctly" at least one training data instance.
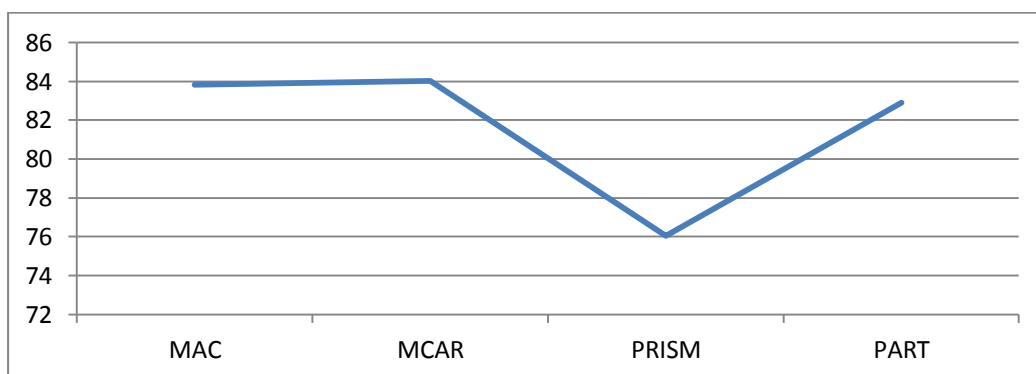
Fig. 2 The classification accuracy (%) for the contrasted algorithms derived from the phishing data

The coverage requires that:

1) The candidate rule body (attribute values) must be contained within the training instance

2) The class of the candidate rule and that of the training instance are similar

This rule evaluation process limits the data coverage per rule since the above tow conditions must be true in order to consider the rule to be part of the classifier. Alternatively, EMAC inserts the candidate rule into the classifier if only the first condition above is true relaxing the second condition (class similarity). This normally reduces overfitting by allowing the rule to cover larger portion of training cases, which shows the smaller classifiers produced by EMAC if compared to MCAR.

Figure 3 depicts the number of rules generated by the contrasted algorithms on the data set we consider in which it clearly shows that the proposed algorithm extracts smaller classifiers than MCAR and PRISM.

advantage of the simplicity of a data mining approach called associative classification that extracts simple yet effective classifiers containing easy to understand chunk of knowledge to solve the website phishing detection. Since phishing features are often correlated, we propose an algorithm that reduces the number of rules by using a novel evaluation method which cuts down the number of rules approximately by 38% if contrasted with other AC algorithms like MCAR and without effecting classification accuracy. The new algorithm has been compared with one AC and two rule based classification algorithms with respect to accuracy rate and classifier size on real websites data set. The data size is 1228 websites and it consists of nine significant features that have been collected from different online sources such as Phishtank and Yahoo directory. The features have been chosen after applying Chi-Square testing measure on larger numbers of features set. After experimentations, the results showed that the proposed algorithm scales well if compared to MCAR,

Fig. 3 The classifier size of MCAR and EMAC derived from the phishing data

The main reason for the fewer number of rules in EMAC classifier if compared to MCAR is due to the way EMAC constructs the classifier in which it considers the candidate rule part of the classifier when only its body is within the training instance and thus no class check is performed by EMAC.

This usually ends up of having the candidate rule covers large number of training instances and therefore several redundant rules will be discarded. In other words, some lower ranked rules will end up having no training data coverage and therefore they will be deleted. PRISM covering algorithm generates the largest classifier since it has no rule pruning at all. As a matter fact PRISM keeps producing rules per class labels as long as there are training instances exist which explains its very large classifiers. On the other hand, PART algorithm utilises rule induction and decision tree pruning heuristics to cut down the possible numbers of rules. To be more precise, it employs information gain approach from decision tree to build partial trees and then a pessimistic error and reduced error pruning methods are applied to remove candidate rules. This explains its small size classifier. Overall, AC algorithms such as MCAR and the proposed algorithm normally extract additional knowledge missed by classic rule based algorithms and thus they end up with more rules in the classifiers.

5. Conclusions

Phishing detection is a vital problem in the online community due to the massive numbers of online transactions performed by users. In this paper, we take

PART and PRISM. Specifically, our algorithm has higher accuracy by 7.77% and 0.93% than PRISM and PART respectively. MCAR has slightly outperformed our algorithm by 0.21% yet derived 56 additional rules in the classifier. In near future we intend to plug our algorithm in a browser to determine on the fly the phishing activity and alert users.

6. References

[1] F Thabtah, Q Mahmood, L McCluskey, and H Abdel-jaber, "A new Classification based on Association Algorithm," *Journal of Information and Knowledge Management*, vol. 9, no. 1, pp. 55-64, 2010.

[2] B Liu, W Hsu, and Y Ma, "Integrating Classification and Association Rule Mining," in *Knowledge Discovery and Data mining (KDD)*, 1998, pp. 80--86.

[3] W Li, J Han, and J Pei, "CMAR: Accurate and efficient classification based on multiple-class association rule," in *Proceedings of the ICDM'01*, San Jose, CA., 2001, pp. 369-376.

[4] F Thabtah, C Peter, and Y Peng, "MCAR: Multi-class Classification based on Association Rule," in *The 3rd ACS/IEEE International Conference on Computer Systems and Applications*, 2005, p. 33.

[5] X Wang, K Yue, W Niu, and Z Shi, "An approach for adaptive associative classification," *Expert Systems with Applications: An International Journal*, vol. 38, no. 9, pp. 11873-11883, 2011.

[6] W Liu, X Deng, G Huang, and A Y. Fu, "An Antiphishing Strategy Based on Visual Similarity Assessment," in *IEEE Educational Activities Department Piscataway*, NJ, USA, 2006, pp. 58-65.

[7] N Abdelhamid , A Ayesh , F Thabtah , S Ahmadi , and W Hadi , "MAC: A multiclass associative classification algorithm," *Journal of Information and Knowledge Management (JIKM)*, pp. 1250011-1 - 1250011-10, 2012.

[8] R M Mohammad, F Thabtah, and L McCluskey, "An Assessment of Features Related to Phishing Websites using an Automated Technique," in *The 7th International Conference for Internet Technology and Secured Transactions (ICITST-2012)*, London, 2012.

[9] M Aburrous, M A Hossain, K Dahal, and F Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert Systems with Applications: An International Journal*, pp. 7913-7921, December 2010.

[10] E Baralis, S Chiusano, and P Graza, "support thresholds in associative classification," in *Proceedings of the 2004 ACM Symposium on Applied Computing*, Nicosia, Cyprus., 2004, pp. 553-558.

[11] E Frank and I Witten, "Generating accurate rule sets without global optimisation," in *Proceedings of the Fifteenth International Conference on Machine Learning*, Madison, Wisconsin., pp. 144-151.

An Associative Classification Algorithm for Categorizing Text

[1]Bashar Aburumman, [2]Hussein Abdel-jaber

[1]*Department of Computing, Ablaqa'a Applied University, Jordan. bashar113@bau.edu.jo*
[2]*Faculty of Computer Studies, Department of Information Technology and Computing, Arab Open University,*
Kingdom of Saudi Arabia, habdeljaber@arabou.edu.sa

Abstract—*Constructing predictive, accurate and interpretable classifiers for categorizing texts is a significant task in text mining and knowledge discovery. Associative classification is a data mining research branch that applies association rule to form classification systems (classifiers). This classification approach has shown high accuracy performance in classifying unseen data if compared with other classification techniques. In this paper, we introduce a TC algorithm based on associative classification for the hard problem of TC where documents are classified based on keywords into the right categories. This algorithm employs an efficient intersection method based on DiffSet and a rule filtering procedure by considering more rules during building the classifier. Experimental results on real world textual data set called the Reuter indicated that the proposed model outperforms current rule based TC techniques.*

Keywords: Text mining, Associative classification, Data mining, Rule items.

1. Introduction

TC is an interesting research area which attracted several researchers because of the large quantities of textual documents online and offline. This field concerns about the automatic classification of textual data to one or multiple classes based on their content keywords [7]. Many different classification approaches were developed to categorize textual data, these approaches can be evaluated mainly by accuracy and the knowledge they produce. Moreover, these classification approaches range from high accuracy methods to low ones such as Naïve Bayes [13] which means that some of them produce high accurate classifiers and others low accurate ones. However, one fundamental measurement criteria is the understandability of the end-user of the resulting classifiers. A recently a new classification data mining technique called Associative Classification(AC) [10] is developed which combines high accuracy and understandability output together based on association rule. AC is a high efficient method that builds more predictive and accurate classification systems than traditional classification methods such as k-nearest algorithm[8] according to experimental studies [1, 4, 10, 18]. AC produces rule's based classifiers on the form (if → then) that are easy to understand and manipulate by end-user. This research is devoted to develop a new model based on AC for TC problem. Mainly we focus on two main steps in the TC problem and these are: Developing an efficient and fast intersection method based DiffSet [19] of the dEclat association rule and adopting it to unstructured

classification data. Secondly, proposing a rule filtering procedure that reduces the number of rules in the outputted classifiers by considering only partially matching during building the classifier. This novel method significantly minimized the number of rules described by the proposed model when compared with current AC models like MCAR [11]. Experimental results on real world textual data set called the Reuter indicated that the proposed model outperforms TC techniques.

2. Related Works

In this section, we summarise common TC learning techniques:

1-K-nearest neighbour (k-NN)

The k-nearest neighbour algorithm (k-NN) [8] is applied to evaluate the degree of resemblance between documents and (K) training data and storing particular amount of classification documents, so that specifying the class of documents that will be tested. The supposed category of a document is forecasted depending on the closet point that already allocated to a specified class. So this approach is simple and wide applied technique for text classification, but the accuracy of this approach is decreased by the existence of noisy or unrelated features and this technique requires high computational resources when the volume of training textual dataset increases.

2-Naïve Bayes Algorithm

Classification based on Naïve Bayes (NB) [13] is a simple statistical classification system that based on employing Bayes' Theorem with strong independence hypothesis. It is considered as probability model that based on independent feature model. The main feature of the Naïve Bayes classifier is that users need small amounts of training data to speculate the variables required for classification but the main drawback of NB classification approach is its relatively low categorization performance due to the conditional independence hypothesis is destructed by real-world data and the performance is bad when features are highly related and neglecting frequency of word occurrences.

3- Support Vector Machine (SVM)

The SVM classification approach depends on the structural risk reducing principle from computational learning theory [14]. The main idea for this principle is finding assumptions that ensure the lowest error rate. In addition, SVM requires positive and negative values for data training set rarely used

in other categorization methods. So it is an efficient statistical model for cases of large feature sets and is applied for cases in which the discriminator surface of two classes is not linear [2] but the main disadvantage of SVMs is their relatively complex training and classifying algorithms that consumes time and memory.

4- Associative Classification (AC)

AC is a research field in data mining that combines association rule discovery and classification [13]. The aim of AC is to build a model (classifier) that contains a number of rules (extracted knowledge) from labelled input data set and used for predicting the class for unclassified test data. Many research experimental studies [4, 10, 18] indicated that AC is a high efficient method that builds accurate and predictive classification systems better than traditional classification approaches such as decision tree [6] and The k-nearest neighbour algorithm (k-NN) [7]. Recently, several AC approaches have been developed such as CMAR [3], MCAR [11], and CACA [9]. AC algorithms learning process depends on two parameters named Support and Confidence, where the support of the item is number of transactions in D that contain the item. An association rule is an expression X $\Box$Y , where X, Y $\Box$ I are two sets of items and X∩Y=φ. While confidence is the probability a transaction contains Y given that it contains X, and is given as support(XY)/support(X)

3. The Proposed Algorithm

In this section, a new AC model is proposed that focuses on developing new frequent items and rule filtering techniques to help improving the rule discovery step and to reduce the number of rules produced during making the classifier. This intelligent model will automatically predict the type of any text document based on their content without using any predefined knowledge except the training textual data set and (the user MinSupp and MinConf thresholds). The new model consists of four main phases: the pre-processing phase, learning the rules, making of the classifier, and classifying new test data as shown in Figure 1.

3.1 Pre-processing Phase

Preparing the input data for mining is considered as an important phases in TC. Since the input textual data sets are often sparse, unstructured, and it may contain noise like records redundancy, incomplete transactions, missing values, and so on. Therefore, the quality of the constructed models is significantly affected by the quality of the input data set.

3.1.1 Stopwords Filtering

Text documents usually contain a lot of words that are not useful for the learning algorithms such as ('is', 'that', 'the'). Such words should be removed within pre-processing phase, because these words negatively affect classifier construction. In the proposed model, we adopted the most popular technique which is SMART stop word list since it is effective and has been utilized in many previous researches on TC.

3.1.2- Tokenization

It is a process that includes dividing the sequence characters into more meaningful Tokens. In TC, the text document is divided into sentences, and words .We use WEKA [16] filtering in the Explorer GUI to tokenize the input database.
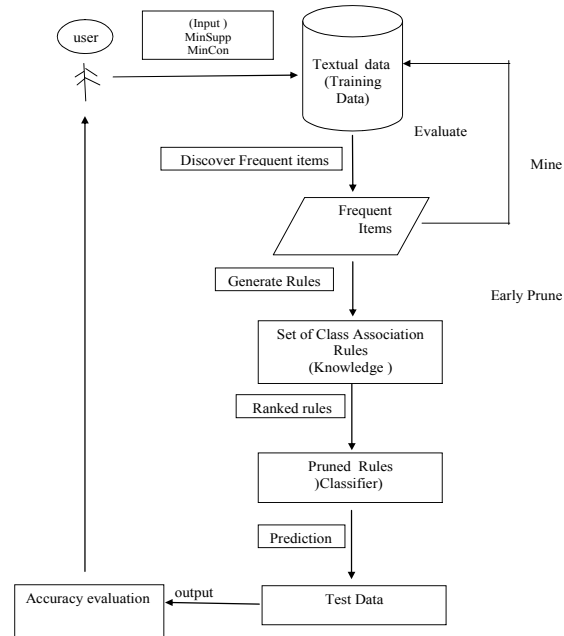
Figure1: main phases of the proposed model.

3.1.3 Stemming

It is the process of reducing derived words to their root, for example, 'construction' to 'construct'. We used in the proposed model a popular technique which is porter stemmer.

3.1.4 Data Representation

We adopted a data format in which input data is arranged as a group of columns where every column has a key identifier, it is called item identifier (IID) and a group of transaction identifiers (TIDs) [19]. We can use the TID of two or more different items of the same level to find locations where they occur together and this determine whether the new item is frequent. This is accomplished using TID intersection between frequent item discovered in step 1 to find possible frequent items in later iterations. This method has been used in [1] and showed promising results.

3.1.5 Features Selection

Most known TC algorithms represent a document as bag of words (BOW) ,the result of using the BOW is an explosion in the number of features .The major disadvantage of this representation is the high dimensions of feature space and information loss of the original texts. Feature selection [17] is one technique to solve such problems. The main idea of feature selection is to select a group of frequent terms that appears in the training set and employing this group as

2

features in TC. In our model we have used a Chi-square [17] to transform the high dimensionality of the Reuter text collection into a numeric matrix and then we use simple TID list intersections to compute the frequent items. The Chi-square is generally employed to evaluate the independence of two random variables, where two variables A and B are considered to be independent if P(AB)=P(A)P(B) or P(A|B)=P(A) and P(B|A)=P(B). In TC, the two random variables represent the frequency of the term (t) and frequency of the class (c).

3.2 Rule Learning

In our model we used dEclat algorithm [19] that utilized the vertical database layout in which we stored the difference of TIDs(Transaction ID's) called (diffset) between a candidate k item set and its prefix k-1frequent item sets, rather than the tids intersection set .We calculated the support by subtracting the cardinality of diffset from the support of its prefix k-1 frequent item set. It is worth mentioning that we are first researchers who adopt dEclat vertical mining in constructing classification models in TC .

```
DiffEclat([P]):
for all Xi 2 [P] do
for all Xj 2 [P], with j > i do
R = Xi [ Xj ;
d(R) = d(Xj) □ d(Xi);
if _(R) _ min sup then
Ti = Ti [ fRg; //Ti initially empty
for all Ti 6= ; do DiffEclat(Ti);
```

Figure 2: Pseudo-code for rule discovery [19].

dEclat is an enhancement of an earlier algorithm called Eclat [20] that uses a vertical database transaction layout, where frequent items are obtained by applying simple tid-lists intersections, without the need for complex data structures. This considerably reduces the size of the memory required to store the tids.
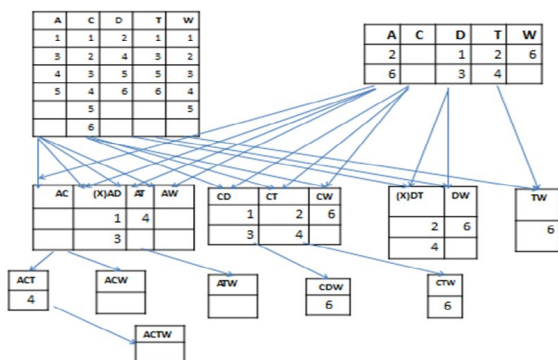
Figure 3: dElcat algorithm illustration [20].

The diffset approach avoids storing the complete tids of each item, rather the difference between the class and its member itemsets are stored. Two items share the same class if they share a common prefix. A class represents items that the

prefix can be extended with to obtain new class. Figure 3 illustrates rules generating using dEclat.

3.3 Rule Ranking

After we generate the rules, we rank them because the top sorted ranking rules play an important role in the classifying test data. The precedence of the rules is determined according to several measures involving confidence, support and rule antecedent length. For our model we consider rule's confidence first, then support and finally rule's length.

3.4 Classifier Construction

We check the produced rules by comparing them with the training data set cases, and only rules that cover at least one training data are kept for classification. So for each ordered rule starting from the top ranked one, we find all cases that match r1's body fully or partially. Once these training cases are located, they will be eliminated and the rule will be appended into the classifier. If the rule has no training data coverage it will be removed. The process is repeated for the remaining ordered rules until all training cases are covered or all ordered rules have been considered.

3.5 Prediction phase

The idea of the proposed prediction method is select the best rules among a set of high confidence rules in order to cover the test data. In classifying a test data, the first rule applicable to the test data classifies it. If no rules matching the test data, our prediction procedure uses the default class rule.

4 Experimental Evaluation

In this section, different classification algorithms are compared with the proposed model according to a number of text categorization evaluation measures including accuracy, number of rules produced, precision, recall and F1. The data set used in the experiments is the Reuter corpus and different classification techniques: MCAR, Decision trees [6], Naïve Bayes [1], and the k-nearest neighbour algorithm (k-NN) [8] have been contrasted with our model. The reason behind selecting these algorithms is to study the performance of our algorithm comparing to classic algorithms as well as AC ones.

4.1 Data Set

A popular TC collection called Reuter has been investigated in this section. The Reuters-21578 is the most widely used test collection for TC research. We used the ModApte version of Reuters-21578. This spilt leads to a corpus of 9,174 documents consisting of 6,603 training and 2,571 testing documents, respectively. We tested all contrasted algorithms on the seven most populated categories with the largest number of documents assigned to them in the training data set. The experiments are conducted on 2.5 GHz Pentium IV machine with 2GB RAM, and the proposed method is implemented using VB.Net programming language with a MinSupp and MinConf of 2%, and 40%, respectively, the

other algorithms were tested using WEKA[16],which is an open source business intelligence tool that implements different machine learning and data mining methods. Table 4.1 represents the number of documents for each category in the Reuters-21578 data set. On these documents we selected the top 1000 features using Chi Square.

Table 1: Number of documents per category (Reuters-21578).

Category Name	Training set	Testing Set
Acq	1650	719
Crude	389	189
Earn	2877	1087
Grain	433	149
Interest	347	131
Money-FX	538	179
Trade	369	117
Total	6603	2571

4.2 Result Analysis

Figure 4 represents the average precision results for the contrasted classification algorithms. The numbers in that figure give an indication that the proposed algorithm outperformed the remaining learning algorithm on the Reuter data set in regards to precision evaluation measure. In particular, it achieved higher precision on average 13.36%, 5.19%, 1.27% and 1.89% respectively than Naïve bayes, C4.5, Knn, and MCAR algorithms, respectively.   The second best performed algorithm with reference to precision was KNN but in general AC algorithms including our model and MCAR performed very competitive on precision figures if contrasted to traditional classification algorithms. The results on precision showed that the statistical method of Naïve bayes was the least scored precision algorithm on the Reuter TC data set. Figure 5 shows again that the Naïve bayes algorithm is the least applicable classification approach towards the Reuter data set due to the low results of recall. This is since Naïve bayes algorithm derives often high class likelihood for majority classes since they are often more representative in the input textual collections like "Earn". In the Reuter data collection, there are some classes that are connected with low numbers of documents. Further, Naïve bayes algorithm is based on independence assumption which is violates real-world data and perform very poorly when features are highly correlated and does not consider frequency of word occurrences. On the other hand, the proposed algorithm and MCAR for instance produced classifiers that represent most of the classes in the input textual data set. Meaning they are not impacted with the unbalanced class labels for the Reuter's documents. In general, most of the considered classification algorithms except Naïve Bayes showed competitive performance with regards to precision and recall on the data set, as their derived results are close to each other.
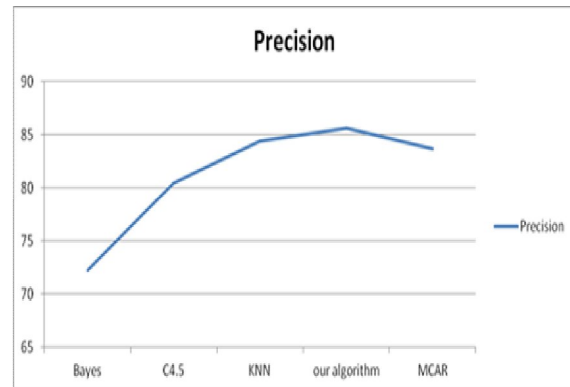
Figure 4: Average precision of the contrasted algorithm.
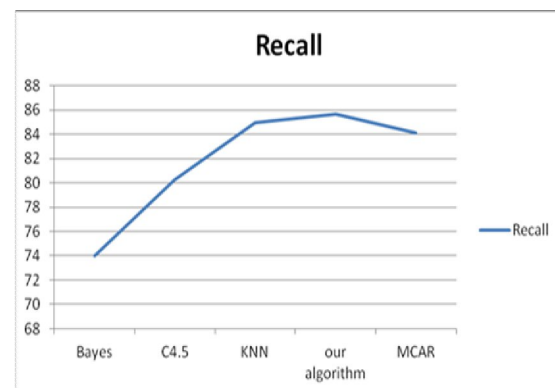
Figure 5: Average recall of the contrasted algorithm.

A deeper investigation on the performance of the documents category has been conducted to show the behaviour of the contrasted algorithms. Tables 2 and Table 3 illustrate the precision, and recall results for each document category and for each algorithm we consider. The recall and precision measures results for proposed algorithm for the document's categories are consistent except the "Interest" class which has achieved the least results. After investigating the Reuter data collection for such class it seems that during the prediction step, the proposed algorithm has misclassified many of the   "Interest" to the rest of the documents classes. This is since the "Interest" class documents contain general terms and the classifier has a hard time to predict its right class. The same principle applies to the most of the classification algorithms used in the experiments in the same context, e.g."Trade" and "Interest" classes.

Figure 6 shows the harmonic mean (F1) results for the contrasted classification algorithms; we notice that the F1 results of the compared algorithms are consistent with previous recall and precision results. This is because the harmonic mean measure depends on recall and precision values. Also the F1 figures indicate that KNN outperformed other traditional TC algorithms due to its high accurate measures results. Lastly, it seems that Naïve bayes is the least applicable algorithm to the TC problem.

4

Table 2: Precision results for each class in the Reuter and by each algorithm.

Class	N. Bayes	C4.5	KNN	Our Algorithm	MCAR
Acq	90.8	85.7	91.5	94.78	90.3
Crude	79.9	74.7	85.1	81.9	88.5
Earn	95	96.3	97.4	99.9	100
Grain	71.7	97.6	88.5	95.44	94.5
Interest	56.5	56.11	73.6	54.9	42.7
Money-	64	76	77.3	76.18	73.4
Trade	47.8	77.12	77.6	96.12	96.6
Average	72.2	80.5	84.4	85.6	83.6

Table 3: Recall results for each class in the Reuter and by each algorithm.

Class	C4.5	KNN	our algorithm	MCAR	N. Bayes
Acq	92.2	84.9	92.5	94.44	90.1
Crude	82.1	76.3	86.3	83.7	87.7
Earn	96.8	95.9	97.2	99.6	99.6
Grain	73.3	98.6	87.9	96.76	96.1
Interest	59.5	48.7	74.4	53.88	44.12
Money-FX	61.8	77	79.1	76.88	75.2
Trade	52.2	80.5	77.2	94.6	95.8
Average	73.9	80.2	84.9	85.6	84

Figure 6 shows the harmonic mean (F1) results for the contrasted classification algorithms; we notice that the F1 results of the compared algorithms are consistent with previous recall and precision results. This is because the harmonic mean measure depends on recall and precision values. This figure insures that the proposed algorithm is the most suitable classification algorithms among the contrasted ones when it comes to the TC. In addition, our classification algorithm outperformed MCAR which is a known AC technique with reference to F1. Lastly, it seems that Naïve bayes is the least applicable algorithm to the TC problem.
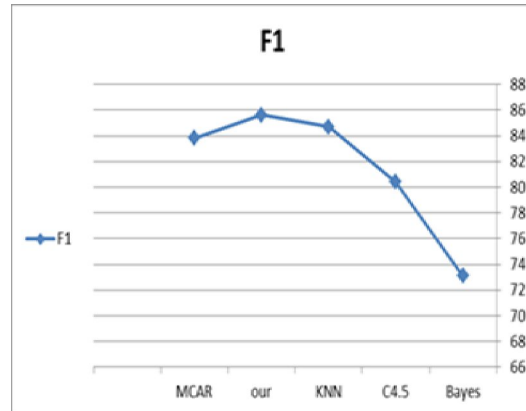
Figure 6: F1 average for the contrasted algorithms.

5 . Conclusions

In this paper, we proposed an AC algorithm for classifying textual data. The proposed algorithm showed high accuracy comparing to traditional classification approaches which indicates that the proposed model is efficient and applicable to the TC problem. Particularly, we utilized a new vertical mining based on DiffSet approach to find the rules. Experiments results using the proposed algorithm and four other known classification algorithms have been carried out. We used precision, recall and harmonic mean (F1) as the measures for comparing the algorithms. The results revealed that thee proposed algorithm outperformed MCAR, decision trees, Naïve Bayes and KNN on the Reuter data set and with the respect to the above mentioned measures. In near future we will look into using the proposed algorithm on uncertain data sets.

References

[1] Abdelhamid N., Ayesh A., Thabtah F., Ahmadi S., Hadi W (2012) MAC: A multiclass associative classification algorithm. To be published in the *Journal of Information and Knowledge Management* (JIKM). 11 (2), pp. 1250011-1 - 1250011-10. WorldScinet.

[2] Ghahreman N, Dastjerdi A (2011) " Semi-Automatic Labeling of Training Data Sets in Text Classification " Canadian Center of Science and Education Vol. 4, No. 6.

[3] Li, W., Han, J. and Pei, J. (2001). " CMAR: Accurate and efficient classification based on multiple-class association rule". In Proceedings of the International Conference on Data Mining (ICDM'01), San Jose,CA, pp. 369–376.

[4] Liu B,. Hsu W, and. Ma Y(1998)," Integrating classification and association rule mining". In Proc. of SIGKDD, pages 80.86.

[5] Mehta M, Agrwal R ,Ressanen J,(1999) "SLIQ: fast scalable classifier for data minig ",Proceeding Intel

conference ,Extending   Database technology (EDBT 96) ,France .

[6] Quinlan, J. (1998)" C4.5: Programs for machine learning". San Mateo, CA: Morgan Kaufmann. the KDD, (pp. 80-86). New York, NY.

[7] Sebastiani, F. (2002). "Machine learning in automated TC". ACM Computing Surveys, 34(1), 1-47.

[8] Tam, V, Santoso, A, and Setiono, R.(2002) , "A comparative study of centroid-based, neighborhood-based and statistical approaches for effective document categorization", Proceedings of the 16th International Conference on Pattern Recognition, pp.235–238.

[9] Tang Z. and Liao Q. (2007). "A New Class Based Associative Classification Algorithm". IMECS 2007: 685-689.

[10] Thabtah,F.,Cowling,P., and Peng, Y(2004)."MMAC: A   New   Multiclass,   Multi-label   Associative Classification Approach". In Proceeding of fourth IEEE International Conference on Data Mining (ICDM .04), pages 217-224. Brighton, UK .

[11] Thabtah, F., Cowling, P., and Peng, Y(2005)." MCAR: Multi-classClassification based on Association Rule Approach". ADIT journal of engineering , vol 2,no 1.In Proceeding of third IEEE International Conference on Computer Systems and Applications . pages 1-7. Cairo,Egypt. 2005.

[12] Thabtah F (2007) . "A review of associative classification   mining"   .Knowledge   Engineering Review, 22 (1). pp. 37-65

[13] Thabtah F., Mahmood Q., McCluskey L., and Abdel-jaber   H (2010a). "A new Classification based on Association Algorithm". Journal of Information and Knowledge Management, Vol 9, No. 1, pp. 55-64. World Scientific .

[14] Vapnik V. (1995)." The Nature of Statistical Learning Theory", chapter 5. Springer-Verlag, New York.

[15] Wang X , Yue K , and Shi Z (2011)," An approach for adaptive associative classification" Expert Systems with Applications Volume 38, Issue 9, Pages 11873–11883.

[16] WEKA: Data Mining Software in Java: http://www.cs.waikato.ac.nz/ml/weka.

[17] Yang, Y. and Pederson, J.O. (1997). "A comparative Study on Feature Selection in TC", In Proceedings of the 14th International Conference on Machine Learning, pp. 412-420.

[18] Yin, X. and Han, J. (2003):" CPAR   Classification based on Predictive Association Rules". In Proc. of the Int. Conf. on Data Mining, SDM. SIAM.

[19] Zaki M and Gouda.K (2003). "Fast Vertical Mining Using Diffsets", pp. 326-335.

[20] Zaki M.J., Parthasarathy S., Ogihara M., and Li W.(1997)." New algorithms for fast discovery of association rules". In Proceeding of third KDD Conference, pages 283-286.

.

SESSION

EXPERT SYSTEMS FOR INTELLIGENT AUTOMATION PURPOSES IN POWER SYSTEMS

Chair(s)

Prof. Ivan Nunes da Silva

Application of Fuzzy System for Analysis of Lightning Overvoltages and Protection of Distribution Systems by Surge Arresters

(Special Session – ICAI'2013: Expert Systems for Intelligent Automation Purposes in Power Systems)

M. A. Araújo, R. A. Flauzino (IEEE member), O. E. Batista, L. A. Moraes (IEEE member)

Department of Electrical and Computer Engineering, University of São Paulo (USP), São Carlos, Brazil.

marcel.araujo@usp.br, raflauzino@sc.usp.br, oureste.batista@usp.br, lucas.moraes@usp.br

Abstract—The protection of electric power systems against temporary overvoltages, due to switching and lightning contributes directly to the quality of the energy supplied to consumers. This paper proposes the implementation of an overvoltage protection system composed of metal-oxide surge arrester, from the application of a fuzzy system for location the point of occurrence of flashover and definition the amplitude of the lightning overvoltages. Furthermore, it is to develop a procedure for the specification of the sizing, placement and total number of surge arresters for protection of distribution lines optimized from the data provided by the fuzzy system.

Index Terms—lightning, lightning overvoltage, surge arrester, fuzzy systems.

I. INTRODUCTION

In the electric power sector, the criteria of reliability and continuity of supply are critical to the constant research and development of procedures to improve the Power Quality (PQ) and related services available to consumers. To achieve adequate operating conditions in an Electric Power System (EPS), and consequently a high level of PQ, two of the requirements are the installation of a Lightning Protection System (LPS) and a grounding system.

The LPS consists of surge arresters, which lead the atmospheric discharge to the ground, at the same time they limit voltage on the equipment to which they provide protection. This limit voltage is composed by the sum of the voltage discharge of the surge arrester and by the induced voltage developed by the discharge current between the surge arrester line and ground leads [1]. The study of LPSs can be employed to solve various problems, some of the most important being:

- Analysis of lightning induced voltages [2];
- Reduction of unplanned shutdowns which result voltage sags and load shedding [3];
- Definition of the location and number of surge arresters to be installed [4];
- Determination of influence of surge arrester placement on phases on the level of shutdowns of the line and on the level of energy absorption of surge arresters [4], [5];

- Comparative assessment of the use of line surge arresters in classical methods of controlled shutdowns in the event of lightning [6];
- Analysis of the use of shield wire and surge arresters with or without spark-gap and assessment of the performance of a line in the event of lightning, so as to include critical regions [7].

In this context, the improvement of the response of distribution lines to lightning induced overvoltages by the installation of shield wire and surge arresters was analyzed in [8]. The effects of footing impedance and also shield wire size on line protection are assessed. The line performed better with the surge arrester than with shield wire, especially if the surge arresters were installed on every pole of the line.

A useful and complete guide to the protection of power systems by surge arresters is available in [9]. This guide discusses the use of metal-oxide surge arresters to protect equipment with rated voltages above 1 kV against overvoltages. In addition, it provides information about the use of surge arresters for protection of substations, transmission and distribution systems and large electric machines.

In [10], a lightning protection system for a Transmission Line (TL) is described, employing a minimum number of surge arresters, placed on different towers with varying distances between them. The dependence of these placements on the tower footing resistance was analyzed. With the implementation of these methods of analysis, the number of unplanned shutdowns was reduced and the reliability of the services provided to the consumer improved.

Such cases show the importance of the study of lightning protection, as well as the relevance of analyzing the sizing, location and number of surge arresters to be installed. Thus, this paper will first present a method to locate the point of occurrence of flashover and respective overvoltage based on fuzzy logic. Then will show a procedure for protection against overvoltages of atmospheric origin, based on the use of surge arresters. To this end, the consequences of direct lightning strikes on Distribution Line (DL) will be investigated.

The rest of the paper is organized as follows: the next section presents the main questions regarding overvoltages of atmospheric origin in power system. The third part gives details about the selection, modeling and implementation of metal-oxide surge arresters. In the last section presents the method of locating the event point of flashovers caused by lightning through a fuzzy system, the responses of computer simulations and the findings of this study.

II. OVERVOLTAGES OF ATMOSPHERIC ORIGIN IN ELECTRIC POWER SYSTEMS

The incidence of lightning on the conductors of TLs and DLs (direct lightning) or in the vicinity (indirect lightning) can cause overvoltages of high magnitude in the lines. If the amplitudes of these overvoltages exceed the levels that the system can withstand, disruptive discharges will occur, which can evolve into electric arcs. This can result in faults between one or more phases and ground and the need to actuate protection devices against overcurrents.

The Rusck model [11], presented in (1), provides the maximum value of the induced overvoltage (U_{max}) at the center of a line relative to the point of discharge of the lightning, for a single-phase infinitely long line over a perfectly conducting ground

$$U_{max} = Z_0 \frac{I_{ph}}{d}\left(1 + \frac{v}{\sqrt{2-v^2}}\right), \tag{1}$$

where Z_0 is the characteristic impedance of the line, Ip the discharge current, h the height of the line, d the distance from the discharge to the center of the line and v the ratio between the discharge wave propagation speed and the speed of light.

In systems with operating voltages exceeding 200 kV, the insulation levels of the equipment withstand overvoltages induced by direct and indirect discharges, because in extra high voltage systems, surges of greater impact occur due to switching operations. However, in transmission and distribution systems with nominal voltages below 200 kV, without protective devices, such as the time analyzed in this paper, direct lightning strikes entail overvoltages and dielectric failure, resulting in the burning of equipment and probably physical damage to its components.

Although the direct incidence of lightning on DLs is less frequent than the indirect one, the study of methods of protection against overvoltages of atmospheric origin on these lines is of great value. This is due to severity of these overvoltages for DLs and because on open field lines in rural areas, which may be the highest structures in its surroundings, and in less occupied urban areas, their occurrence is more likely.

III. MODELING OF METAL-OXIDE SURGE ARRESTER

The surge arresters are one of the most commonly used devices for protection and insulation coordination of various types of electrical equipment and systems, and thus were chosen to be modeled in this article. The zinc oxide (ZnO) surge arresters, currently, are the type often used on DLs and TLs by virtue of the nonlinear characteristic of their blocks of resistors in the region of low intensity current, because this property implies a lower drained leakage current. Other features that make them used the most are short response time to transients, low residual voltage, high thermal stability and high energy absorption capacity when faced with temporary and transient overvoltages [12]-[14].

Thus, next will be presented the requirements for implementation of the Pinceti and Giannettoni model [15] in ATP software, run in the graphical interface ATPDraw, as well as the electrical characteristics of the surge arresters needed on the line employed in this analysis.

A. Characteristics of Pinceti and Giannettoni Model

The Pinceti and Giannettoni model was validated by comparison with the conventional ATP model and with the workgroup 3.4.11 model of the IEEE [16]. This model was chosen because it represents adequately the dynamic characteristics of the surge arrester, it requires only electrical parameters to build the circuit and it does not need iterative corrections during its computer simulations. These features make it a very attractive model in terms of computational effort and the ready availability of necessary data from manufacturers.

Fig. 1 illustrates the model proposed by Pinceti and Giannettoni, and (2) and (3) demonstrate the parameters necessary to determine inductances L_0 and L_1. In addition, in (2) and (3), Vn corresponds to the nominal voltage of the, varistor to $V_{R1/T2}$ the residual voltage due to a current surge of modulus 10 kA and 1.2 µs rise time, $V_{R8/T2}$ the residual voltage for a current of wavefront 8 µs and $V_{R8/20}$ the residual voltage for a current of 10 kA and waveform 8x20 µs.
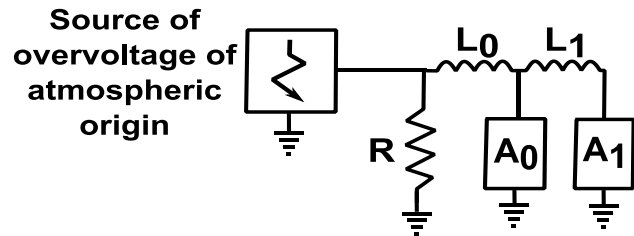
Figure 1: Model proposed by Pinceti and Giannettoni, adapted from [15].

$$L_0 = \frac{1}{12}\frac{V_{R1/T2} - V_{R8/T2}}{V_{R8/20}}V_n\,[\mu H], \tag{2}$$

$$L_1 = \frac{1}{4}\frac{V_{R1/T2} - V_{R8/T2}}{V_{R8/20}}V_n\,[\mu H]. \tag{3}$$

The characteristics of nonlinear resistors A_0 and A_1, derived from the model proposed in [16], are determined by multiplying voltages A_0 and A_1 in Table I by the residual voltage due to a surge current of modulus 10 kA and 1.2 µs rise time, and will be defined in subsection B.

TABLE I: CHARACTERISTICS OF THE NONLINEAR RESISTORS ACCORDING TO [15].

Current (kA)	A_0 Voltage (pu)	A_1 Voltage (pu)
2.00E-06	0.81	0.623
0.1	0.974	0.788
1	1.052	0.866
3	1.108	0.922
10	1.195	1.009
20	1.277	1.091

B. Electrical characteristics of the modeled line and surge arrester

The sizing of the surge arrester chosen for the line under study was based on the line operating data, the overvoltages determined by the simulations of lightning and the data provided in a surge arrester catalog.

The line parameters were taken from the data in [17], which show that the nominal voltage was 69 kV phase-phase and 39.8 kV phase-to-ground. The maximum amplitude of the lightning current was taken as 10 kA, since this was the most frequent value. Also, the keraunic index (I_k) of the region of the line is 60, according to the Brazilian standard [18], which results in a Flash Density (FD) of 1.89, calculated by (4).

$$FD = 0.0024 I_k^{1.63}. \tag{4}$$

From these data and the manufacturer's information contained in [19], the electrical characteristics of the surge arrester modeled in this article were defined. For each system voltage, the combined data from Tables II, III and IV, which contain the guaranteed protection characteristic of surge arresters, provide a range of values for the maximum system voltage (Um) and for the nominal voltage (Ur).

TABLE II: GUARANTEED PROTECTION RANGE OF SURGE ARRESTER, ADAPTED FROM [19].

Maximum System Voltage	Nominal Voltage	Maximum Residual Voltage for 8x20μs Wave of Current	
Um	Ur	5kA	10kA
[kVrms]	[kVrms]	[kV peak]	[kV peak]
52	42	103	109
	48	118	125

TABLE III: MINIMUM Ur OF SURGE ARRESTER FOR LIGHTNING, ADAPTED FROM [19].

System Grounding	Fault Duration	System Voltage Um [kV]	Minimum Nominal Voltage Ur [kV]
Effective	≤1s	≤100	≥0.8xUm
Effective	≤1s	≥123	≥0.72xUm

TABLE IV: LIGHTNING PROTECTION CHARACTERISTIC OF SURGE ARRESTER, ADAPTED FROM [19].

Maximum System Voltage	Nominal Voltage	External Insulation		
Um	Ur	1.2/50 μs	60 Hz	250/2500 μs
		dry	humid (10s)	humid
[kVrms]	[kVrms]	[kV peak]	[kVrms]	[kV peak]
52	42-60	310	150	250
	66	370	180	300

Considering that the chosen surge arrester must have a continuous operating voltage above the system nominal voltage, using the Table II, it was adopted one of 52 kV, a value immediately above the phase voltage of the line. To determine Ur, Table III was analyzed, according to which Ur≥0.8xUm, since the grounding of the line is taken as effective and thus Um≤100 kV; that is, Ur is 41.6 kV. Thus, in Table II the value equal to or just above that found for Ur should be taken; thus, the Ur adopted is 42 kV.

Next, the parameters of (2) and (3) were determined from the data in Tables II and IV, which gives V_n equal to 42 kV, $V_{R1/T2}$ equal to 310 kV, $V_{R8/T2}$ and $V_{R8/20}$ equal to 109 kV, L_1 equal to 19.36 μH, L_2 equal to 6.45 μH and characteristics of nonlinear resistors A_0 and A_1, as shown in Table V.

TABLE V: CHARACTERISTICS OF NONLINEAR RESISTOR IN THE SURGE ARRESTER MODELED.

Current (kA)	A_0 Voltage (kV)	A_1 Voltage (kV)
2.00E-06	251.1	193.13
0.1	301.94	244.28
1	326.12	268.46
3	343.48	285.82
10	370.45	312.79
20	395.87	338.21

C. Implementation and simulation of the surge arrester model

The circuit in Fig. 1 was implemented with nonlinear resistors parameterized with data previously recorded with a 1 MΩ linear resistor, to avoid numerical instabilities during the simulation. A source was used to simulate lightning of waveform 1.2x50 μs and discharge current 10 kA. The response of the simulation, is displayed in Fig. 2, in which the consistency of the behavior of the model may be noted, because when the surge arrester was subjected to a voltage greater than its nominal voltage, its impedance fell, allowing the atmospheric impulse current to be dissipated. Also verifies that the value of the overvoltage resulting was limited, and recovered the high impedance value of surge arrester after the end of the request voltage.
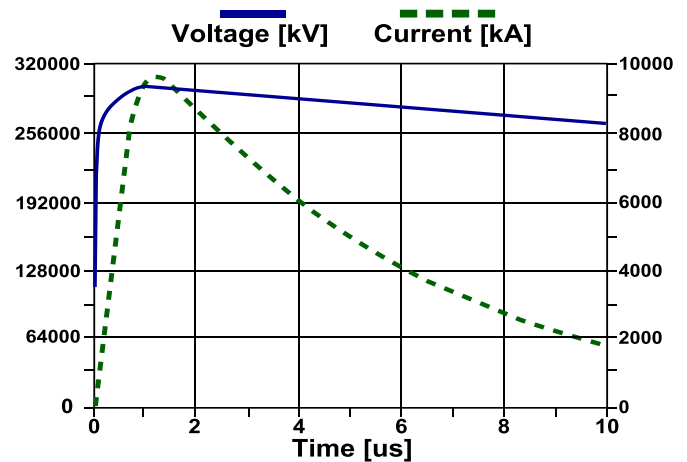
Figure 2: Response of surge arrester model to waveform 1.2x50 μs and discharge current 10 kA.

IV. RESULTS

To analyze the incidence of direct lightning on the line under study, initially, simulations were carried out of lightning with waveform 1.2x50 µs discharge currents 10 kA, 4.5 kA or 2.5 kA, and impedance of the air ionization channel 1 kΩ or 3 kΩ. Next, sections of the line were configured according to the JMarti model, since in this type of simulation, the dependence on frequency is relevant [20]. And also it was used PI model because the line length was 8.915 kilometers and therefore less than 15 kilometers, as required in [21] for this type of simulation.

Was simulated the applying lightning in the 9 points of the line shown in Fig. 3, with which were obtained voltages and currents for each of the three phases at each point of the line.
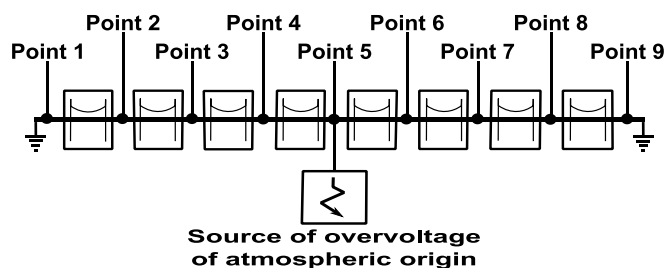
Figure 3: Simplified model of the line for simulations of lightning.

Thus, with the data from the simulations has built a multilayer fuzzy inference system, as shown in Fig. 4 and Fig. 5, from the automatic structural and parametric adjustment presented in [22] and [23]. After obtaining the currents and voltages, these data were converted to the respective linguistic terms that were specified through relevance functions. Then, the linguistic terms were used in the evaluation of fuzzy rules, and finally, by the application the inference procedures, by applying the inference procedures, results were obtained which allowed identify the point of happening flashover and the amplitude their overvoltages.
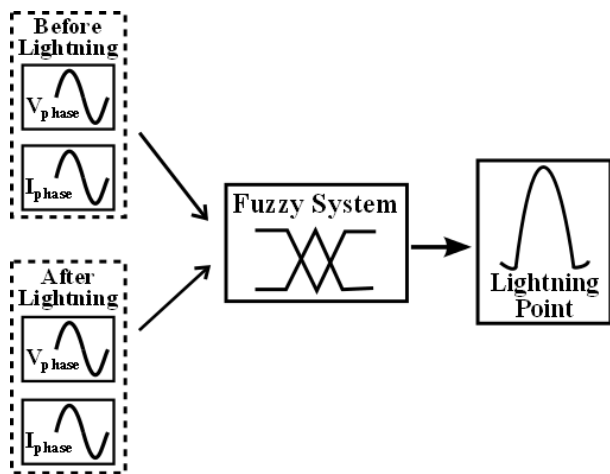
Figure 4: Fuzzy system for locating the point of incidence of lightning.
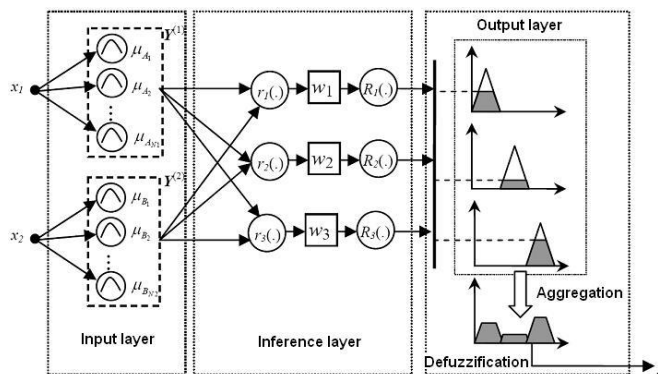
Figure 5: Multilayers fuzzy system.

For the purpose of evaluating the performance of the proposed method, a set of test samples was applied to the fuzzy system. The estimation showed average error of 0.0055%, standard deviation of 0.0143% and maximum error of 0.0437%. Fig. 6 shows the results of this test, in which each level represents a point of event flashover, and where it is possible to note the precision of the method developed. Emphasize that the point of occurrence of flashover is defined by the peak value of voltage assess and by the place where it is measured. Already Fig. 7 contains the histogram of errors, from which it can be seen that the vast majority of the errors occur when its percentage value is very low, again showing the accuracy of the procedure implemented.
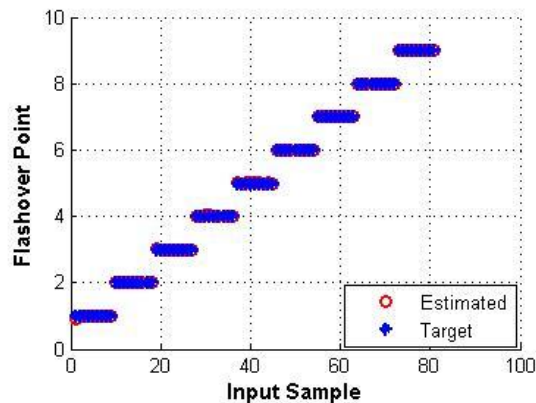
Figure 6: Test of the locating the point of flashover happening.
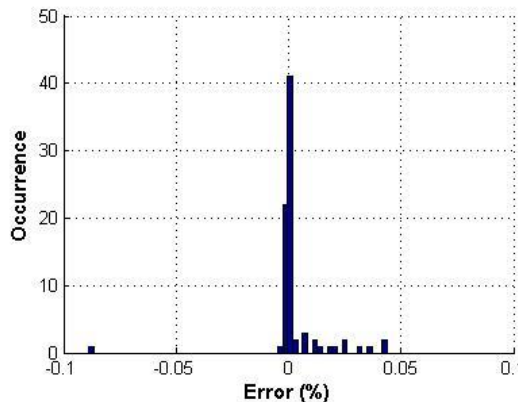
Figure 7: Histogram of the error of point location of the event Flashovers.

From the most critical situation, new simulations of lightning were carried out, with the addition of 2, 3 and 5 sets of surge arresters at points 1 and 9, 1, 5 and 9, and 1, 3, 5, 7 and 9, respectively. It should be noted that a set of surge arresters consists of 3 surge arresters, one per phase, and that the overvoltages on the line were acquired by meters located at points 2, 5 and 8, in all situations.

With the addition of 2 sets of surge arresters at the ends of the line, that is, at points 1 and 9, the overvoltages generated on the line and voltages and currents in the surge arresters as shown in Table VI. From this table it is concluded that 2 sets of surge arresters are not sufficient to protect the line, since the maximum overvoltage measured at point 5 is 2,586 kV.

However, the surge arrester was observed to function correctly, because the maximum residual voltage measured on its terminals was 265 kV, which does not exceed the limit of 310 kV established by the manufacturer. In addition, the greatest current found, which was 8.7 kA, also does not exceed the limit of 10 kA withstand able by the surge arrester.

TABLE VI: MAXIMUM VOLTAGE AND CURRENT VALUE ON THE LINE WITH 2 SETS OF SURGE ARRESTERS.

Line with 2 sets of surge arresters					
Equipments		Point of Measurement	Phase A	Phase B	Phase C
Surge Arrester 1	Current [A]	1	3,181	-3,049	8,754
	Voltage [kV]	1	-233	-220	265
Surge Arrester 2	Current [A]	9	2,909	2,118	8,159
	Voltage [kV]	9	-209	-184	242
Meter of voltage [kV]		2	-819	-569	1,408
Meter of voltage [kV]		5	-1,599	-1,442	2,586
Meter of voltage [kV]		8	679	492	1,985

Later, with the addition of 3 sets of surge arresters at points 1, 5 and 9, as illustrated in Fig. 4, the results in Table VII were obtained, where it is seen that the line is protected, since the maximum overvoltage measured in this situation was found at point 2 and was 244 kV. Moreover, the residual voltage and the current in the surge arresters did not exceed the limits established by the manufacturer, because they were 162 kV and 9.4 kA, respectively.
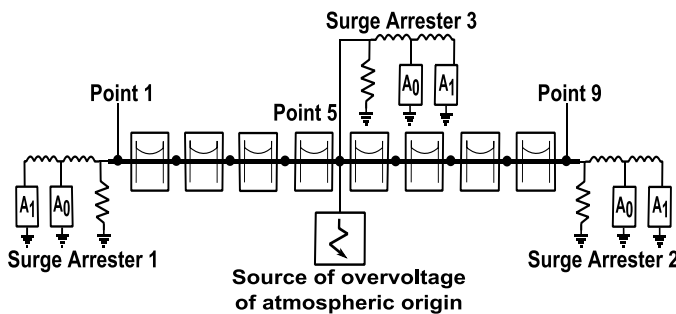
Figure 4: Simplified model of line with surge arresters at points 1, 5 and 9.

TABLE VII: MAXIMUM VOLTAGE AND CURRENT ON LINE WITH 3 SETS OF SURGE ARRESTERS.

Line with 3 sets of surge arresters					
Equipments		Point of Measurement	Phase A	Phase B	Phase C
Surge Arrester 1	Current [A]	1	-322	-289	-454
	Voltage [kV]	1	-114	-133	142
Surge Arrester 2	Current [A]	5	0.94	1.61	9,480
	Voltage [kV]	5	148	132	162
Surge Arrester 3	Current [A]	9	-141	-97	364
	Voltage [kV]	9	-112	-106	125
Meter of voltage [kV]		2	-136	-111	244
Meter of voltage [kV]		5	155	132	161
Meter of voltage [kV]		8	-118	-103	133

Finally, adding 5 sets of surge arresters at points 1, 3, 5, 7 and 9, the data in Table VIII were obtained, where it is found that the line is also protected with this configuration, because the maximum overvoltage was measured at point 2 and is 237 kV. Also, the residual voltage and the current in the surge arresters did not exceed the limits established by the manufacturer, reaching 179 kV and 9.2 kA, respectively.

TABLE VIII: MAXIMUM VOLTAGE AND CURRENT ON LINE WITH 5 SETS OF SURGE ARRESTERS.

Line with 5 sets of surge arresters					
Equipments		Point of Measurement	Phase A	Phase B	Phase C
Surge Arrester 1	Current [A]	1	-122	-157	263
	Voltage [kV]	1	-78	-118	124
Surge Arrester 2	Current [A]	3	-157	-164	578
	Voltage [kV]	3	-134	-128	179
Surge Arrester 3	Current [A]	5	1.83	2.49	9,247
	Voltage [kV]	5	84	97	110
Surge Arrester 4	Current [A]	7	-178	-122	7,154
	Voltage [kV]	7	-138	-124	177
Surge Arrester 5	Current [A]	9	58	32	197
	Voltage [kV]	9	-92	-91	112
Meter of voltage [kV]		2	-124	-102	237
Meter of voltage [kV]		5	138	113	146
Meter of voltage [kV]		8	-97	-91	136

V. CONCLUSIONS

With the results obtained it is concluded that the implemented fuzzy system for to identify the point of occurrence of flashovers was efficient and robust, as exposed by the responses to the test performed and found by the measured errors. Furthermore, it is observed that the fuzzy system allows greater agility for analysis of lightning overvoltages under the DL analyzed since it determines the location of Flashovers and the amplitude their voltages at these points, contributing determination the most critical point that must be protected in line.

The overvoltages resulting from discharges on the line, with 3 and 5 sets of surge arresters, feature very few

differences between one another. Therefore, considering the technical/economic viability of the line protection scheme, the best solution is to use only 3 sets of surge arresters arranged at points 1, 5 and 9, balancing the protection offered against the high cost of buying and installing these devices. It was also concluded that the use of only one set of surge arresters at any point on the line, including in section 5, which the highest overvoltage occurs, is not sufficient for its protection and for this reason these results were not presented.

Finally, it should be noted that this article fulfills its objective of contributing with methods and information about the project of a protection system against overvoltages of atmospheric origin based on sizing and specification of surge arresters. We point out that the procedures carried out on the line in question could be applied to other electrical systems, provided that be their electrical characteristics for sizing, allocation and quantification of surge arresters to be installed.

REFERENCES

[1] "IEEE Guide for Improving the Lightning Performance of Electric Power Overhead Distribution Lines", *IEEE Std 1410-2010 (Revision of IEEE Std 1410-2004)*, pp. 1-73, 2011.

[2] S. Yokoyama, K. Miyake, H. Mitani, and N. Yamazaki, "Advanced observations of lightning induced voltage on power distribution lines", *IEEE Transactions on Power Delivery*, Vol. 1, n° 2, p. 129-139, 1986.

[3] D. C. Miranda, L. V. Cunha, S. Visacro, R. N. Dias, and C. R. Mesquita, "Resultados da aplicação de metodologia inovadora para melhoria de desempenho da LT Guilman Amorim – Ipatinga 1, 230 kV, frente a descargas atmosféricas", *XIX Seminário Nacional de Produção e Transmissão de Energia Elétrica (SNTPEE)*, Rio de Janeiro, RJ, 2007.

[4] T. A. Short and R. H. Ammon, "Monitoring Results of the Effectiveness of Surge Arrester Spacing on Distribution Line Protection", *IEEE Transactions on Power Delivery*, vol. 14 pp. 1142-1150, July 1999.

[5] K. G. Ringler, P. Kirkby, C. C. Erven, M. V. Lat and T. A. Malkiewicz, "The Energy Absorption Capability and Time-to-Failure of Varistors Used in Station-Class Metal-Oxide Surge Arresters", *IEEE Transactions on Power Delivery* vol. 12, n°. 1, pp. 203-212, Jan. 1997.

[6] T. E. McDermott, T. A. Short and J. G. Anderson, "Lightning Protection of Distribution Lines", *IEEE Transactions of Power Delivery*, vol. 9, no. 1, pp. 138–152, Jan. 1994

[7] A. Piantini, "Lightning Protection of Overhead Power Distribution Lines", In: 29th International Conference on Lightning Protection - ICLP 2008, Uppsala, Sweden, p. 1-29, 2008.

[8] T. Thanasaksiri, "Improving the lightning performance of overhead distribution line", *TENCON 2004, IEEE Region 10 Conference*, Vol. 3, p. 369-372, 2004.

[9] *IEEE Guide for the Application of Metal-Oxide Surge Arresters for Alternating Current-Systems.* IEEE Std. C62.22-2009, July, 2009.

[10] K. Munukutla, V. Vittal, G. T. Heydt, D. Chipman and B. Keel, "Practical evaluation of surge arrester placement for transmission line lightning protection. *IEEE Transaction on Power Delivery*, Vol. 25, n° 3, July, 2010.

[11] S. Rusck, "Induced lightning overvoltages on power transmission lines with special reference to the overvoltage protection of low voltage networks". Ph.D. dissertation, Royal Institute of Technology, Stockholm, 1957.

[12] M. Modrusan, "Tests on high-voltage metal oxide surge arresters with impulse currents", *Fourth International Symposium on High Voltage Engineering*, Athens, Greece, 1983.

[13] M. De Nigris, A. Sironi, I. Bonfanti, F. Giornelli, C. Valagussa and L. K. Shing, "Most recent developments in surge arresters technology and testing", *International Conference on Power System Technology Proceedings*, Beijing, China, 1998.

[14] R. D. Melchior, "Fault testing of gapless zinc oxide transmission line arresters under simulated field conditions", *IEEE Transaction on Power Delivery*, Vol. 10, n° 2, 2003.

[15] P. Pinceti and M. Giannettoni, "A simplified model for zinc oxide surge arresters", *IEEE Transactions on Power Delivery*, vol. 14, n° 2, pp.393-398, 1999.

[16] IEEE Working Group 3.4.11, "Modeling of metal oxide surge arresters", *IEEE Transactions on Power Delivery*, vol. 7, n° 1, pp. 302-309, 1992.

[17] E. Biasotto, "Um procedimento de análise para a repotenciação de linhas de subtransmissão de 34,5 kV para 69 kV", MSc Thesis, Scholl of Engineering of São Carlos, University of São Paulo, São Carlos, SP, 2009.

[18] NBR 5419. Proteção de Estruturas Contra Descargas Atmosféricas, 2005.

[19] ABB, "High Voltage Surge Arresters Buyer's Guide", Edition n° 9, 124p, 2012.

[20] J. R. Marti, "Accurate modeling of frequency-dependent transmission lines in electromagnetic transient simulations", IEEE Trans. on Power App. And Syst., vol. PAS-101, pp. 147--155, n° 1, 1982.

[21] W. Diesendorf, *Insulation Coordination in High Voltage Electric Power Systems*. London, Butterworth, 1974.

[22] R. A. Flauzino and I. N. da Silva, "Efficient parametric adjustment of fuzzy inference system using unconstrained optimization," *Lecture Notes in Computer Science*, vol. 4507, pp. 399–406, 2007.

[23] R. A. Flauzino and I. N. da Silva, "Tuning of Fuzzy Inference Systems Through Unconstrained Optimization Techniques," *International Journal of Computer Research*, vol. 12, pp. 243–252, 2003.

BIOGRAPHIES

Marcel Ayres de Araújo was born in Bauru, Brazil, in 1985. He graduated in Electrical Engineering at the University of São Paulo (USP), São Carlos, Brazil in 2010. Currently he is studying to obtain a M.Sc. degree at the same university. Has interest in research related to protection of power systems, fault location, artificial neural networks, smart grids, distributed generation and electric power systems.

Rogério Andrade Flauzino was born in Franca, Brazil, in 1978. He graduated in electrical engineering and received M.Sc. degree in electrical engineering from the São Paulo State University (UNESP), Brazil, in 2001 and 2004, respectively. He received Ph.D. degree in electrical engineering from the University of São Paulo (USP), Brazil, in 2007. Currently he is an Assistant Professor at the University of São Paulo. His research interests are within the field of fault identification in power distribution system, computational intelligence and nonlinear optimization. He has published more than 50 papers in congress proceedings, international journals and book chapters.

Oureste Elias Batista was born in Goiânia, Brazil, in 1988. He graduated in Electrical Engineering at the University of São Paulo (USP), São Carlos, Brazil in 2011. Currently he is studying to obtain a M.Sc. degree at the same university. His research interests are within the fields of intelligent systems, such as fuzzy inference systems and artificial neural networks, and electric power systems, such as distributed generation, fault location, smart grids and energy management.

Lucas Assis de Moraes was born in Limeira, Brazil, in 1988. He graduated in Electrical Engineering at the University of São Paulo (USP), São Carlos, Brazil in 2011. Currently he is studying to obtain a M.Sc. degree at the same university. His research interests are within the fields of intelligent systems, such as fuzzy inference systems and artificial neural networks, and electric power systems.

Development of an Approach for Choosing the Input Variables for a Forecast Fuzzy Inference System

(Special Session – ICAI'2013: Expert Systems for Intelligent Automation Purposes in Power Systems)

L. A. Moraes; R. A. Flauzino; M. A. Araújo; O. E. Batista
Department of Electrical and Computing Engineering
University of São Paulo – USP
São Carlos, Brazil
lucas.moraes@usp.br; raflauzino@sc.usp.br; marcel.araujo@usp.br; oureste.batista@usp.br

Abstract—**In this paper an approach for better choosing the inputs for a forecast fuzzy inference system is developed. The multilayer fuzzy inference system is tuned and applied to estimate the time series of active power demand in a substation feeder. Iteratively, after each estimation, the vector containing the sample errors is correlated with the time series, obtaining the next input of the system. This way, it is attempted to estimate the largest possible horizon reaching the minimum forecast error. The obtained results are satisfactory, showing that the methodology for choosing non-trivial inputs for an intelligent forecast system is capable of achieving great accuracy for different horizons.**

Index Terms—**Electricity distribution, fuzzy inference systems, intelligent systems, modeling and simulation of dynamic systems, time series forecasting.**

I. INTRODUCTION

The estimation of future values of a time series can be applied in many different areas, such as economy (stock market, inflation taxes, investment interests and exchange rates), weather (temperature, rain, snow and wind) and industry (supply chain management, product demand and raw material prices). The electric power systems are also one of the various forecast application areas. In this field there are many variables that need to be estimated, such as energy prices [1-6], demands [7-11] and loads [12-21].

There are many methods used to forecast time series, with the objective to minimize the prediction error. Some of them define that the term to be estimated depends linearly on the N past terms, such as Auto Regression (AR) and Moving Average (MA), and some of them use artificial intelligence, like Artificial Neural Networks (ANN) and Fuzzy Systems (FS) to establish non-linear dependencies. Besides, two or more of these methods of estimation can be combined to create a hybrid method, like the Auto Regressive Moving Average (ARMA). It is also possible to mix linear and intelligent methods [7], [13], [18].

Maybe due to the influence of linear forecast methods, it is common to observe in many time series forecast applications using intelligent systems the use of the N last terms of the series as inputs to forecast the actual term. But since intelligent methods can create non-linear dependencies, it is interesting to propose a different way to choose the inputs of an intelligent forecast system. This is the objective of this paper.

The developed approach for better choosing the inputs is applied at the FS developed by [22] to forecast time series power demand at the city of Ubatuba, Brazil. The accuracy of the estimation is verified with the Absolute Error Histogram (AEH) and with the Mean Square Error (MSE). It is wanted to predict the greatest horizon possible using non-trivial inputs, that is, using different inputs than the N previous terms of the time series. Since it is a short-term estimation, this paper motivation is the real-time control of the city's substation.

This paper is organized as follows: Section II describes the multilayer fuzzy inference system used. Section III explains how the FS was adjusted. Section IV exposes the methodology used to obtain the results, which are in Section V. Finally, Section VI contains the conclusions of this paper.

II. MULTILAYER FUZZY INFERENCE SYSTEM

The fuzzy inference systems may be treated as systems that use the concepts and operations defined by the fuzzy set theory and by the fuzzy reasoning methods, since they use the fuzzy inference process to perform their operational functions. Basically, these operational functions include the inputs fuzzification of the system, the inference rules associated to it, the aggregation of rules and the later defuzzification of the aggregation results, which represent the outputs of the FS [23].

Considering the operational functions performed by the fuzzy inference systems, it is convenient to represent them by a three layers model. Thus, a fuzzy inference system may be given by the sequential composition of an input layer, an inference layer and an output layer.

A. Input Layer

The system inputs fuzzification has the purpose of determining the membership degree of each input related to the fuzzy sets associated to each input variable. To each input variable, as many fuzzy sets as necessary can be associated. This way, given a FS with only one input, to which there are N fuzzy sets defining it associated, then the output of the input layer is a column vector with N elements, which are representing the membership degrees of this input in relation to those fuzzy sets.

If we define the input of this FS with one only input x, then the input layer output of the FS is the vector I_1, that is

$$I_1 = \left(\mu_{A_1}(x) \quad \mu_{A_2}(x) \quad \cdots \quad \mu_{A_N}(x) \right)^T, \quad (1)$$

where $\mu_{A_k}(\cdot)$ is the membership function defined to x input, which is referring to the k-th fuzzy set associated to this input.

The generalization of the input layer concept for a FS having p input variables can be achieved if we consider each input of this FS being modeled as a sub-layer of the input layer. Taking into account this consideration, the output vector of the input layer $I(x)$ is then defined by

$$I(x) = \left(I_1(x_1)^T \quad I_2(x_2)^T \quad \cdots \quad I_p(x_p)^T \right)^T, \quad (2)$$

where x_i is the i-th input of the FS and $I_k(\cdot)$ is the k-th vector of membership functions associated to the x_k input.

B. Inference Layer

The set of rules has fundamental importance to the correct functioning of the fuzzy inference system. There are several methods for the extraction of fuzzy rules from the tuning set.

In this paper, the FS has initially all the possible inferred rules. Therefore, the tuning algorithm has the task of weighting the inference rules. The weighting of the inference rules is an adequate way to represent the most important rules in the FS, or even to allow that conflicting rules are related to each other without any verbal completeness loss. Thus, it is possible to express the i-th fuzzy rule as

$$R_i\big(I(x)\big) = w_i r_i\big(I(x)\big), \quad (3)$$

where $R_i(\cdot)$ is the function representing the fuzzy weight value of the i-th fuzzy rule, w_i is its weight factor and $r_i(\cdot)$ represents its fuzzy value.

C. Output Layer

The output layer of the fuzzy inference system aims to aggregate the inference rules, as well as the defuzzification of the fuzzy set generated by the aggregation of inference rules.

In the fuzzy inference systems design, the choice of not only the aggregation method but also the defuzzification method constitutes a very important decision. The aggregation method of the fuzzy inference rules must be in such a way that the fuzzy set resulting from aggregation is capable of adequately representing the knowledge contained in this set of fuzzy rules. By analogy, the method chosen for the defuzzification must be capable of expressing, in a crisp value, the fuzzy set resulting from the fuzzy aggregation.

III.   ADJUSTMENT OF THE FUZZY INFERENCE SYSTEM

The formalization of a FS in the form of a multilayer system can be justified not only by the different operational division of each one of these layers, but also by the presence in each of them of different free parameters.

This way, the mapping f between the input space x and the output space y may be defined by

$$y = f(x, mf_{In}, w, mf_{Out}), \quad (4)$$

where mf_{In}, w and mf_{Out} respectively represent the vectors of the input membership functions parameters, the weight of the inference rules and the output membership functions parameters. Therefore, mf_{In}, w and mf_{Out} represent the free parameters of the FS and for this reason it is more suitable to rewrite (4) as presented in

$$y = f(x, \Theta), \quad (5)$$

where Θ is the vector resulting from concatenation of the free parameters involved to system, that is

$$\Theta = \left(mf_{In}{}^T \quad w^T \quad mf_{Out}{}^T \right). \quad (6)$$

The definition of the energy function to be minimized remains in function of the fuzzy mapping. Considering that the tuning set $\{x, y\}$ is fixed during the whole adjustment process, it may be written as

$$\xi_{(x,y)} = \xi_{(x,y)}(\Theta), \quad (7)$$

where ξ represents the energy function associated to the fuzzy inference system f.

In problems like this, involving the minimization of energy functions, it is desired that, after any iteration, the energy function value is lower than that value obtained in the previous iteration. There are several techniques used to solve unconstrained optimization problems. A detailed description of the unconstrained optimization techniques may be found in [24]. The choice of the most adequate technique to be used is conditioned to the form by which the energy function is defined. For example, the Gauss-Newton method for the unconstrained optimization may be more applicable in problems where the energy function is defined as

$$\xi(\Theta) = \frac{1}{2} \sum_{i=1}^{m} e^2(i), \quad (8)$$

where $e(i)$ is the absolute error in relation to the i-th tuning pattern.

In this paper, a derivation of the Gauss-Newton method is used for the fuzzy inference system tuning. The Gauss-Newton expression to update the vector Θ is defined by

$$\Theta_{next} = \Theta_{now} - \frac{1}{2}(J^T J)^{-1} g, \quad (9)$$

where g is the gradient of ξ expressed in (7) and J is the Jacobian matrix of e presented in (8). The optimization algorithm used was the Levenberg-Marquardt method [25]. The Levenberg-Marquardt method can handle well ill-conditioned matrices $J^T J$ by altering (9) to

$$\Theta_{next} = \Theta_{now} - \frac{1}{2}(J^T J + \lambda I)^{-1} g. \quad (10)$$

The calculation of the matrices J and the vectors g were performed through the finite differences method.

IV. METHODOLOGY

The first step of the methodology is to choose as the first input a known variable that behaves similarly to the unknown variable to be predicted. This grants great forecast accuracy. To do so, it is calculated the autocorrelation of the time series. The sample with the greatest autocorrelation value along the time series data array is chosen as the first input.

The second variable to compose the set of input variables is the one that best correlates with the vector normal to the projection of the demand-to-be-estimated vector in the vector corresponding to the first input. Proceeding like this, the two first inputs have different behaviors, collaborating with the estimation precision. This is better than using two look-alike inputs (for example, the two with the greatest autocorrelation value), what would give a worse prediction with the same computational effort.

The calculation of this second input variable is done by

$$d_n(t) = d(t) - \frac{\langle d(t)|d(t-m)\rangle}{\|d(t-m)\|^2} d(t-m), \qquad (11)$$

where $d_n(t)$ is the normal vector, $d(t)$ is the demand projection, $d(t-m)$ is the first input, m is the number of sample lags between the first input and the value to be estimated, and $\langle \cdot | \cdot \rangle$ is the scalar product between the arguments.

After choosing the first two inputs, a forecast is made. The forecast error, then, is correlated with the past data. The array position (excluding the already chosen ones) which shows the greatest correlation with the prediction error is chosen as the next input, and another forecast is made. This process is repeated as many times as necessary, always increasing one input per iteration, aiming to reduce the estimation error.

It is worth to remark that during all the input choosing process, no other parameter of the FS is altered. The only modification between two consecutives iterations is the number of inputs, which is increased by one.

V. RESULTS

The power demand data used in this paper are shown in Fig. 1. The values were measured every five minutes for 210 days. The peak value was registered at December 31st.

In a day, the average power demand is that one showed by the blue curve in Fig. 2. The green curve at this same figure shows the 90% percentile, what means that at 90% of the days the demand is below the green curve.

To choose the first input variable it was plot the autocorrelation graph for a day, shown in Fig. 3. The greatest autocorrelation value (0.9927) is the one given by the sample delayed five minutes (or one lag) from the sample to be predicted.
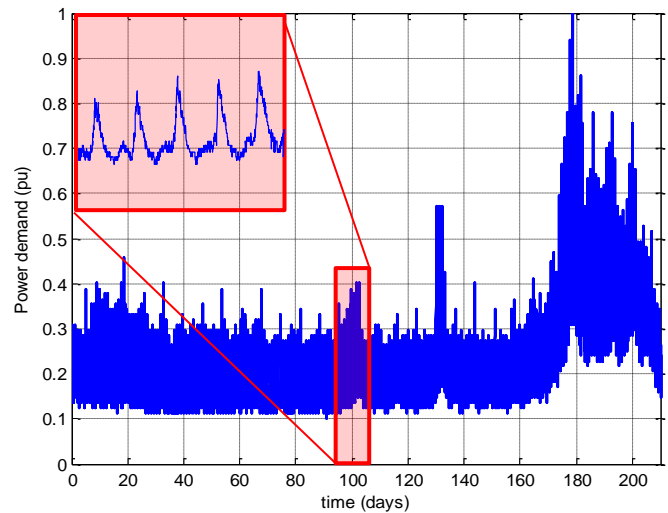
Figure 1.   Power demand time series.
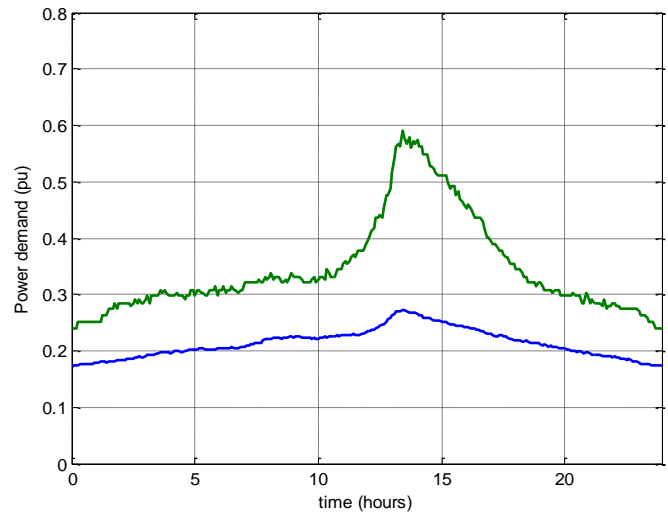
Figure 2.   Power demand daily average (blue) and 90% percentile (green).
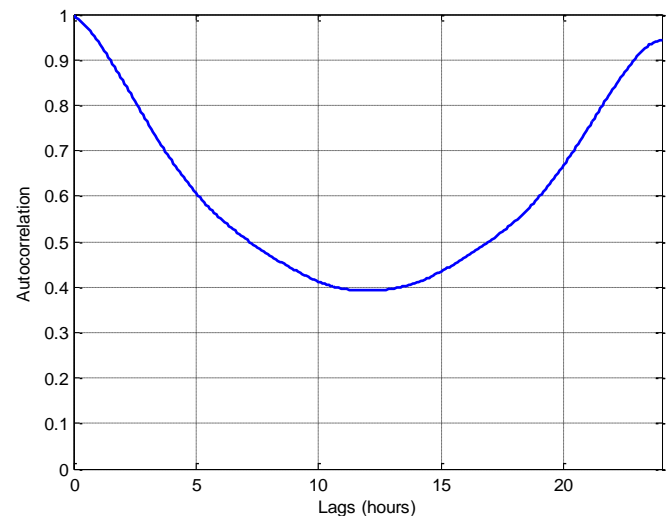
Figure 3.   Time series autocorrelation over a day.

The second input was chosen by analyzing the correlation between the vector calculated in (11), with $m = 1$, and the time series. The best result indicated that the sample with 30 lags should be this input.

Having the two first inputs, $(t - 1)$ and $(t - 30)$, a forecast was made, trying to estimate the sample (t). The training MSE is shown in Fig. 4 and the AEH is shown in Fig. 5. It is possible to notice that the training MSE ended up in a small magnitude (of 10^{-4}), and the error histogram remember the appearance of a normal distribution with average zero, meaning that the forecast was good.
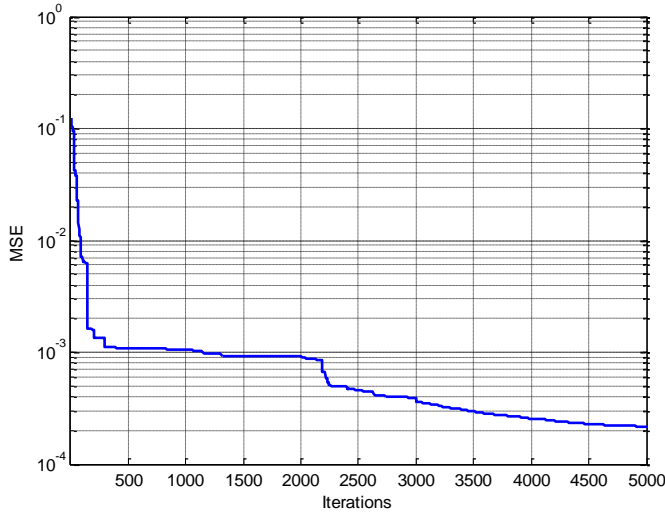
Figure 4.   Training MSE for the first forecast.
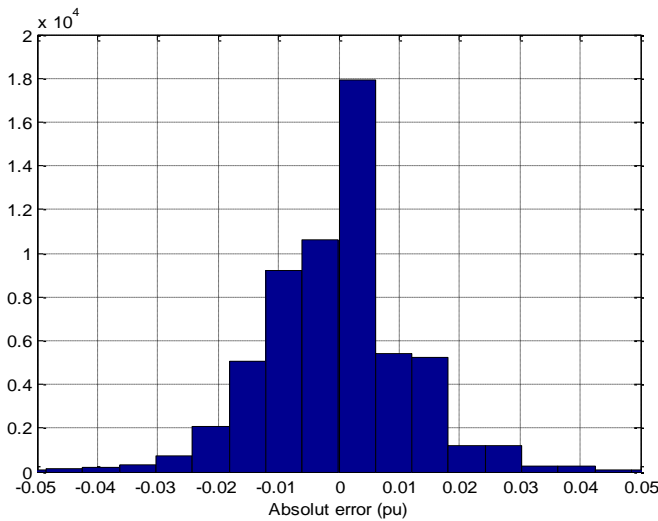
Figure 5.   First forecast absolute error histogram.

Following the proposed methodology, it was calculated the correlation between the estimation error and the time series. This correlation is shown in Fig. 6. Analyzing it, the variable with the highest correlation value is the one that is 1h20min (16 samples) delayed in relation with present time. So, using the FS with the same parameters, and including this new input variable, a second forecast was made, now with three input variables: $(t - 1)$, $(t - 30)$ and $(t - 16)$. The training MSE for this forecast is shown in Fig. 7, and the AEH in Fig. 8.
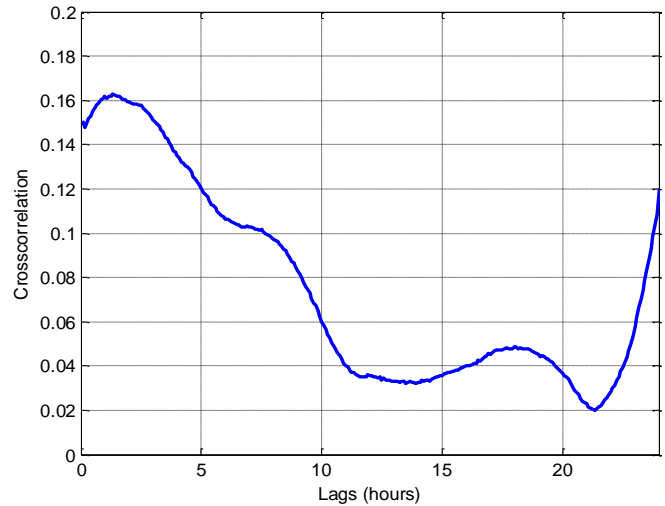
Figure 6.   Correlation between first estimation error and time series.
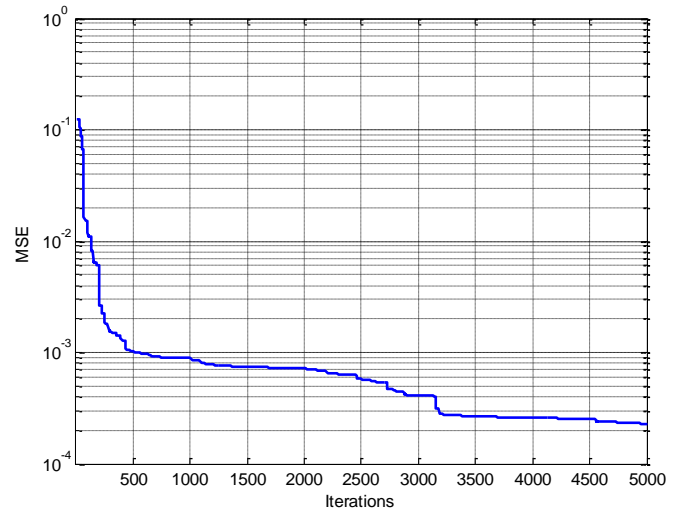
Figure 7.   Training MSE for the second forecast.
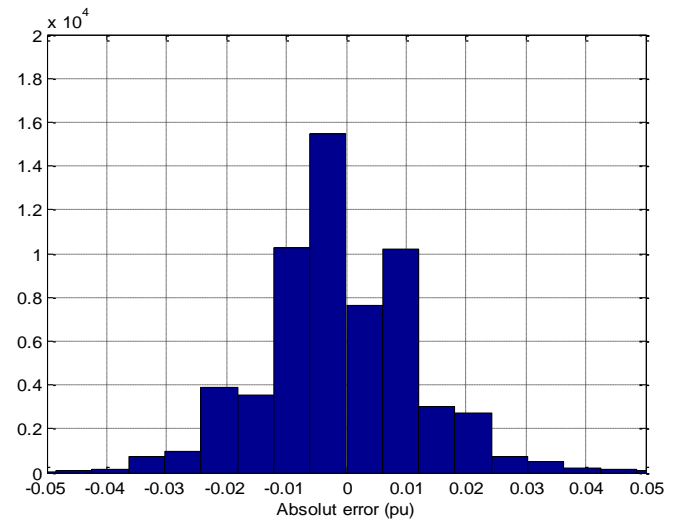
Figure 8.   Second forecast absolute error histogram.

It is possible to notice that the results of the first prediction are very similar to the results of the second one, that is, the majority of the absolute error is below 0.01. That happened because both of them tried to forecast the sample (t), which is a horizon of only one sample. But for a good real-time control of the substation, it is needed to forecast a longer horizon. So it is convenient to compare the two forecasts performed in many different horizons. The result is shown in Fig. 9. The blue curve represents the MSE for the forecast with two inputs and the green one the MSE for the forecast with three inputs.
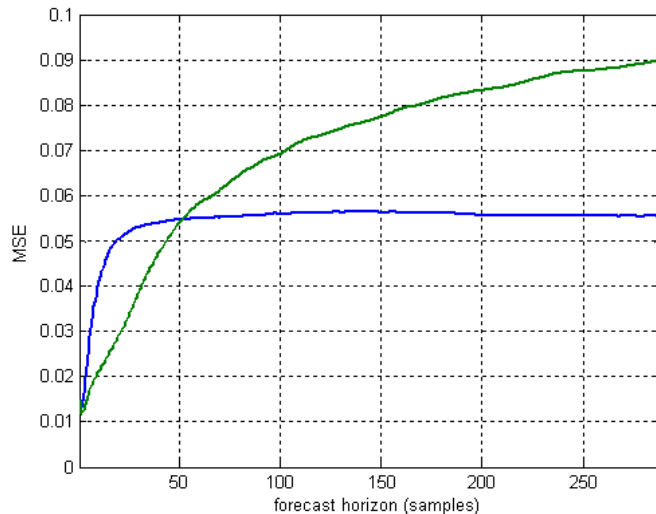
Figure 9.   MSE comparison between the first (blue) and the second (green) forecast.

It is clear that for horizons longer than 52 samples, the forecast with two inputs is better, but on the other hand, for horizons shorter than 52 samples, the forecast with three inputs gives better results. This is an unexpected result, since it was imagined that more inputs should result a more accurate forecast for any horizon.

It is possible to affirm that the forecast error with fewer inputs grows faster but stabilizes in a lower level, while the forecast error with more inputs grows slower but stabilizes in a higher level. This is an important result, since it is crucial to define the number of inputs and the forecast horizon, aiming to minimize the forecast error.

VI.   CONCLUSIONS

This paper has achieved its objective of proposing a new and efficient methodology for forecasting with accuracy a time series using a small number of inputs. However, an unexpected result showed that more inputs doesn't necessarily mean more accuracy. The longer is the forecast horizon fewer inputs are needed to keep prediction errors acceptable.

The methodology presented in this paper can be applied in any time series forecasting fields. However, to do so, it is extremely necessary to analyze carefully the horizon to be predicted, so the correct number of inputs can be chosen. Once defined the forecast horizon, it is convenient to analyze the forecast error constantly. If after increasing one input the prediction gets worse, this means that the previous configuration is the best one for that case of study.

REFERENCES

[1] S. C. Chan, K. M. Tsui, H. C. Wu, Y. Hou, Y.-C. Wu e F. F. Wu, "Load Price Forecasting and Managing Demand Response for Smart Grids - Methodologies and Challenges," *IEEE Signal Processing Magazine,* vol. 9, pp. 68-85, 2012.

[2] F. Lira, C. Muñoz, F. Nuñez e A. Cipriano, "Short-term forecasting of electricity prices in the Colombian electricity market," *IET Generation, Transmission & Distribution,* vol. 3, pp. 980-986, 2009.

[3] A. Motamedi, H. Zareipour e W. D. Rosehart, "Electricity Market Price Forecasting in a Price-responsive Smart Grid Environment," em *Power and Energy Society General Meeting*, 2010.

[4] A. Motamedi, H. Zareipour e W. D. Rosehart, "Electricity Price and Demand Forecasting in Smart Grids," *IEEE Transactions on Smart Grid,* vol. 3, n. 2, pp. 664-674, 2012.

[5] A. Petrella e S. Sapio, "A time series analysis of day-ahead prices on the Italian power exchange," em *International Conference on the European Energy Market*, 2009.

[6] H. Zhou, X. H. Wu e X. G. Li, "An ANFIS model of Electricity Price Forecasting Based on Subtractive Clustering," em *Power and Energy Society General Meeting*, San Diego, 2011.

[7] N. Amjady e A. Daraeepour, "Midterm Demand Prediction of Electrical Power Systems Using a New Hybrid Forecast Technique," *IEEE Transactinos on Power Systems,* vol. 26, n. 2, pp. 755-765, May 2011.

[8] J. Deng, "Energy Demand Estimation of China Using Artificial Neural Network," em *Third International Conference on Business Intelligence and Financial Engineering*, 2010.

[9] W. Jin-ming e L. Xin-heng, "The Forecast of Energy Demand on Artificial Neural Network," em *International Conference on Artificial Intelligence and Computational Intelligence*, 2009.

[10] W. Schellong e F. Hentges, "Energy Demand Forecast for a Cogeneration System," 2011.

[11] W. Xiao-jia, W. Hai-jiang e H. Li-qiang, "Electricity Demand Forecasting Based on Threepoint Gaussian Quadrature and Its Application in Smart Grid," em *6th Conference on Wireless Communications Networking and Mobile Computing*, 2010.

[12] M. B. Abdul Hamid e T. K. Abdul Rahman, "Short Term Load Forecast Using an Artificial Neural Network Trained by Artificial Immune System Learning Algorithm," em *International Conference on Computer Modelling and Simulation*, 2010.

[13] A. A. E. Desouky e M. M. E. Kateb, "Hybrid adaptative techniques for electric-load forecast using ANN and ARIMA," *IEE Procedings - Generation, Transmission & Distribution,* vol. 147, pp. 213-217, 2000.

[14] O. E. Dragomir, F. Dragomir, R. Gouriveau e E. Minca, "Medium Term Load Forecasting Using ANFIS Predictor," em *18th Mediterranean Conference on Control & Automation*, Marrakech, Morocco, 2010.

[15] Y.-D. Gu, J.-Z. Cheng e Z.-Y. Wang, "An Fuzzy Forecasting Algorithm for Short Term Electricity Loads Based on Partial Clustering," em *International Conference on Machine Learning and Cybernetics*, Guilin, 2011.

[16] M. B. A. Hamid e T. K. A. Rahman, "Short Term Load Forecasting Using an Artificial Neural Network Trained by Artificial Immune System Learning Algorithm," em *12th International Conference on Computer Modelling and Simulation*, 2010.

[17] H.-T. Yang e C.-M. Huang, "A New Short-Term Load Forecasting Approach Using Self-organizing Fuzzy ARMAX Models," *IEEE Transactions on Power Systems,* vol. 13, n. 1, pp. 217-225, 1998.

[18] H.-T. Yang, C.-M. Huang e C.-L. Huang, "Identification of ARMAX Model for Short Term Load Forecasting: An Evolutionary Programming Approach," *IEEE Transactions on Power Systems,* vol.

11, n. 1, pp. 403-408, 1996.

[19] J. Yasuoka, J. L. P. Brittes, H. P. Schmidt e J. A. Jardini, "Artificial Neural Network-Based Distribution Substation and Feeder Load Forecast," em *CIRED2001*, 2001.

[20] W. Yuill, R. Kgokong, S. Chowdhury e S. P. Chowdhury, "Application of Adaptive Neuro Fuzzy Inference System (ANFIS) based Short Term Load Forecasting in South African Power Networks," em *45th Universities' Power Engineering Conference*, Cardiff, Wales, 2010.

[21] H.-t. Zhang, F.-y. Xu e L. Zhou, "Artificial Neural Network for Load Forecasting in Smart Grid," em *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics*, Qingdao, 2010.

[22] R. A. Flauzino, "Identificação e Localização de Faltas de Alta Impedância em Sistemas de Distribuição Baseadas em Decomposição por Componentes Ortogonais e Inferência Fuzzy," Tese de Doutorado, São Carlos, 2007.

[23] J. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 23, pp. 665-685, 1993.

[24] D. P. Bertsekas, Nonlinear Programming, Athenas Cientific, 1995.

[25] D. Marquardt, "An Algorithm for Least Squares Estimation of Nonlinear Parameters," *Journal of the Society for Industrial and Applied Mathematics,* pp. 431-441, 1963.

Applying a Fuzzy System for Robust Fault Location in a Distribution System with Distributed Generation

O. E. Batista, R. A. Flauzino, L. A. Moraes, M. A. Araujo and I. N. da Silva
Department of Electrical and Computer Engineering
University of São Paulo, USP
São Carlos, Brazil
oureste.batista@usp.br; raflauzino@sc.usp.br; lucas.moraes@usp.br; marcel.araujo@usp.br

Abstract—**Although Distributed Generation (DG) is a sustainable alternative to meet the growing demand for electricity, the intermittency of their primary sources and variability in its operation introduces additional uncertainties to the Power Distribution Systems (PDS). Because of its great potential for penetration in the PDS, it is important to investigate their impact to the grid's safety and reliability. In contribution to such investigations, this article conducts a study of the variability in fault currents and voltages measured in the substation. For this, computer simulations of a system with DG were developed based on the IEEE 13 Node Test Feeder. The analysis was structured in the calculation of the apparent impedance of the faulty phase, for which there was variability greater than 30% in the module and that 15% angle. Due to this variability was proposed a fuzzy system for robust fault location in distribution feeders with DG, whose accuracy error was less than 12 m.**

Index Terms— **Distributed power generation, Fault location, Power distribution, Robustness analysis, Fuzzy systems.**

I. INTRODUCTION

Together with the increased use of renewable energy, distributed generation (DG) has shown great prominence in discussions about the future of electric power distribution systems, being associated with concepts such as smart grids and microgrids [1], [2]. Since the benefits of DG boost its consolidation, its impacts on the electric system have taken the attention of many researchers, regulatory agencies, electricity distributors and sponsors for technical feasibility studies [3]–[6]. Given that conflicts are highlighted in protection systems, as in [7], the high penetration of DG makes essential the development of studies on its impact on the safety and reliability of the system, as well as solutions and alternatives to mitigate them.

The presence of DG is responsible for loss of radial characteristic of Power Distribution Systems (PDS), changing the power flow in your direction and magnitude [8]. Under this new configuration, deterministic methods for fault location will have its effectiveness hampered due to the increased complexity of the system and the introduction of significant uncertainties that come from the variability and

intermittency of power supply of the DG unit [9]–[13]. The maneuvers of connection and disconnection of generators at multiple points and at undefined times result in significant and random variations in the grid configuration and in its electrical parameters, requiring strategies robust enough to interpret different situations as a result of the same type of operation or failure [14].

This situation leads to a new paradigm for determining the performance of solutions in PDS. The methods and techniques must, in addition to criteria for precision and accuracy, provide greater robustness against inputs with considerable uncertainty level [15], [16]. In this sense, the purpose of the study is to evaluate the degree of variability in the behavior of a faulted PDS, in a scenario with diverse situations of connection and disconnection of distributed generators. These situations were evaluated by means of computer simulations, in which it was registered the voltage and current fault phasors measured at the substation. With these measures, a system was developed for fault location based on fuzzy logic to deal with uncertainty through a possibilistic approach.

II. DISTRIBUTED GENERATION IMPACTS

A. The High Level Penetration

Motivated by technical, economic and environmental reasons, the DG has emerged as an option for the growing electricity demand and in response to technological advances, sensitive loads and the creation of new energy policies increased [2]. The DG is commonly associated with renewable sources and efforts to reduce the environmental impacts will boost its penetration in PDS [5]. This penetration will cause changes in the behavior of the distribution grid, taking it from a passive, one-way flow characteristic of the substation to the consumers, to a more active, with bidirectional flow due to branches that consume, generate and store energy [17].

Due to the increase in complexity of the PDS with DG, one of the main objectives of the concept of smart grids is to provide the necessary infrastructure for optimized integration. Studies on this integration describe a series of benefits to the grid, but also assess the challenges. These benefits involve aspects related to the reliability, robustness, economy and

sustainability, highlighting the improvement in voltage profile, reducing losses in transmission and distribution, increasing reliability, using renewable sources, reducing the environmental impact and supporting peaks. The main challenges involve investment projects, grid changes and its new behavior, such as high initial cost, islanding issues, loss of radiality, coordination and selectivity problems, increase levels of short-circuit, voltage fluctuations and interharmonics introduction [5], [6].

B. The Uncertainties Increase

In the PDS, the spread branch, the electric unbalance, the dynamics of loads, the voltage and current levels and other peculiarities give them an intrinsic level of uncertainty. At this level of uncertainty it is still added up operational variations and disturbances, such as grid reconfiguration, failure of equipment, errors in modeling and simulation, noise measurement and climate change [18]. Although DG's objective is to promote increase in the robustness and reliability of the system, their penetration introduces uncertainties, which added to the already existing, make deterministic approach inadequate to model the behavior of the system, analyze its performance and make decisions with the permissible margin of error. In this scenario, there are two approaches capable of incorporating uncertainties [9], [19].

- Probabilistic: variables represented by probability density functions (mean and variance), exploiting the stochastic behavior. Statistical tools are used.

- Possibilistic: variables represented by their range of variation and evaluated on several possible scenarios. Interval mathematics and fuzzy logic concepts are used.

When DG is based on renewable energy such as wind, solar and tidal power, there is a big challenge: the variability in the generation, since it relies on primary sources with intermittent behavior, variant depending on geographical location and difficult to predict [4], [20], [21]. In contrast to the objectives of the GD, the variability induce power fluctuations throughout the day which can cause reduction in the robustness of the electrical system [3]. There is also the dependence of generation of a DG unit with the demand and generation of other units on the same feeder, because exceeding the ability of consumption there will be reverse flow of reactive power, from the consumer to the substation, and the losses will increase [9]. Whatever the approach used, the fact is that the uncertainties resulting from the penetration of DG at a high level is relevant and will reduce significantly the effectiveness of deterministic planning techniques, requiring that different scenarios are taken into consideration [10].

C. Fault Location

The fault location is a task that has a direct impact on the level of security and economy of operation of Electrical Power Systems (EPS). Its efficiency is associated with speed and accuracy in the estimation of fault point. Traditional methods for fault location are commonly grouped into two categories, based on impedance and based on traveling waves. The first one uses measurements of fault voltages and currents in calculations of electrical circuits. The second relies on the time interval between an initial wave and its reflections on the fault point. Both have their limitations and compromise between robustness and accuracy is the major problem of these methods [16].

The changes in power flow and the direction and magnitude of fault currents compose the main impact of DG penetration in the PDS, the loss of the radial characteristic under which protection systems were projected. In addition to problems of coordination and selectivity, highlighted in [7], for systems with DG, traditional methods exhibit significant errors, which increase in direct proportion to the fault impedance and power generated downstream of the fault point [8], [22], [23]. The different types of DG technology, such as natural gas turbines, steam turbines (fossil fuel or biomass), internal combustion engines (diesel), fuel cells, microturbines, small hydropower plants (SHP), photovoltaic cells and wind generation, have particular electrical characteristics. This fact associated with the type of grid connection, such as the use of frequency inverters, contributes to the existence of different forms of energy fluctuations, uncertainties in the operation and faults response [12]. These aspects require more robustness to the fault location, at a level compatible to the uncertainties present in the input variables [16].

III. Robustness Analysis Proposal

The robustness concept is often used in control theory, representing the ability of a system to remain stable in the presence of disturbances and uncertainties. In a robust system, the effect of varying the inputs in the accuracy of the output is minimized (Fig. 1). A robust method for locating faults should be applicable for variations of fault types, fault impedance, saturation of current transformers (CTs), system configuration, etc. [16]. Accuracy is the major reference for measuring the performance of a system and validating its application. The uncertainties, however, affect accuracy in a way that scenarios uncertain to robustness should receive equal attention [15].

The study proposed in this paper approaches the uncertainty in the operations of connecting and disconnecting DG units to the grid, showing the degree of variability in the measurements of fault voltages and currents seen by the substation, depending on various combinations of entry and exit of generators in the feeder. This study was motivated by the growing discussion of the uncertainties in the operation of DG units, in addition to those from the stochastic behavior of primary sources.
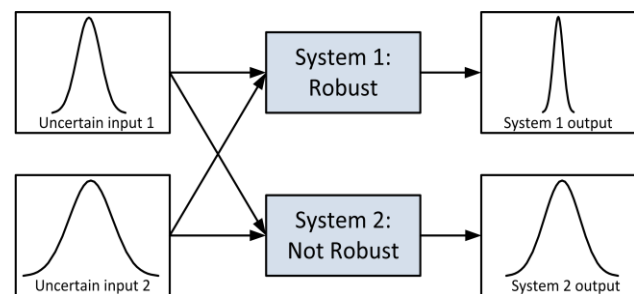
Figure 1.   Robustness illustration.

Reference [24] describes that the sources of uncertainty in the operation of DG units include the number of units connected to the grid at a certain time, its location and the power imported to the system. In [14] it is stated that different consumers may have different strategies to operate his generator and the processes of connection and disconnection to the grid can be random. Their results show that randomness in operation causes increased losses and reduces the power transfer capacity. Regarding the treatment of uncertainties, [11] highlights the possibilistic approach, given that not all uncertainties are necessarily stochastic, because some variables depend on investment decisions or operation of the owner or the operator, defining a variability based in certain rules, what can be better handled by fuzzy inference systems.

A. Power Distribution System Configuration

To evaluate the variability in the response to faults, it was modeled a distribution system based on IEEE 13 Node Test Feeder [25], using the toolbox SimPowerSystems™ and the Simulink® graphical environment present in the software MATLAB®. The original system was adapted to the integration of DG, as illustrated in Fig. 2.

The voltage regulator 650-632 of the branch was removed because the DG already promotes improvement in the voltage profile. The switch of the branch 671-692 was kept closed in all simulations. Representing the penetration of DG, six synchronous generators were connected to the consumption points 611, 634, 646, 652, 675 and 680. The generators were represented by their subtransient model (Fig. 3), with 0.03 pu armature resistance (r_a), 0.24 pu synchronous subtransient reactance (x_s''), configured in terminal voltage control mode (1 pu) and active power set at 425 kW, that is, the bus at its terminals is of the type PV.
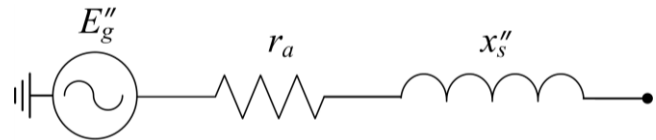
Figure 3. Synchronous generator subtransient equivalent.

The three-phase system in question has an input voltage of 115 kV and a transformer with power of 5 MVA reduces the voltage distribution in 4.16 kV. The feeder has 1.55 km in its largest branch (650-652) and approximately 944 m on the laterals. The consumer units are served with a voltage of 4.18 kV, and only one of these, connected to node 634, utilizes a distribution transformer with power of 500 kVA that reduces the voltage to 480 V. In the feeder, 3,466 kW and 2,102 kvar are consumed, which means a loading of almost 75% in the substation transformer, considering the reactive compensation of 700 kvar. The loads were equally distributed among the three phases to minimize the unbalance of power. The feeder was represented by the line model with distributed parameters, in order to obtain a more faithful response of the transient behavior and electrical coupling between phases. Each DG unit was connected to the grid through transformers equal to that present in the branch 633-634.

B. Simulations and Analysis

The simulated cases involved all combinations of states of connection and disconnection of the six generators, which includes the case without DG, totaling 64 situations for each point of fault (distance from the substation). The faults were applied at 11 points along the main branch (650-652), covering the whole feeder. The system was evaluated under single-line-to-ground (SLG) type faults, because of their high rate of occurrence, and three-phase faults, due to their severity, contemplating a symmetrical scenario and another asymmetrical one [26].

The variety of cases represents the uncertainty associated with the operation of the DG units, generating changes in the conformation of the system. Traditional methods for fault location, as in [27], show a better performance for low impedance faults (LIF), so this feature was chosen so that the results represent the variability in a more optimistic degree. In each simulation the voltage and current phasors per phase were recorded after fault. To examine the variability of measurements in terms of robustness, the transfer function defined as:

$$H_k = \frac{\dot{V}_k}{\dot{I}_k} = \frac{|\dot{V}_k|\angle\varphi_{\dot{V}_k}}{|\dot{I}_k|\angle\varphi_{\dot{I}_k}} = |Z_k|\angle(\varphi_{\dot{V}_k} - \varphi_{\dot{I}_k}), \quad (1)$$

was evaluated, where k is the faulty phase (a, b or c); V is the voltage; I is the current; φ is the phase angle and Z represents the apparent impedance, which includes the mutual impedances. This relationship combines the behavior of voltages and currents, and highlights the dynamic behavior of the grid. The quantification of variability (VAR) can be expressed in terms of percentage ratio between the range
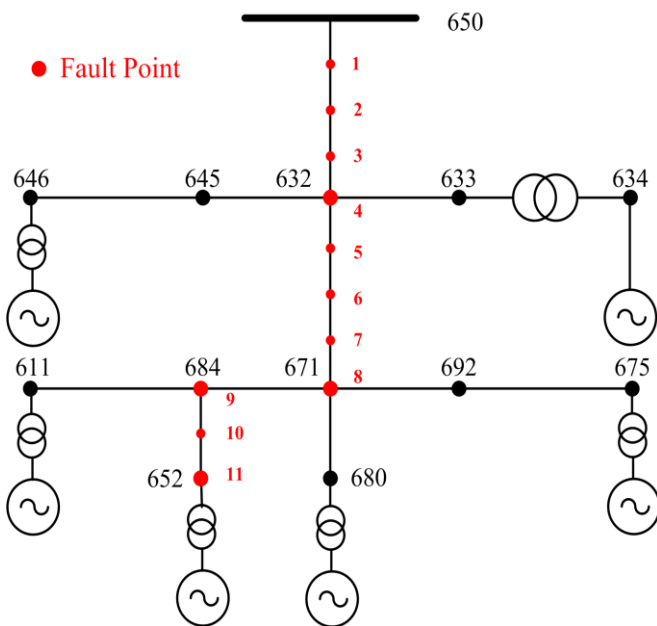
Figure 2. Adapted IEEE 13 node test feeder.

given by the extremes of the variation and the value for the case without DG, defined in (2) for the module (|Z|) and in (3) for the angle (φ).

$$VAR_{|Z|} = \frac{\max(|Z|) - \min(|Z|)}{|Z|_{\text{withoutDG}}} \qquad (2)$$

$$VAR_{\varphi} = \frac{\max(\varphi) - \min(\varphi)}{\varphi_{\text{withoutDG}}} \qquad (3)$$

IV.   RESULTS AND DISCUSSIONS

The analysis showed that the symmetry of the three-phase short circuits is capable of significantly reducing the variation in the apparent impedance in relation with SLG short circuits. The variability in the faulty phase to three-phase faults was more than 17 times lower, with a maximum value of 1.3%, lower than the minimum value for SLG faults. For these reasons, this paper will focus on the behavior of the system under the occurrence of SLG faults. The variation in apparent impedance defined in (1), for the faulty phase ("a"), as a function of fault point can be visualized in Fig. 4 for the module and in Fig. 5 for angle.
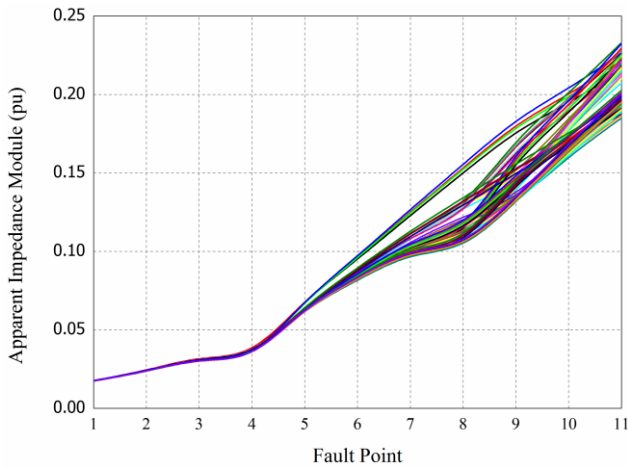
Figure 4.   Phase "*a*" apparent impedance module variation.
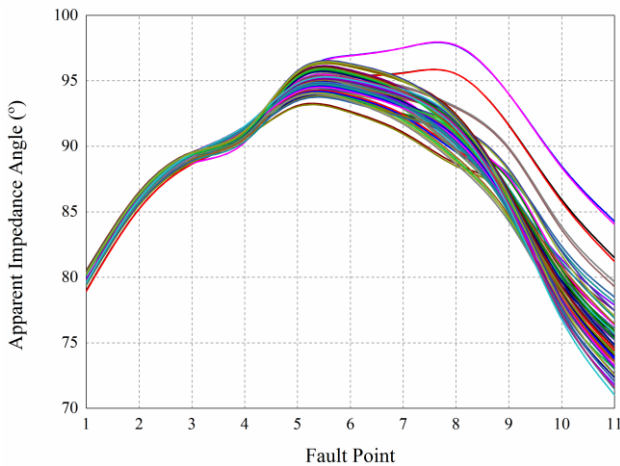
Figure 5.   Phase "*a*" apparent impedance angle variation.

Figures 4 and 5 show the variation arising from all 64 combinations of DG units operation and the evolution of the system's behavior under faults throughout the feeder. At the beginning of the feeder, faults occur much close to generators and so the entry and exit of these do not cause large variations. As the fault point moves away from the substation and approaches the generator, the variability in the operation of these influences significantly the measurements of voltage and current, since the resulting variability is proportional to the amount of generators near the point and the degree of proximity. Besides the faulty phase, variability can be observed in the apparent impedance of the other phases. Table I contains the values of the metrics defined in (2) and (3) to quantify the variability considering SLG faults in phase "a".

TABLE I.   APPARENT IMPEDANCE VARIABILITY

Fault Point	Distance (m)	Variability (%)					
		$|Z_a|$	$|Z_b|$	$|Z_c|$	φ_{Za}	φ_{Zb}	φ_{Zc}
1	152	1.9	75.4	83.6	2.0	703.9	120.0
2	305	3.3	79.7	85.8	1.5	669.6	107.7
3	457	5.0	83.3	86.3	1.1	641.2	103.0
4	610	6.9	86.1	86.6	1.4	621.9	102.0
5	762	9.1	84.4	88.5	2.9	603.7	82.5
6	914	16.4	85.9	91.0	4.6	602.4	86.5
7	1,067	24.5	89.3	94.3	6.9	604.4	93.1
8	1,219	33.4	93.4	97.8	9.7	606.9	101.3
9	1,311	29.1	90.5	95.5	10.6	613.5	97.7
10	1,433	22.8	85.7	91.1	13.6	623.3	101.1
11	1,554	22.3	82.2	87.2	16.3	630.3	102.0

As expressed in table I, the apparent impedance variability reached 33% for the module and 16% for the angle, in the faulty phase. Healthy phases variability is even higher and this phenomenon is associated with the magnetic coupling between the phases. For the three-phase system under study it follows that

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} Z_{aa} & Z_{ab} & Z_{ac} \\ Z_{ba} & Z_{bb} & Z_{bc} \\ Z_{ca} & Z_{cb} & Z_{cc} \end{pmatrix} \cdot \begin{pmatrix} I_a \\ I_b \\ I_c \end{pmatrix} \qquad (4)$$

and dividing each row by its phase current is obtained the apparent impedance defined in (1), now in terms of the mutual and the proper impedances:

$$Z_k = Z_{kk} + \frac{I_i}{I_k} Z_{ik} + \frac{I_j}{I_k} Z_{jk}, \qquad (5)$$

where k is the phase in question; i and j the other phases; Z_{kk} the proper impedance of phase k; and Z_{ik} and Z_{jk} the mutual impedance from phase i to phase k and from phase j to phase k. To the faulty phase, while $I_k \gg (I_i$ and $I_j)$, Z_k reflects the

proper impedance of the phase, what happens with faults near to the substation, because I_k is high. Along the feeder I_k presents descending magnitude, causing Z_k suffer gradual influence of other phases. For the healthy phases $I_k << (I_i$ or $I_j)$, where i or k is the faulty phase, close to the substation and along the feeder the ratio of currents continues to promote the strong influence of the faulty phase, propagating this variability in the other phases through the mutual impedance.

The variability shown in Table I is indicative of a robust system for fault location should give greater attention to the values of voltage and current in faulty phase, since healthy phases there is great variation. The post-fault data contain much information about the fault position , but say little about the state of connection of generators to the grid ($S_i = ON$ or OFF) and without communication infrastructure (smart grid) the state vector (S_1, S_2, S_3, S_4, S_5, S_6) is unknown. This information can be extracted from the data pre-fault, since the load reduces the substation with the entry of the generators. In this way, the intelligent system for robust fault location will have the conformation shown in Fig. 6, whose inputs are the module and phase of pre and pos-fault apparent impedances, composing two pairs of input. The output is the estimation of distance from the substation to fault point, in meters.
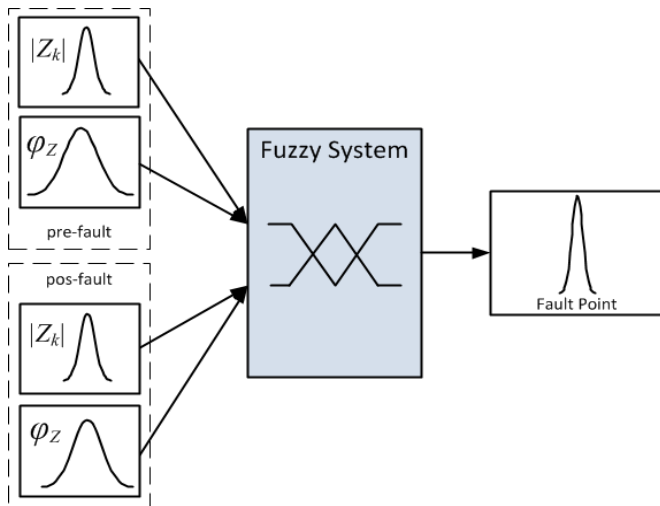
Figure 6.   Robust fault location system.

With the database formed by the simulations was possible to construct a multilayer fuzzy inference system (Fig. 7), from the automatic structural and parametric adjustment presented in [28] and [29]. The proposed tuning algorithm is based on the Mandani architecture, in which consequential members of fuzzy rules are represented by linguistic variables. The use of this type of fuzzy inference system, due to its structure, it provides robustness, offers a high semantic level and a good generalization capacity. The great advantage of this approach is to keep the interpretability of the fuzzy system, unlike the parametric architectures, such as the algorithm ANFIS (Adaptive Network-based-Fuzzy Inference Systems) in which the resulting functions are the real domain polinomais [30]. Figure 7 shows the schematic diagram of the fuzzy multilayer model for a system composed of two inputs and three inference rules enabled.
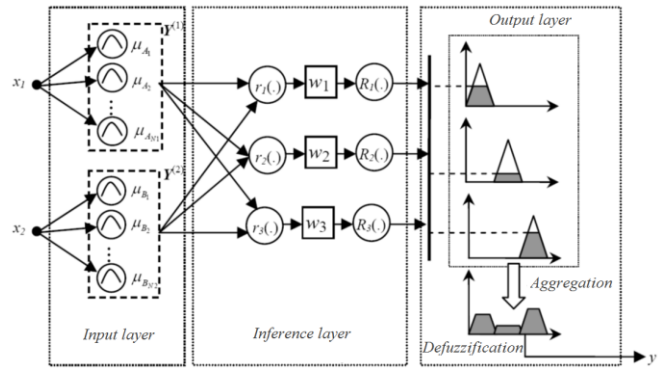
Figure 7.   Multilayers fuzzy system.

To evaluate the performance of robust fault location tool, a set of samples test was applied to the system. The estimation showed average error of 11.07 m, standard deviation of 12.25 m and maximum error of 68.53 m. System testing is represented graphically by Fig. 8. Each level represents the same fault point for the 64 connection of generators combinations and therefore the criterion of robustness is built into the criterion of accuracy. Figure 9 contains the histogram of the error, by which it should be noted that the highest errors occurred at generators closer point.
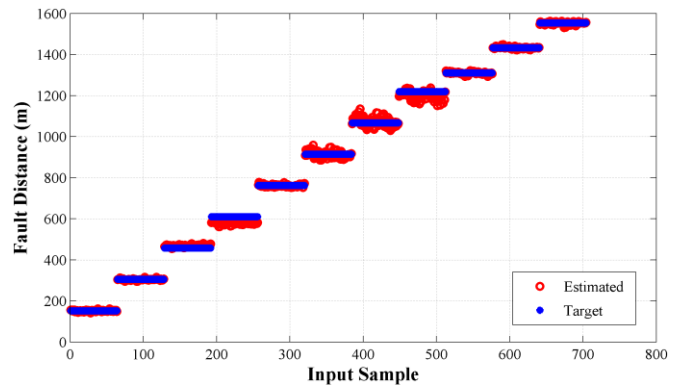
Figure 8.   Robust fault location test.
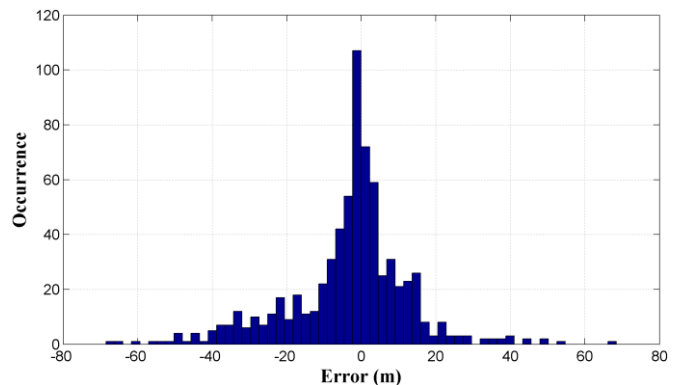
Figure 9.   Error histogram of the robust fault location.

V.   Conclusions

Even from an optimistic point of view for the fault location methods, that is, applying only LIF, the PDS with DG showed a variability in the apparent impedance higher than 30% in its modulus and 15% in its angle for SLG faults in relation to the system without DG. This level of variation in the system behavior is the result of the uncertainty in the operation of the DG unit. Since this is not the only source of uncertainty in this scenario, the methods of locating faults on distribution feeders with distributed generation should be robust. For this purpose a fuzzy system was set to the task of robust fault location. The estimation showed an average error of 11 m and a maximum error of 68 m, which represents a response acceptable in view the facility in to complement the fault point identification by visual inspection. In future works it is intended to extend this study to cases involving other types of fault, different fault resistances, fluctuations in the generation of DG units, different types of generators, specific power for each DG unit and representation of the generators by their complete models.

Acknowledgment

References

[1] A. Ipakchi and F. Albuyeh, "Grid of the future," IEEE Power and Energy Magazine, vol. 7, no. 2, pp. 52–62, Mar. 2009.

[2] A. Soroudi, "Possibilistic-Scenario Model for DG Impact Assessment on Distribution Networks in an Uncertain Environment," IEEE Transactions on Power Systems, vol. 27, no. 3, pp. 1283–1293, Aug. 2012.

[3] D. E. Newman, B. A. Carreras, M. Kirchner, and I. Dobson, "The Impact of Distributed Generation on Power Transmission Grid Dynamics," in 2011 44th Hawaii International Conference on System Sciences, 2011, no. January, pp. 1–8.

[4] R. Villafafila-Robles, A. Sumper, B. Bak-Jensen, and E. Valsera-Naranjo, "Probabilistic analysis in normal operation of distribution system with distributed generation," in 2011 IEEE Power and Energy Society General Meeting, 2011, pp. 1–8.

[5] J. A. Silva, H. B. Funmilayo, and K. L. Bulter-Purry, "Impact of Distributed Generation on the IEEE 34 Node Radial Test Feeder with Overcurrent Protection," in 2007 39th North American Power Symposium, 2007, pp. 49–57.

[6] W. El-Khattam, Y. G. Hegazy, and M. M. A. Salama, "Investigating Distributed Generation Systems Performance Using Monte Carlo Simulation," IEEE Transactions on Power Systems, vol. 21, no. 2, pp. 524–532, May 2006.

[7] M. T. Doyle, "Reviewing the impacts of distributed generation on distribution system protection," in IEEE Power Engineering Society Summer Meeting, 2002, vol. 1, pp. 103–105.

[8] E. Ebrahimi and A. Ghanizadeh, "Impact of Distributed Generation on Fault Locating Methods in Distribution Networks," in International Conference on Renewable Energies and Power Quality (ICREPQ'12), 2012, pp. 1–5.

[9] P. Chen, Z. Chen, and B. Bak-Jensen, "Probabilistic load flow: A review," in 2008 Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, 2008, no. April, pp. 1586–1591.

[10] G. Carpinelli, G. Celli, F. Pilo, and A. Russo, "Distributed generation siting and sizing under uncertainty," in 2001 IEEE Porto Power Tech Proceedings (Cat. No.01EX502), 2001, vol. vol.4, p. 7.

[11] A. Soroudi, M. Ehsan, R. Caire, and N. Hadjsaid, "Possibilistic Evaluation of Distributed Generations Impacts on Distribution Networks," IEEE Transactions on Power Systems, vol. 26, no. 4, pp. 2293–2301, Nov. 2011.

[12] J. Qiang, Z. Shuo, and L. Yong-li, "A study on capacity of distributed generation and its effect on short circuit current at micro-grid operation mode," in 2011 4th International Conference on Electric Utility Deregulation and Restructuring and Power Technologies (DRPT), 2011, pp. 1109–1112.

[13] W. El-Khattam, Y. G. Hegazy, and M. M. A. Salama, Stochastic analysis of electrical distributed generation systems, vol. 2. 2003, pp. 2–5.

[14] M. Zangiabadi, R. Feuillet, and H. Lesani, "An approach to deterministic and stochastic evaluation of the uncertainties in distributed generation systems," in Electricity Distribution - Part 1, 2009. CIRED 2009. 20th International Conference and Exhibition on, 2009, pp. 1 – 4.

[15] C. Zheng, D. Xinzhou, and L. Chengmu, "Robustness of one-terminal fault location algorithm based on power frequency quantities," in IEEE Power Engineering Society Summer Meeting,, 2003, pp. 1118–1122.

[16] D. X. D. Xinzhou, C. Z. C. Zheng, H. X. H. Xuanzhou, W. K. W. Kehong, and L. C. L. Chengmu, "Optimizing solution of fault location," in IEEE Power Engineering Society Summer Meeting, 2002, vol. 3, pp. 1113–1117.

[17] F. Ruiz-Rodriguez, J. C. Hernández, and F. Jurado, "Probabilistic load flow for radial distribution networks with photovoltaic generators," IET Renewable Power Generation, vol. 6, no. 2, pp. 110– 121, 2012.

[18] F. Alvarado, Y. Hu, and R. Adapa, "Uncertainty in power system modeling and computation," in [Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics, 1992, pp. 754–760.

[19] M. E. El-Hawary, Electric Power Applications of Fuzzy Systems. IEEE, 1998.

[20] A. Hadian and M.-R. Haghifam, "Placement of DG with stochastic generation," in IEEE PES T&D 2010, 2010, pp. 1–7.

[21] D. Villanueva, J. L. Pazos, and A. Feijoo, "Probabilistic Load Flow Including Wind Power Generation," IEEE Transactions on Power Systems, vol. 26, no. 3, pp. 1659–1667, Aug. 2011.

[22] S. Jamali and V. Talavat, "Dynamic fault location method for distribution networks with distributed generation," Electrical Engineering, vol. 92, no. 3, pp. 119–127, Aug. 2010.

[23] J. Faig, J. Melendez, S. Herraiz, and J. Sánchez, "Analysis of Faults in Power Distribution Systems With Distributed Generation," in International Conference on Renewable Energies and Power Quality, 2010, p. 6.

[24] W. El-Khattam, Y. G. Hegazy, and M. M. A. Salama, "Stochastic power flow analysis of electrical distributed generation systems," in 2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491), 2003, pp. 1141–1144.

[25] W. H. Kersting, "Radial distribution test feeders," in 2001 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.01CH37194), vol. 2, pp. 908–912.

[26] W. A. Elmore, Protective Relaying: Theory and Applications, 2nd ed. New York: Marcel Dekker, 2004, p. 410.

[27] S.-J. Lee, M.-S. Choi, S.-H. Kang, B.-G. Jin, D.-S. Lee, B.-S. Ahn, N.-S. Yoon, H.-Y. Kim, and S.-B. Wee, "An Intelligent and Efficient Fault Location and Diagnosis Scheme for Radial Distribution Systems," IEEE Transactions on Power Delivery, vol. 19, no. 2, pp. 524–532, Apr. 2004.

[28] R. A. Flauzino and I. N. da Silva, "Efficient parametric adjustment of fuzzy inference system using unconstrained optimization," Lecture Notes in Computer Science, vol. 4507, pp. 399–406, 2007.

[29] R. A. Flauzino and I. N. da Silva, "Tuning of Fuzzy Inference Systems Through Unconstrained Optimization Techniques," International Journal of Computer Research, vol. 12, pp. 243–252, 2003.

[30] Guillaume, S., "Designing fuzzy inference systems from data: An interpretability-oriented review," IEEE Transactions on Fuzzy Systems, vol.9, no.3, pp.426-443, Jun 2001

Efficient Integration of Intelligent Systems and Experimental Tests for Power Transmission Transformer Diagnosis

(Special Session – ICAI'2013: Expert Systems for Intelligent Automation Purposes on Power Systems)

C.G. Gonzales, M.A. Carrapato, T.S. Bonfim, P.I. Costa, D.H. Spatti, R.A. Flauzino, I.N. Silva

Abstract--**This paper presents detailed intelligent techniques concepts commonly used in power transformer analysis. It will be given more attention to Dissolved Gas Analysis (DGA), Transformer Turn Ratio (TTR) and Induced Voltage tests. The main purpose of the work is to establish a database and intelligent system specification that stores all data acquired from testing and therefore make it possible, for an expert system based on intelligent tools, make integration and data analysis.**

Keywords— Power transformers, expert systems, intelligent systems.

I. INTRODUCTION

Due to the recent energy demand increasing, which is directly linked to regulatory changes in the electricity sector, it is also increasing the interest and concern about reliability issues and power quality. Following this context, it can be highlighted the development of methodologies dedicated to fault detection and location to all segments of the electrical system, since from the generation, through transmission and reaching the distribution.

Failures within each of these sectors leads to undesirable effects, such as power interruption and transient fluctuations, which can damage both companies and consumers equipment's, creating economic losses and decreasing power quality.

Power transformers are considered one of the most important assets for the transmission systems in nowadays due to its operational capability and to be a critical point of the system. Thus, in a case of a power transform failure, many consumers will be interrupted.

Because it's extremely expensive the acquisition of a new transformer or the maintenance after a failure, several methodologies have been proposed in literature with the goal for both efficient detection [1-2] and location [2-4] of transmission transformers failures.

These approaches are mainly centered in: DGA [5-9], TTR [10] and induced voltage tests [11-12].

Failures detection approaches by DGA in the insulation oil tank is based on the observation of partial discharges,

discharges of high energy, corona effects, among others, interact with the transformer insulation oil, performing changes in its chemical composition and physical properties.

In general, these changes are accompanied by the release of gases, which can be dissolved in insulating oil. The volume of these gases dissolved can be determined by chromatographic and / or physicochemical analysis. The last one can be realized in the installation site of the transformer. The gas chromatography only can be realized in a very equipped laboratory, using insulation oil samples.

The TTR tests aims to diagnose the transformer over its winding (primary and / or secondary), where it is checked the relationship between the turns of the high-voltage and low-voltage transformer under analysis [10].

The induced voltage tests are used to check dielectrics issues in the transformers, and must be realized in field. However, it should be noted that the induced voltage tests may be different depending on the transformer class, as pointed in [13-14].

Following the entire presented context, this work aims to guide the design of an expert system capable to provide solutions and help de decision making by maintenance engineers, when analyzing failures in transmission transformers. In other hand, for achieving this goal all data from tests must be stored and handled properly and then integrated into the expert system platform, which will analyze the failure.

II. DGA

The diagnosis of the transformers operating conditions is fundamental in order to improve economic and reliable operation. The aging and deterioration of transformers determine the end of his life because the presence of failures and maintenance procures can decrease its reliability or availability. In that way it is important to know about the deterioration mechanisms and also have means for predictive maintenance actions.

The deteriorated insulation oil is one that has products from oxidation, while the contaminated insulation oil is one that has water and other substances than those commonly found in the oil [8].

The mineral oil degradation process can be presented in three distinct phases. The initial stage is characterized by the formation of chemically unstable peroxides which release oxygen and create oxy-cellulose compounds with the

722   *Int'l Conf. Artificial Intelligence | ICAI'13 |*

2

insulation paper. The intermediate phase is characterized by the formation of gases. The final stage is represented by the formation of sludge, which mainly consists of a resinous substance resulting from the polymerization of acids and other compounds.

The gases dissolved in insulating oils are derived from failures occurred inside the transformer. The relationship of these gases can be classified according to the type of failure, and its formation may indicate the severity of the failure.

Generally, identification and quantification of the types of gas are used to carry out a maintenance planning. Using DGA, several methods are proposed in the related literature in order to evaluate natural aging of the insulation as well as detection of incipient failures. The motivation dedicated to DGA failures detection in transformers is justified by the amount of possible failures which can be identified and the reliability of the results [9].

III. TTR

The Transformer Turns Ratio is used to determine the real relation between the windings. These methodology is capable to identify if any short-circuit is occurring between the turns. The transformer can't be energized for the procedure. After the test, it can be possible to say if there is necessary or not a internal inspection of the transformer. Exists two ways for this test: by the *bridge* method or the method of voltage measurement.

The *bridge* method consists to apply a voltage in a phase winding using equipment known as a *bridge* and measuring the induced voltage in this equipment. The procedure is repeated for all phases and tap positions, in a sequential manner. During the measurement procedure, only transformation relations between windings with the same magnetic flux can be measured. Generally, this method uses an AC voltage of 220V and nominal frequency of the network; however, other values can be used.

The theoretical empty value of the TTR is given by the voltage on the high voltage winding divided by the voltage on the low voltage winding. This relationship is set in the standard transformer, until the balance can found, with 0% error.

In the TTR method, the voltages on the windings are measured simultaneously to determine the ratio of transformation. Measures should present results of less than 0.5% of error.

Therefore, the TTR are intended to verify the relationship between the number of turns of the high voltage side to the number of turns of the low voltage side [14].

This test must be performed with the power transformer without load and operating with nominal voltage applied to the coil which has the smallest number of turns. In the other winding must be applied a voltage of 0.5% in relation to the nominal. The standard related in [14] was designed for three-phase four wires transformers connected in Y, which neutral is considered. Consequently, for other types of transformers or connections must be necessary numerical changes such as shown in [13].

In the work presented in [6] was developed a prototype for analysis and *in site* testing of distribution transformers called TXWIN. The TXWIN allows the user to perform a no-load transformer test, losses and TTR automatically, using a graphical interface. Additionally, this interface also contains other tests such as: resistance measure, high voltage tests an insulation test.

It should be mentioned that the prototype also has the ability to analyze the results, generate custom reports and store the data for each of the performed test on a database. The TTR tests must be performed based on the standards IEC76, BS171 and / or ANSI C57.

IV. INDUCED VOLTAGE TEST

The induced voltage test aims to check the insulation between the phases of the transformer, between the turns of the winding and between the input terminals of the winding and ground. During the test, normally a voltage is applied to the low side, while the high side is left open and grounded at a common point.

Since the voltage used during the tests is higher than the operating voltage of the low side of the transformer, the frequency of the voltage test must be higher at least twice as the operation frequency in order to preventing over-saturation of the core. The voltage test is selected according to the winding with a higher operating voltage. It is also possible to perform tests using operational voltages.

The test voltage could be measured using a voltage divider or voltage transformer connected to the high side. It's important highlight that the test time cannot be less than 15 seconds.

The test can consider successfully realized when no surge, breakdown voltage or increasing current is observed. The tests are considered long term or short term according to the reference voltage of 72.5 kV, as described in IEC 60076-3. Transformers with operating voltage not exceeding 72.5 kV, partial discharge measurements are not mandatory. However, for transformers with a voltage exceeding 72,5 kV, it is necessary to acquire measurements of partial discharges during the induced test voltage.

V. INTELLIGENT SYSTEMS

As previously mentioned, this paper aims to present a method for storing, integrating and analyzing data from tests on transmission transformers. Hence, this section presents the fundamental concepts of intelligent systems, which is used during the stage of data integration in order to obtain a decision making support tool.

It must been emphasize that the operator decision usually infers the maintenance of the transformer. So, a bad decision could result in erroneous removal of a healthy transmission transformer or not remove a faulty transformer.

For this reason, the use of intelligent systems based tools becomes extremely relevant as bad decisions increase maintenance costs to the company and decreases reliability.

3

A. Fuzzy Inference System

Called Fuzzy Systems are derived from the theory of fuzzy sets and fuzzy logic, developed by L. A. Zadeh in 1965, in order to represent knowledge from uncertain and imprecise data. The Fuzzy Systems consist of a way to approximate a computational decision a human decision. Figure 1 shows a block diagram expressing a simplified operation of a fuzzy system.
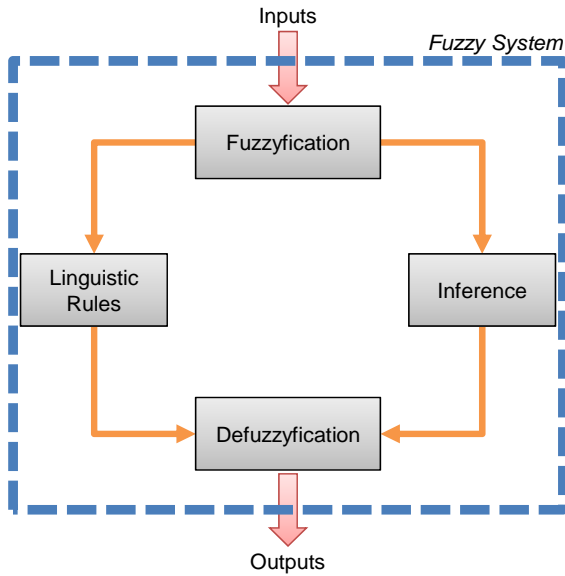
Fig. 1. Block diagram of a fuzzy system.

In the "Fuzzyfication", the input values (in this case, the information provided by the acoustic emission sensors, gas concentration and electrical measurements) are aquired and conditioned, becoming fuzzy sets. Similarly, the block "Defuzzyfication" is responsible for transforming the output of the fuzzy system in non-fuzzy values (ie, indicate the type of internal failure and the location of it). The block "Linguistic Rules" store the linguistic sentences which main function is represent all knowledge about the problem.

These rules may be given by specialists or automated methods such as ANFIS (Adaptive Neuro-Fuzzy Inference System). The "Inference" block does the mapping of a system by linguistic rules. Therefore, joining rules with fuzzy sets acquired by the "Fuzzyfication", the system is able to determine the behavior of the output variables, by the "Defuzzyfication" and generate the corresponding Outputs for given Inputs values.

When using a fuzzy inference system, rules and fuzzy sets are adjusted by information specialists. However, in some cases, due to the complexity and nonlinearity of the problem, it is necessary to use hybrid systems such as neuro-fuzzy systems, where adjustments are performed in an automated way according to the data set representing the process. Regardless of the setting, the whole system has fuzzy linguistic rules that can be represented as follows:

$$R_i: \textbf{If } Input\ 1\ is\ x_1\ \textbf{and}\ Input\ 2\ is\ x_2$$

$$\textbf{Then } Output\ is\ y_i = a \cdot x_1 + b \cdot x_2 + c$$

In the "Inference" block a several methods can be used for this procedure. The commonly used methods are Mamdani and Takagi-Sugeno.

B. Artificial Neural Networks

The Artificial Neural Networks (ANN) are computational models inspired in the human brain which have de capability to obtain knowledge by means of a learning procedure. Among the various architectures of ANN, the Multilayer Perceptron (MLP) is the most used. This type of architecture is usually applied in pattern recognition tasks, functional approximation, identification and control [15]. The structure of an ANN can be designed according to Fig. 2.
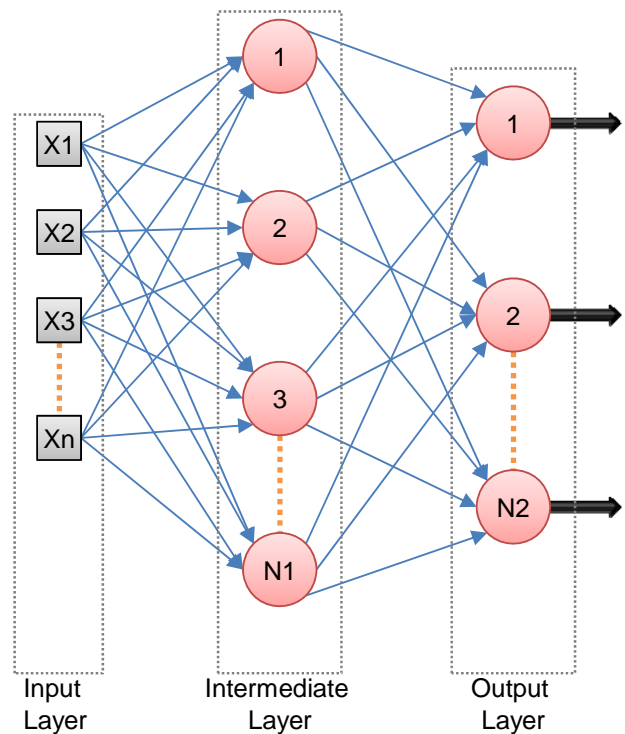
Fig. 2. ANN structure type MLP.

As can be seen in Fig. 2, the ANN structure is basically composed of the "Input Layer", "Intermediate Layer" and "Output Layer". There is between the layers a set of synaptic weights, represented by a matrix which is adjusted during the training phase. It should be also mentioned that for each neuron (intermediate layers and output) is needed to apply activation functions in order to limit the output. Other factors that must be explored are the training and validation stages of an ANN.

During the MLP ANN training phase, some algorithms can be applied. The most used is the *backpropagation* algorithm which calculates the descend gradient to reach the best-fit matrix of synaptic weights. In addition to the *backpropagation* algorithm, the *Levenberg-Marquardt* algorithm has been widely used because of its ability to accelerate the

convergence process due this method is a Newton approximation method-based for nonlinear systems [15].

The validation stage has the purpose to verify the integrity of the training previously conducted, analyzing the learning ability (generalization) of the ANN.

VI. Storage, Integration and Data Analysis

As previously mentioned, this paper main objective is to present an approach for storing, integrating and analyzing data from transmission transformers tests. Figure 3 shows the proposed approach for this goal.
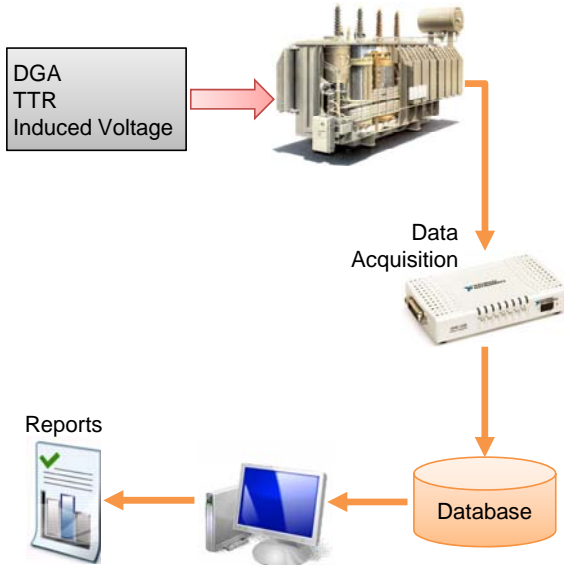
Fig. 3. Proposed Approach for storage, integration and data analysis from transmission transformers tests.

A. Data Storage

The data acquired after the tests performed on the transformer under analysis must be appropriate stores in a database. Therefore, for this purpose an Entity-Relationship Model (ERM) of this database was developed in order to maintain proper documentation and shall also facilitate queries to stored data. By Fig. 4 can be observed the ERM implemented which can be used for any management system database (eg MySQL, Microsoft SQL Server, and Oracle).
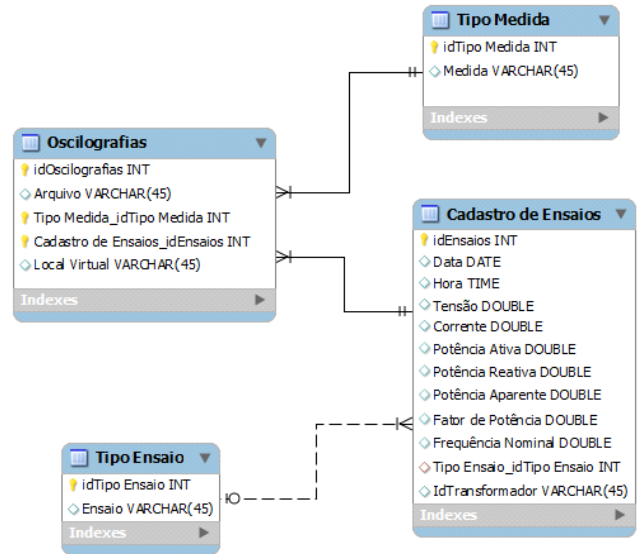
Fig. 4. Entity-Relationship Model for the propsed approach.

It can be noted by means of Fig. 4 that the database has a main table called "Cadastro de Ensaios" where are registered all tests for each transformer. For this reason, there is a field called "idTransformador", which should be linked to the company asset database. In this table others electrical information's can be registered, such as: voltage and current (rms values), active, reactive and apparent power, power factor, and nominal frequency of the test. The history of the entire test can be saved in proper fields of date and time. Finally, it is possible to notice the field "TipoEnsaio_idTipoEnsaio" representing a foreign key responsible for the relationship with the "TipoEnsaio" table.

This table has the function to store all possible tests that can be realized. Therefore, this database becomes very flexible, because it is independent of the test type performed, consequently could also be used for transformers in transmission or distribution.

In addition to the above tables, it is important to note the existence of the table "Oscilografias" that store the oscillography files paths and identifies the type of measurement that was performed for each file. The table "Tipo Medida" is responsible to store de log of all possible measurements that can be realized in a test.

B. Selecting Data to be Integrated in the Expert System

After the data storage procedure it is necessary apply a selection process that will determine which data will integrated in the Expert System. This subsection focuses primarily on some techniques that can be applied for this process.

The database is formed by a large number of attributes (each variable / stored measurements), which must be properly in relevance evaluated for the purpose of the transformer diagnosis. The attributes selection process is defined by [16] as the determination of a subset of attributes that will be responsible for generalizing the information contained in the database. This process is dependent to the response expected

for such database.

Attribute selectors are very used when sorting algorithms are unable to obtain the pattern generalization, or when these algorithms are impractical due to the large number of inputs.

In this study, the two assumptions may be considered because, by using all the acquired data becomes much more complex generalized solutions, in addition, there are also a large number of attributes that make the expert system very inefficient.

Methods for this purpose can be divided into two classes: *wrappers* [17] and *filters* [18-19]. *Filters* differ *wrappers* just about the independence of the learning algorithm that will be used later. Typically, *wrappers* have considerable performance when the learning algorithm is supervised [16-20].

The performance of the wrappers was previously cited as significant due to performance tests in the work [17] which several databases commonly used to benchmark were used and verified the performance of this attribute selector.

Some authors support the idea that the attributes selection is a central point for classification and identification tasks, since many times the selection of attributes combined in expert systems provide low efficiency and high computational costs [21].

C. Data Analysis by Expert Systems

Since the data was properly selected from those previously stored in the database, it becomes possible to analyze them. However, this analysis can be performed either by an expert system based on ANN or on a fuzzy-based inference system.

An ANN system needs a previous training to produce generalized diagnoses results for the operator.

In other hand, fuzzy-based inference systems are adjusted by expert knowledge and require no training. In this case, the use of fuzzy inference system becomes a more attractive way due the capability to extract knowledge for uncertain data and treat imprecise information.

VII. CONCLUSIONS

The diagnosis of transmission transformers failures is extremely necessary since this equipment has the largest aggregate cost both for acquisition and maintenance from power companies.

Therefore, the DGA, as well as the TTR tests and Induced Voltage tests are essential for the maintenance of equipment, and brings many benefits such as reduced risk of unexpected failures and unplanned shutdowns; extension of the transformer lifetime, decreasing maintenance costs and reducing outage time.

Besides, a system that performs storage, integration and analysis of data makes the diagnosis even more reliable and contributes effectively to the decision making support that should be performed by the operator.

VIII. REFERENCES

[1] X. Wang, B. Li, H. T. Roman, O. L. Russo, K. Chin and K. R. Farmer, "Acousto-optical PD Detection for Transformers", *IEEE Trans. on Power Delivery,* vol. 21, pp. 1068-1073, 2006.

[2] S. M. Markalous, S. Tenbohlen and K. Feser, "Detection and Location of Partial Discharges in Power Transformers using Acoustic and Electromagnetic Signals", *IEEE Trans. on Dielectrics and Electrical Insulation,* vol. 15, pp. 1576-1583, 2008.

[3] E. Mohammadi, M. Niroomand, M. Rezaeian and Z. Amini, "Partial Discharge Localization and Classification Using Acoustic Emission Analysis in Power Transformer", *31st International Telecommunications Energy Conference,* 2009.

[4] G. F. C. Veloso, L. E. B. Silva, G. Lambert-Torres and J. O. P. Pinto, "Localization of Partial Discharges in Transformers by the Analysis of the Acoustic Emission", *IEEE International Symposium on Industrial Electronics,* 2006.

[5] J. R. Guardado, J. L. Naredo, P. Moreno e C. R. Fuerte, "A comparative study of neural network efficiency in power transformers diagnosis using dissolved gas analysis", *IEEE Transactions on Power Delivery*, v. 16, n. 4, p. 643-647, 2001.

[6] M. Duval, "A Review of faults detectable by gas-in-oil analysis in transformers", *IEEE Electrical Insulation Magazine*, v. 18, p. 8-17, 2002.

[7] M. M. Imamura, "Uma Abordagem Neural Para Identificação de Gases Dissolvidos em Óleo Isolante de Transformadores em Função de Parâmetros Físico-Químicos", *Dissertação de Mestrado em Engenharia Industrial*, Faculdade de Engenharia, Universidade Estadual Paulista, Bauru, 2000.

[8] M. Milasch, "Manutenção de Transformadores em Líquido Isolante", *Editora Edgar Blucher Ltda*, 1984.

[9] H. Herman, M. J. Shenton, G. C. Stevens e R. J. Heywood, "A new approach to condition assessment and lifetime prediction of paper and oil used as transformer insulation", *Proceedings of 7th International Conference on Solid Dielectrics*, p. 473-476, 2001.

[10] S. P. Zhu, J. Shi, Y. S. Lim, C. I. Tan e Y. K. Cheong, "Transformer Testing and Analysis with a Window-based Graphic User Interface", *International Conference on Energy Management and Power Delivery*, p. 410-415, 1995.

[11] A. J. Kachler, "On-Site Diagnosis of Power and Special Transformers", *IEEE International Symposium on Electrical Insulation*, p. 362-367, 2000.

[12] H. Mohseni, A. S. Akmal, J. Jadidian, E. Hashemi, E. Agheb, A. Naieny, A. Ghorbandaeipour e K. Niayesh, "Induced Voltage Test of Distribution Transformers Using State Frequency Inverters", *IEEE/PES Transmission & Distribution Conference and Exposition*, p. 1-5, 2008.

[13] IEEE Std. C57.12.00-2006 for Standard General Requirements for Liquid-Immersed Distribution, Power, and Regulating Transformers, "IEEE Standard for Standard General Requirements for Liquid-Immersed Distribution, Power, and Regulating Transformers", 2007.

[14] IEEE Std. C57.12.90-1993, "IEEE Standard Test Code for Liquid-Immersed Distribution, Power, and Regulating Transformers and IEEE Guide for Short-Circuit Testing of Distribution and Power Transformers", 1993.

[15] I. N. Silva, D. H. Spatti and R. A. Flauzino, "Redes Neurais Artificiais para Engenharia e Ciências Aplicadas - Curso Prático", ArtLiber, 2010.

[16] H. Liu, H. Motoda, "Feature Selection for Knowledge Discovery and Data Mining", Kluwer Academic Publishers, 1998.

[17] R. Kohavi, G.H. John, "Wrappers for feature subset selection", *Artificial Intelligence*, No. 97, 273-324, 1997.

[18] H. Almuallim, T.G. Dietterich, "Learning with many irrelevant features", *Proceedings of the 9th National Conference on Artificial Intelligence*, pp. 547-552, 1991.

[19] H. Liu, R. Setiono, "A probabilistic approach to feature selection: a filter solution", *Proceedings of the 13th International Conference on Machine Learning*, pp. 319-327, 1996.

6

[20] M.A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning", *Proceedings of the 17th International Conference on Machine Learning*, pp. 359-366, 2002.

[21] M.A. Hall, G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining", *IEEE Transactions on Knowledge and Data Engineering*, No. 15, 1-16, 2003.

Expert System for Accurate Diagnosis of Power Transmission Transformer

C.G. Gonzales[1], T.S. Bonfin[1], P.I. Costa[1], M.A. Carrapato[1], I.N. Silva[2], R.A. Flauzino[2], and D. Spatti[2]
[1]São Paulo State Electric Power Transmission Company (CTEEP), Bauru, SP, Brazil
[2]University of São Paulo, USP/EESC/SEL, São Carlos, SP, Brazil

Abstract – *This paper presents a tool for managing, storing and processing of various experimental tests performed on power transmission transformers. Such devices represent one of the most valuable assets for utilities and their service removals (by non-scheduled ways or failures), result in long periods of disruption for consumers as well as large financial losses. Thus, there is a constant search involving the development of tools to accomplish the task of identifying prominent failures, especially those that combine conventional and intelligent tools for this purpose.*

Keywords: *fault diagnosis; expert systems; intelligent systems; power transmission transformer.*

1 Introduction

Equipment belonging to the power transmission system has a complex maintenance plan, aimed at testing compliance.

There are specific procedures that enable investigate different scenarios of operation of these devices with or without the presence of failures. Such experimental tests are mostly performed manually, which can affect the accuracy associated with the metrological aspects [1].

The power transmission transformer is currently considered one of the critical points of the system, since its service removal promotes various disorders for consumers and also for utilities.

When submitted to failure condition, testing procedures become critical, because elevate the level of result precision to levels that do not allow incorrect diagnoses [2-4].

Such needs encourage the development of new tools to produce efficient results, which aim to solve these problems in an automated way, combining and enhancing existing tools or proposing new methodologies [5-9].

In this context, we can highlight the integration of techniques, methods and algorithms for obtaining optimal results.

The consistency between tools has proven to be one of the major challenges for the scientific and technical community, because the processes of detecting and locating failures in power transmission transformers are associated with various nonlinearities, besides infinity of data collected that need to be properly treated [5].

The combination of expert tools around a particular goal in order to get optimal results is known by Expert System [10].

In fact, in such systems, the great difficulty lies exactly in the processes involved in the evaluation of the results produced by the tools used in solving the subproblems. It is also important to highlight that the routines for the tests may display the results in different ways, where data in addition to having different formats also differ in volumes [11].

Thus, this paper presents all technical and scientific criteria involving the development of an expert system responsible for managing experimental tests performed on power transmission transformers, taking into account all aspects normative pointed in [12-14].

Additionally, we will present the main concepts involving the integration of conventional and intelligent tools, exploring aspects of data storage, signal processing, training of artificial neural networks, tuning of fuzzy systems and feature selection techniques, which effectively contribute to the processes of detecting and locating failures in power transmission transformers

2 Data integration

As previously mentioned, data integration is a task of great importance in this study, since these data can be obtained either from preexisting databases in power utilities or by means of tests to be performed in transformers. Moreover, it is important to note that these tests may generate different data in various formats and, therefore, the integration of all possible data to be analyzed becomes a complex task.

Therefore, we developed a database that will be responsible for integrating these data. In addition, an attribute selection stage is employed in order to reduce the dimensionality of the data used as input for the expert system.

2.1 Database modeled and implemented to achieve integration of data

After carrying out experimental tests on the transformer to be analyzed, the data are then obtained and, therefore, it is pertinent to store them in a database that is reliable and safe from the point of view of the power utility. Therefore, we obtained an Entity-Relationship model of this database for this purpose. This model was derived based on the test reports provided by the utility. In order to better illustrate these reports, an example may then be visualized by means of Figure 1.

Ensaio de Fator de Potência
Autotransformadores

DADOS GERAIS

EXECUÇÃO DO ENSAIO

1	Alta e Baixa	Carc.	Terc.	Ground	-			+	
2	Alta e Baixa	Carc.	Terc.	Guard	C A			+	
3	Alta e Baixa	Carc.	Terc.	UST	C AT			+	
4	Terc.	Carc.	Alta e Baixa	Ground	-			+	
5	Terc.	Carc.	Alta e Baixa	Guard	C T			+	
6	Terc.	Carc.	Alta e Baixa	UST	C TA			+	
7	Todos	Carc.	-	Ground	C Todos			+	

VALORES CALCULADOS

Ensaio 1 - 2	C AT
Ensaio 4 - 5	C TA
Capacitâncias	

CORRENTE DE EXCITAÇÃO

OBSERVAÇÕES

Fig. 1. Example of test report used by the power utility.

In addition to the test reports, we also use data reports used for maintenance and inspection of transformers and devices that compose them (Figure 2).

Relatório de Manutenção de Acessórios de Transformadores

DADOS GERAIS

PONTOS DE VERIFICAÇÃO

Lubrificações / Vazamentos	Aterramento / Vazamentos
Vazamentos/Membranas/Bolsas / Indicador de nível / Registros e Tubulações / Secador de Ar	Funcionamento / Caixa de ligação / Vazamentos / Fluxostato
Vazamentos / Contatos	Funcionamento / Caixa de ligação
Válvulas / Bujões / Vazamentos	Vazamentos / Caixa de Terminais / Ensaios de Atuações
Funcionamento / Caixa de Ligação / Ruído anormal	Nível de Óleo / Porcelana / Vazamentos
Guarnição da porta / Régua borne / Fiação / Iluminação interna / Resistência de aquecimento / Componentes / Sinais de aquecimento	Estado de conservação / Óleo da cuba / Juntas de vedação / Sílica gel / Visores
Indicador de nível do transformador / Indicador de nível do comutador / Válvula de alívio de pressão / Relé de gás / bucholz	Termômetro de óleo / Termômetro de enrolamento / Rele de fluxo / pressão comutador / Bolsa / Membrana furada

OBSERVAÇÕES

Fig. 2. Example of maintenance/inspection report used by the power utility.

Thus, the database has a main table named "Cadastro Ensaios", which are cataloged all tests for each transformer. For this reason, there is a field called "idTransformador", which should be related to a database of records of assets of power utility. Also, in this table are values stored as current and voltage (rms values), as well as measures of real, reactive and apparent power, power factor, and nominal frequency used for the respective test. It may still be

observed that there are time and date fields, which are stored so that there is a history of testing for each transformer. Finally, it is possible to notice the field "Tipo Ensaio_idTipoEnsaio" represents a foreign key so that there is the relationship with the table "Tipo Ensaio".

Table "Tipo Ensaio" was created in order to register all possible tests to be performed. Therefore, this database becomes very flexible because it is independent of the type of test to be carried out, and could also be used for both transmission and distribution transformers.

In addition to the above tables, it is important to note the existence of the table "Oscilografias", which includes the paths of oscillography files and identifies the type of measurement that was performed and is stored in each file. For this reason, a table "Tipo Medidas" was also created, which is responsible for storing the registration of the possible types of measures that can be collected.

2.2 Selection of data to be integrated into the expert system

After the data is properly stored in the structure of the database proposed in Subsection II.A, these should be selected in order that the expert system can present a failure diagnosis more appropriate. This data selection procedure can be visualized by means of Figure 3.

Reducing the Number of Attributes → Attributes Selection → Correlation Consistency Wrapper
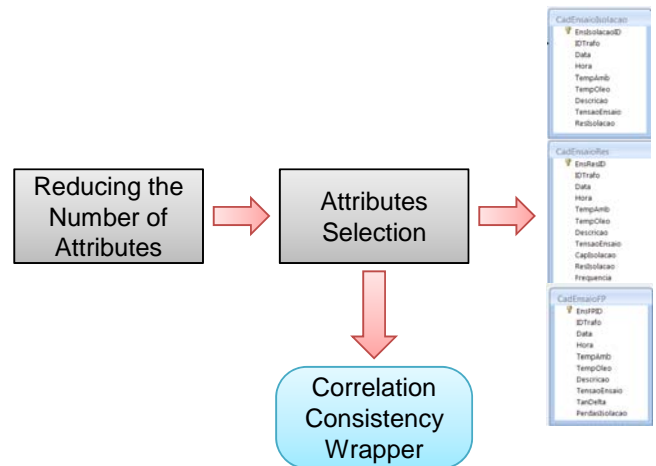
Fig. 3. Procedure for data selection.

This procedure is also known in the correlated literature as attribute selection or feature selection. Therefore, some computational tools can be used in order to obtain a reduction of the number of attributes to be used as eventual inputs to the expert system.

It should be mentioned that the process of selection of attributes is defined by [15] as the determination of a subset of attributes that will be responsible for generalizing the information contained in the database, according to the response that is expected from this database, in this case, the diagnosis of the transformer.

Among the methods of selection of attributes that are more employed, we can highlight the following:

- Wrapper [16];
- Correlation [17-18];
- Consistency [17-18].

The primary difference between these methods is that the

wrapper uses a supervised algorithm to check the cost of one attribute or subset of selected attributes. Due to this fact, in some cases, the wrapper achieves further satisfactory response than methods based on correlation or consistency. However, the computational effort of the wrapper tends to be much larger than the others, except for methods that employ exhaustive searches to obtain the best subset of attributes. In order to better understand the method wrapper, the same is then illustrated by Figure 4.
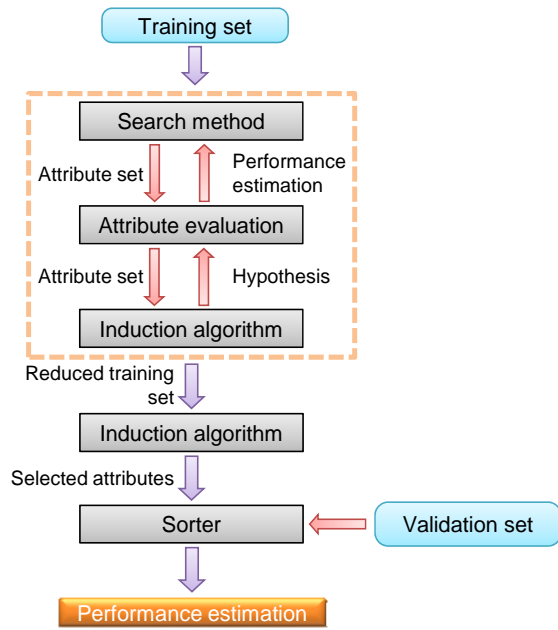
Fig. 4. Wrapper method for attribute selection.

It is noteworthy that some authors defend the idea that the selection of attributes is crucial for tasks aimed at identifying and classifying patterns, since the non-selection of the attributes can take the expert system to a low performance and high cost computing [19].

3 Implementation of the decision support system

Since the data properly selected from those previously stored in the database, it becomes possible to analyze them. However, this analysis must be performed by the expert system, which is represented by a computational system which has been developed according to the block diagram shown by Figure 5.
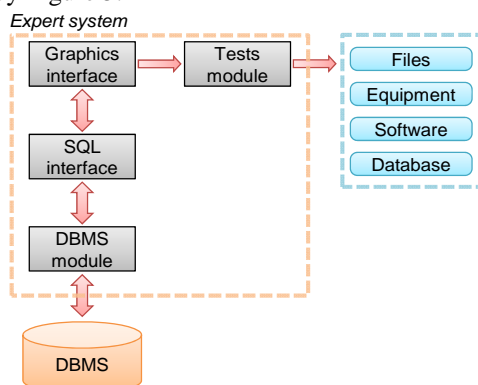
Fig. 5. Structure of the computer system with emphasis on the integration of data from experimental tests.

It is noted by Figure 6 that the data may already be stored in the following media:

- File;
- Software;
- Database;
- Equipment.

However, if they are stored in the file, their readings can be taken and, through a computational implementation, this file can be read properly and the data stored in the database previously provided. If these data are already stored in a database, one can create a lookup table in the database proposal, which is a mirror of that in which the data are located.

Thus, the more complex cases are observed when there is a need for data export via software or when they are stored in the internal memories of those devices that perform the tests.

Also with respect to Figure 5, it is important to note that the database is represented by a module called DBMS (Database Management System), because this type of system has better management capacity and data integration. In order to systematize a DBMS, Figure 6 shows the same in the form of a block diagram.
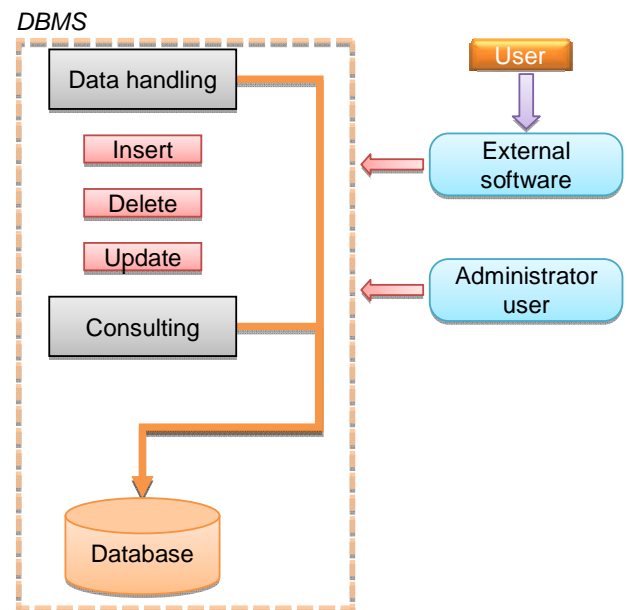
Fig. 6. Block structure for the data preprocessing.

It is noteworthy that the systematics presented in this block diagram has some characteristics that make them attractive from the point of view of the storage, management and integration of data, i.e.:

- become more agile handling and access to information;
- reduces the time for software development;
- provides information on the time required;
- enables to integrate information from distributed databases;
- reduces redundancy and inconsistency of information;
- enables the sharing of data;
- applies security restrictions through the manager of the database, and
- reduces problems related to data integrity.

Finally, it should be noted the overall structure of the computer system, which comprises a module of data preprocessing, a processing core and a graphical user interface (Figure 7). Thus, each of the parts of this system will be explained in greater detail in the following subsections.
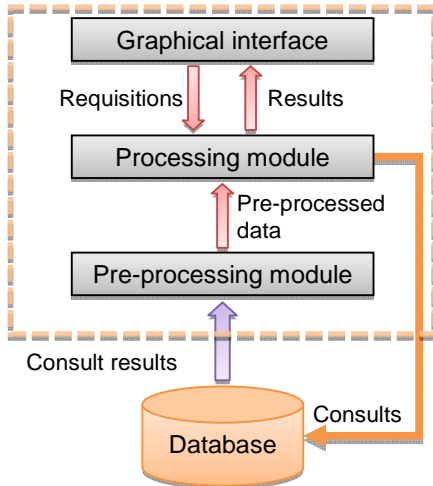
Fig. 7. General structure for the computer system.

3.1 Data preprocessing module

The module of data preprocessing is mainly composed by attribute selectors. Therefore, the structure can be defined as shown in Figure 8.
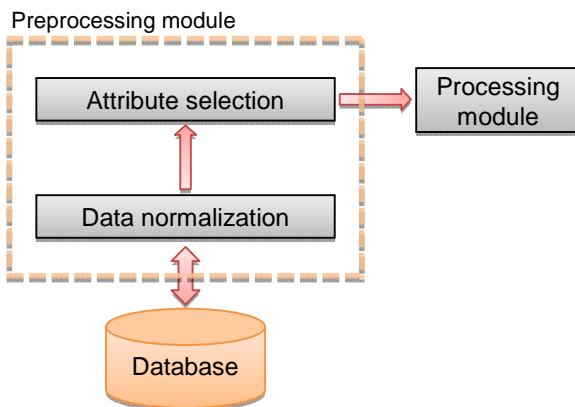
Fig. 8. Block structure for data preprocessing module.

3.2 Data processing Core

The data processing core proposed in this research was designed employing intelligent systems, which are used fuzzy inference systems and artificial neural networks. Thus, it is possible to define this stage of the computer system as shown by Figure 10.

Importantly, an expert system based on artificial neural networks need a previous training for a diagnosis to be provided to the operator. By contrast, systems based on fuzzy inference are adjusted via expert knowledge and require no training. So, in this case, the use of fuzzy inference systems becomes a more attractive alternative, since it allows to add expert knowledge to the ability of this class of intelligent system has to deal with imprecise and uncertain data.
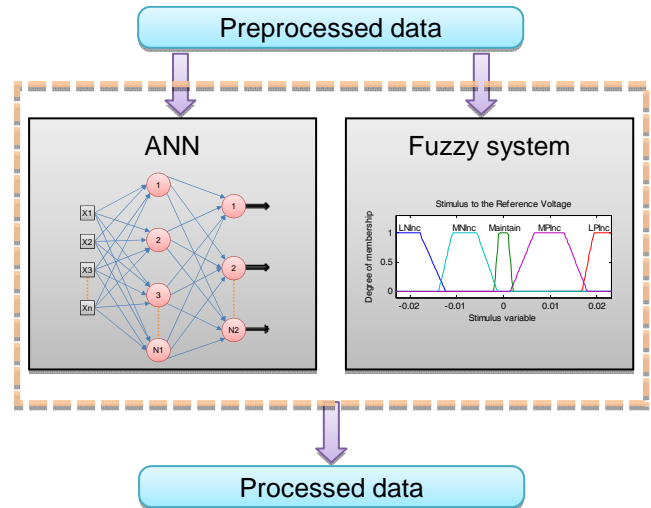
Fig. 9. General structure for the data processing core

3.3 Graphical user interface

Finally, this module is a Graphical User Interface (GUI), which will be responsible for handling all requests made by the user/operator. Thus, all necessary information will be available through the GUI. It is noteworthy that the interface is still being implemented; however, some parts have already been developed and can be viewed by means of Figures 10 and 11.

Fig. 10. Startup screen for the computer system.

This screen (Figure 10) shows the institutions and companies that are parts of this research and development project.

Fig. 11. Main interface of the computational system with emphasis on the selection tree of experimental tests.

Through the main interface, the user can handle all the required information, which may also view test data that have already been performed on transformers, as well as insert new records of experimental tests.

It is important to note that there is a selection tree for experimental tests (left side of the interface), where the user selects the desired test

It is necessary to highlight the selection tree for experimental tests, which allows the user to select as tests previously reported as to access the reports of inspection/maintenance of some of the assets of the power utility, namely:

- transformer accessories;
- switches;
- MCCP switches;
- mechanical thermometers.

In order to better illustrate the selection tree for experimental tests, it can be seen in Figure 12.
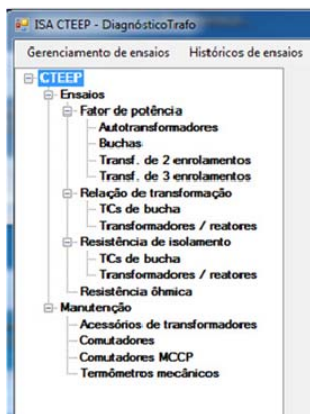
Fig. 12. Selection tree for experimental tests.

Thus, each of the experimental tests appears in the area on the side, as shown in Figures 13, 14, 15 and 16.
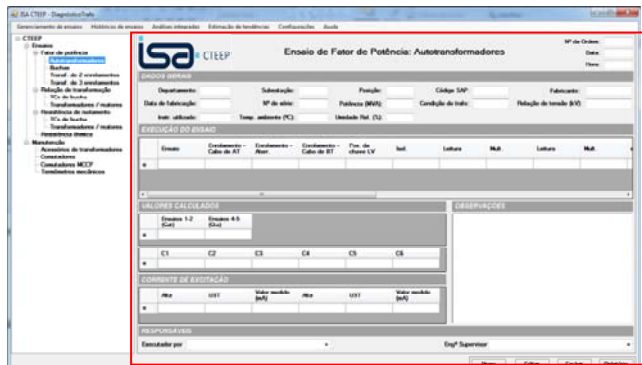
Fig. 13. Main interface of the computational system with emphasis on the viewing area of the power factor experimental tests.
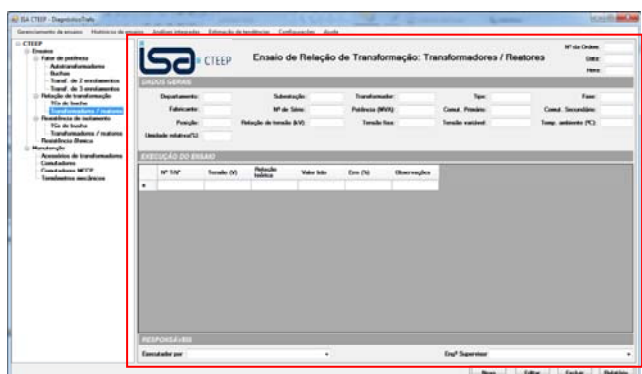
Fig. 14. Main interface of the computational system with emphasis on the viewing area of the transformation ratio experimental tests.
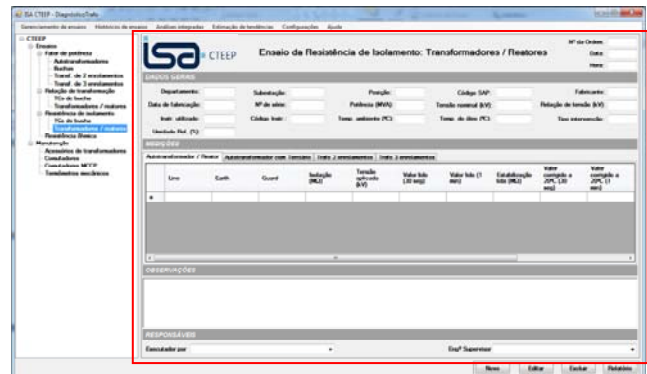
Fig. 15. Main interface of the computational system with emphasis on the viewing area of the insulation resistance experimental tests.

Fig. 16. Main interface of the computational system with emphasis on the viewing area of the Ohmic resistance experimental tests.

Some characteristics of the computer system proposed should be highlighted in relation to aspects of the graphical interface, which are connected to the possibilities of handling by the user. Among these characteristics include the following:

- management of experimental tests;
- historical experimental tests;
- integrated analyzes;
- estimation of tendencies;
- settings of the expert system.

The ability to manage the experimental tests is provided so that the user can register new experiments, change existing experimental tests, exclude experimental tests, conduct planning (standard step) and check the procedural consistency of experimental tests (Figure 17).
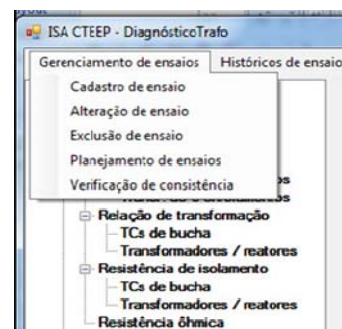
Fig. 17. Management menu of experimental tests.

The history of experimental tests is now done important for the estimation of possible failures of transformers is perceived by the expert system (Figure 18).
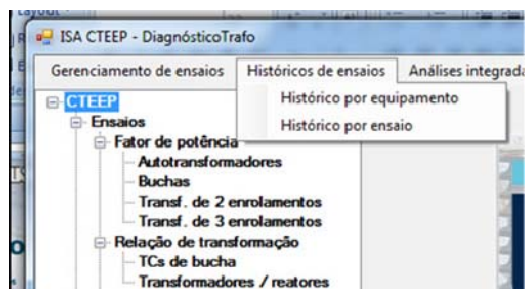
Fig. 18. History menu for experimental tests.

Furthermore, there is the possibility of performing integrated analyzes, which constitutes a fundamental part of the software, since it is by this menu that any experimental tests data will be integrated so that a solution is provided on the condition of the equipment that is under analysis.

It is noteworthy that this menu will allow an analysis of the equipment, employing both current data from experimental testing as their historical data.

Finally, as a major feature of this interface, there is tendency estimation, which is responsible for analyzing the data from all tests has been conducted to the device and then provide an estimation of its useful life

These last two characteristics can be seen by Figure 19, which shows the graphical interface highlighting these tasks.
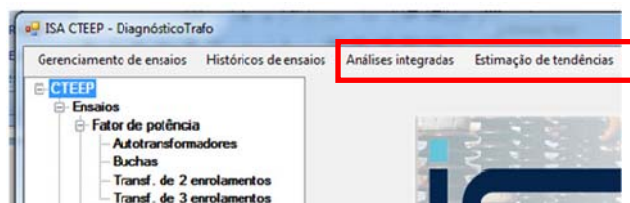
Fig. 19. Menu of integrated analyzes and estimation of tendencies.

4 Conclusions

Due to the high cost of both acquisition and maintenance of power transmission transformer, the diagnosis of such equipment becomes thus imperative. Therefore, all tests and analyzes proposed in this research project are required in order, as well as the integration of data obtained for each of the tests. Thus, we note the importance of a system to provide the correct storage and data analysis and integration, which makes the transformer diagnosis even more accurate.

5 Acknowledgements

6 References

[1] X. Wang, B. Li, H. T. Roman, O. L. Russo, K. Chin and K. R. Farmer, "Acousto-optical PD Detection for Transformers", *IEEE Trans. on Power Delivery*, vol. 21, pp. 1068-1073, 2006.

[2] S. M. Markalous, S. Tenbohlen and K. Feser, "Detection and Location of Partial Discharges in Power Transformers using Acoustic and Electromagnetic Signals", *IEEE Trans. on Dielectrics and Electrical Insulation*, vol. 15, no. 6, pp. 1576-1583, 2008.

[3] E. Mohammadi, M. Niroomand, M. Rezaeian and Z. Amini, "Partial Discharge Localization and Classification Using Acoustic Emission Analysis in Power Transformer", *31st International Telecommunications Energy Conference*, 2009.

[4] G. F. C. Veloso, L. E. B. Silva, G. Lambert-Torres and J. O. P. Pinto, "Localization of Partial Discharges in Transformers by the Analysis of the Acoustic Emission", *IEEE International Symposium on Industrial Electronics*, 2006.

[5] J. R. Guardado, J. L. Naredo, P. Moreno e C. R. Fuerte, "A comparative study of neural network efficiency in power transformers diagnosis using dissolved gas analysis", *IEEE Transactions on Power Delivery*, vol. 16, no. 4, pp. 643-647, 2001.

[6] M. Duval, "A Review of faults detectable by gas-in-oil analysis in transformers", *IEEE Electrical Insulation Magazine*, vol. 18, pp. 8-17, 2002.

[7] M. M. Imamura, "Uma Abordagem Neural Para Identificação de Gases Dissolvidos em Óleo Isolante de Transformadores em Função de Parâmetros Físico-Químicos", *Dissertação de Mestrado em Engenharia Industrial*, Faculdade de Engenharia, Universidade Estadual Paulista, Bauru, 2000.

[8] M. Milasch, "Manutenção de Transformadores em Líquido Isolante", *Editora Edgar Blucher Ltda*, 1984.

[9] H. Herman, M. J. Shenton, G. C. Stevens e R. J. Heywood, "A new approach to condition assessment and lifetime prediction of paper and oil used as transformer insulation", *Proceedings of 7th International Conference on Solid Dielectrics*, pp. 473-476, 2001.

[10] S. P. Zhu, J. Shi, Y. S. Lim, C. I. Tan e Y. K. Cheong, "Transformer Testing and Analysis with a Window-based Graphic User Interface", *International Conference on Energy Management and Power Delivery*, pp. 410-415, 1995.

[11] A. J. Kachler, "On-Site Diagnosis of Power and Special Transformers", *IEEE International Symposium on Electrical Insulation*, pp. 362-367, 2000.

[12] H. Mohseni, A. S. Akmal, J. Jadidian, E. Hashemi, E. Agheb, A. Naieny, A. Ghorbandaeipour e K. Niayesh, "Induced Voltage Test of Distribution Transformers Using State Frequency Inverters", *IEEE/PES Transmission & Distribution Conference and Exposition*, pp. 1-5, 2008.

[13] IEEE Std. C57.12.90-1993, "IEEE Standard Test Code for Liquid-Immersed Distribution, Power, and Regulating Transformers and IEEE Guide for Short-Circuit Testing of Distribution and Power Transformers", 1993.

[14] IEEE Std. C57.12.00-2006 for Standard General Requirements for Liquid-Immersed Distribution, Power, and Regulating Transformers, "IEEE Standard for Standard General Requirements for Liquid-Immersed Distribution, Power, and Regulating Transformers", 2007.

[15] H. Liu, H. Motoda, "Feature Selection for Knowledge Discovery and Data Mining", Kluwer Academic Publishers, 1998.

[16] R. Kohavi, G.H. John, "Wrappers for feature subset selection", *Artificial Intelligence*, no. 97, pp. 273-324, 1997.

[17] H. Almuallim, T.G. Dietterich, "Learning with many irrelevant features", *Proceedings of the 9th National Conference on Artificial Intelligence*, pp. 547-552, 1991.

[18] H. Liu, R. Setiono, "A probabilistic approach to feature selection: a filter solution", *Proceedings of the 13th International Conference on Machine Learning*, pp. 319-327, 1996.

[19] M.A. Hall, G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining", *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 3, pp. 1437-1447, 2003.

Using Computational Vision and Artificial Neural Networks in Electrical Insulator Preventive Maintenance

I.N. Silva[1], R.A. Flauzino[1], V.H. Mustacio[2], S. Vallin[2], and J.F.R. Silva[2]

[1]University of São Paulo, USP/EESC/SEL, CP 359, 3566-590, São Carlos, SP, Brazil
[2]ELEKTRO – Electricity and Services S/A, Rua Ary Antenor de Souza, 321, 13053-024, Campinas, SP, Brazil

Abstract—*This paper presents a neural approach aiming classification of images of electrical insulators in order to perform preventive maintenance. One of the main motivations of this research is the verification that there is an appropriate moment for cleaning of power distribution system insulators, which allows to perform decontamination without compromising its functionality. Additionally, from the investigations reported by the related literature, another important motivation is the lack of products or alternatives that can be efficiently implemented. With the development of the proposed approach, there is then the possibility to identify and delineate only those areas where there is need to perform washing maintenance, because it was also witnessed the existence of points that were washed without due need. Thus, these actions aim to optimize the resources dedicated to this preventive maintenance, contributing both to define those points that need maintenance as well as to avoid it at points that are still with satisfactory quality.*

Keywords: *image processing; artificial neural networks; power distribution systems; electrical insulators; preventive maintenance.*

1 Introduction

One of the most common causes of outages, both in transmission and distribution systems of electricity, are the leakage currents in insulators, which are more prevalent in areas with industrial pollution, or very saline atmosphere, such as that found in coastal regions. The most currently employed electrical insulators are porcelain base with the coating of a continuous thin layer of vitreous enamel, which contributes to the improvement of the mechanical and electrical properties.

A drawback of these insulators is that their surfaces are made of metal oxides that have high energy or high wettability. This characteristic can lead to the formation of water film which, if continued throughout the height of the insulator, is able to conduct current and closing an electric arc, whose phenomenon is known as short circuit. This problem is aggravated by the increasing environmental pollution and the increasing concentration of industries on the coast, because the pollution and the salts will be deposited (with the passage of time) on the insulator surface until reaching a threshold concentration. As a consequence, in rain or fog, this continuous film can become conductive and cause a short circuit.

Regarding the coast of Brazil, the practice of the insulators washing in order to remove this contaminant film is the most employed here due to its simplicity and efficiency. However, according to the aspects outlined in the literature correlated, the materials that constitute the insulator of ceramic type tend to accumulate water, and the practice of periodic cleaning (in time periods not suitable) may facilitate the appearance of leakage currents.

In fact, overhead power circuits that operate in polluted areas are subject to failures which are caused by the degradation of equipment, and the insulator is one of the components most affected. Indeed, electric arcs caused by the reduction of its performance produce a large leakage current flowing on the surface, which contributes to permanently damage the device [1]-[4].

Due to the growing demand for electricity, the transmission and distribution systems are also being enhanced in recent years, which means a greater amount of insulators present in electrical networks [5]. As an example, in [6], the authors emphasize the importance of this device to the current power system, especially the ceramic insulators due to their good mechanical and electrical characteristics, and also their affordable costs.

In power transmission systems, the amount of insulators, associated with their separation distances, has allowed them to support (before having its functionality compromised) a reasonable amount of electric arcs on their surfaces. However, these characteristics are not already so intrinsic in power distribution systems, where defects occur more frequently, especially in coastal regions [1].

Several solutions have been adopted to minimize this problem, highlighting three important lines of action. The first is the periodic inspection and cleaning of insulators aiming removal of deposits. Another line of research is one that aims to improve the constructive aspects. Finally, there is also the use of tools to estimate climatic conditions favorable for the emergence of electric arcs. However, studies that address the appropriate time involving the cleaning of insulators are not found in the literature related to the power distribution system.

In the research field of new construction aspects are highlighted those based on changing the geometry of the

insulator in order to impede the leakage currents, as well as to facilitate the natural cleaning of pollutants on its surface by the action of wind and rain, or in changing the material used in the surface coating [7,8].

Polymeric insulators are also gaining space in place of ceramic, especially in the U.S., where such devices have been used in power transmission systems for over 30 years. Its disadvantage lies in its weak resistance against degradative effects caused by sunlight and inclement weather [9], where such characteristics can not be disregarded in the coastal regions of Brazil.

In this same line of research highlight those hybrid insulators, usually porcelain coated with silicon films, or even those insulators arising from processes that enable the application of polymeric materials capable of providing a large electrical performance against contaminants [7,8,10].

But the electric arc is probably one of the phenomena associated with more severe failures of insulators, causing here a very negative impact on power quality. Several researchers have investigated this phenomenon for many years, in order to then search for correlation model relating the voltage levels, the number of insulators and density of pollution, which aim to map the processes that are responsible for causing the electric arc on the surface of these devices [11].

Another line of research that aims to mitigate the problems that result in electric arcs on insulators consists of estimating the conditions that allow to identify the levels of pollutants, whether arising from saline contamination or industries; capable of compromising the operation of this device, being still of fundamental importance to know the electrical processes involved with the formation of arcs from the leakage currents.

In this context, there is a need to understand the behavior of the leakage current in relation to weather conditions and also in cases of pollution, especially those investigations that attempt to characterize the signatures of the electrical signals of the arcs on the surface of insulators, such as those presented in [12-14].

As an example, in [15], the authors perform a complete treatment involving the parameter estimation of contaminants through the linear regression technique, which demonstrate all aspects of making the estimation model of saline conditions. This model allows to determine how severe the contamination is, and the best maintenance interval of insulators. The major contribution of this proposal is the fact that the negative coefficients obtained for the linear estimation model show that the concentration of salt deposited on the insulator increases with the decrease of the input variables. The variables considered as inputs were temperature, humidity, wind direction and rainfall.

Improvements in the estimation procedures of pollutants can also be achieved with the adoption of intelligent tools, highlighting especially those based on artificial neural networks and fuzzy inference systems, such as the proposals presented in [16-19]. These tools are promising in solving this

kind of problems, because they can be easily condensed into compact computation algorithms, and embedded in digital hardware after finalizing their learning processes. It is also highlighted here the good results obtained in this work due to the conditions analyzed. However, it should also note that all of them were evaluated for conditions involving power transmission systems; consequently, significant investigations are then necessary to adapt these proposals to the power distribution systems.

Thus, from images of electrical insulators, which may be collected by UAV's (Unmanned Aerial Vehicles), the goal of this research is then to propose an intelligent system that performs classification of such images in order to indicate if it is already in appropriate time to carry out cleaning (washing) of insulators. Therefore, the tool developed can be of great value for the purposes of preventive maintenance, thus avoiding unscheduled interruptions in the power distribution system, or by preventing that maintenance procedures are performed without due need.

2 Image processing of insulators for features extraction

2.1 Images employed for identification of deposited soot

We analyzed a total of 488 images of insulators, and 181 of these were identified as having soot deposited on its surface. The remaining images of the insulators were considered suitable for operation. In order to illustrate images, differentiating insulators with (and without) soot deposited, a set of images of insulators considered suitable for working are shown by Figure 1, while Figure 2 illustrates how the soot (contaminant) is presented visually.

Fig. 1. Insulators considered suitable for working.

Fig. 2. Insulators presenting soot deposited on the surface

2.2 Preprocessing for contrast enhancement

Preprocessing of images aims to improve its characteristics so that the identification process of insulators covered with soot can be improved. Since they are images taken in daylight, there is the existence of regions with large contrast gradient between them, which takes place in deficit of contrast on surfaces under the same exposure, and also in large gradients of contrast in objects whose faces are exposed at different angles relative to the light source.

Depending on the geometry of the insulators, the presence of significant contrast gradients in their respective images affects the distinction of colors, which may result in identification less than expected compared to those with soot deposited on its surface.

To overcome this limitation was then adopted the contrast adaptive equalization [20]. This algorithm operates on small regions, denoted by tiles, instead of employing the image as a whole. The contrast of each tile is equalized so that it comes close to a normal distribution. The tiles adjacent to one that is equalized are combined by employing a bilinear interpolation to avoid the formation of artificial boundaries.

Figure 3 illustrates how the improvement of image contrast makes the colors are better discriminated, and the location of soot region is more easily discriminated.

Fig. 3. Example of application of contrast adaptive equalization: (a) Original image (b) Image with contrast equalized.

2.3 Process for obtaining discriminant features

Through images of insulators presented earlier is possible to see some particularities that make this problem complex and unconventional solution. Among these peculiarities, one, perhaps the major, refers to small discrepancies between the color of the ceramic insulator with respect to the soot which overlaps on its surface. This aspect was one of the motivators for the adoption of pre-processing technique presented in [20].

Another feature of this problem is the size of the images. Since it is a region cropped from a larger image, then there is no guarantee that the dimensions of the images are identical. Thus, a tool for extracting features related to the existence of soot deposited on insulators shall be insensitive or minimally robust to these differences. Pondering these aspects, it was decided here by using the normalized histogram of the RGB channels of the image in order to extract the features.

The average difference between the histograms for each color channel of the insulator images can be analyzed by Figure 4 through Figure 6. By visual inspection of the histograms presented in Figure 5 and Figure 6 observes that the channels G and B have more data to differentiate between positive and negative images to identify the soot over the R channel
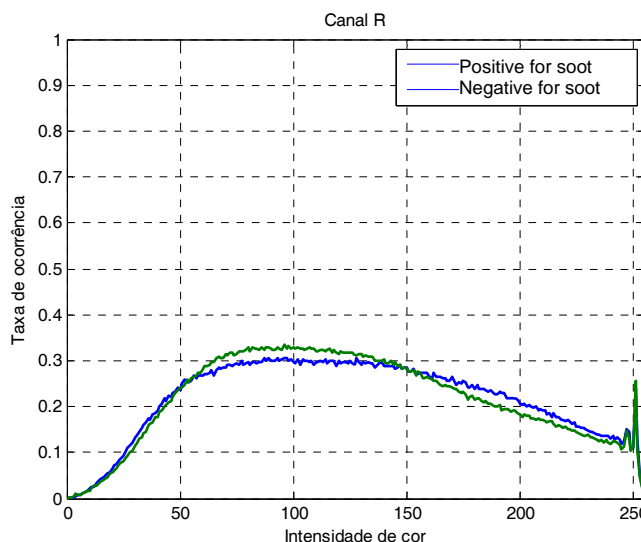
Fig. 4. Histogram of the R channel for the images analyzed.
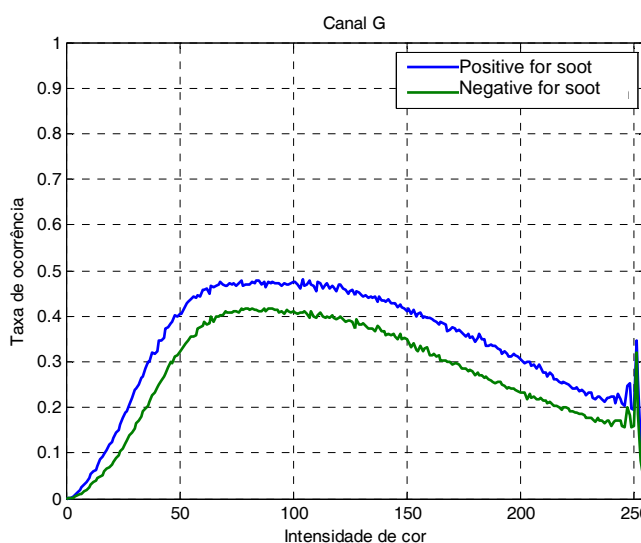
Fig. 5. Histogram of the G channel for the images analyzed.
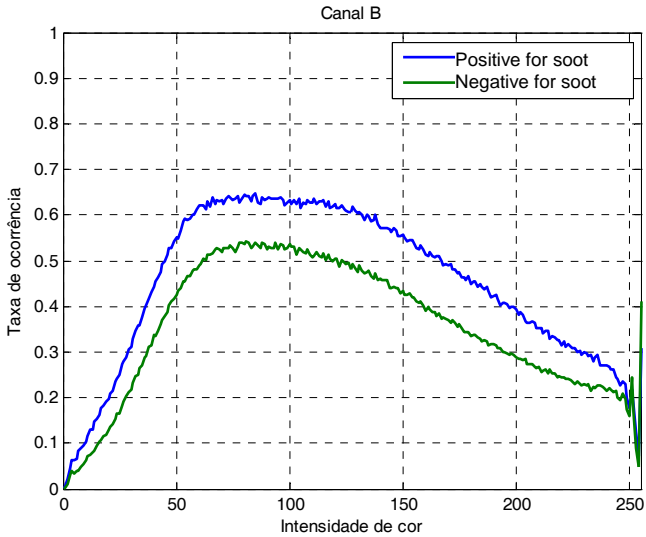
Fig. 6. Histogram of the B channel for the images analyzed.

The use of the histogram of RGB channels circumvents the problem of dimensional non-uniformity of the images to be analyzed, but another arises, which is the amount of characteristics, i.e., three channels (each with 256 levels for the color histogram) result in 768 features.

This impressive number of features motivates the adoption of dimensionality reduction techniques. To perform this task, the use of the technique of Principal Components Analysis (PCA) was here chosen. For each RGB channel, two components were then adopted, and the input space from 768 features was now reduced to six, but they are better correlated with the classes to be identified than the original components individually observed.

By means of Figure 7 are presented in graphic form those coefficients which are multiplied by the R channel characteristics, which result in the two main components adopted for that channel. In Figure 8 and Figure 9 the same is done for channels G and B.
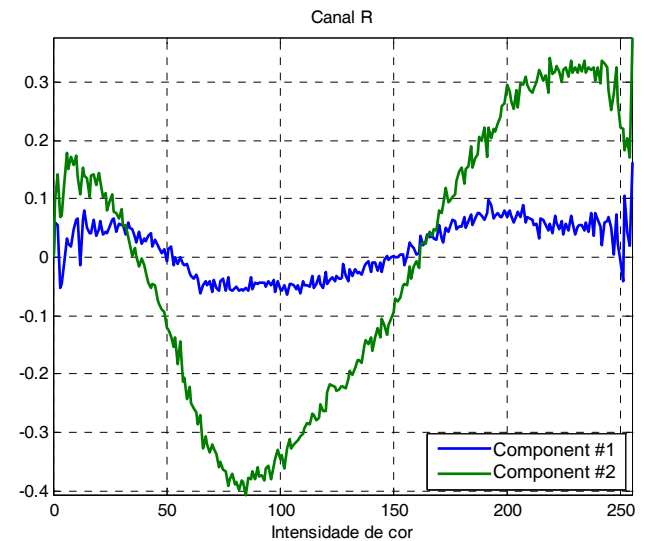
Fig. 7. Linear coefficients for determination of the main components of the R channel.
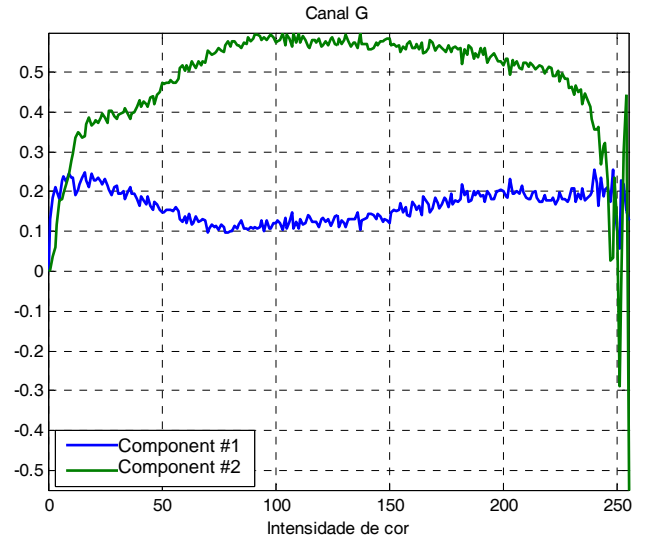
Fig. 8. Linear coefficients for determination of the main components of the G channel.
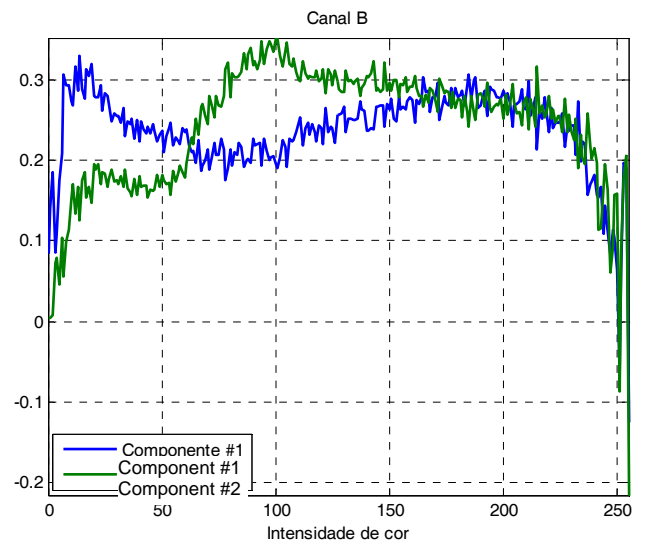
Fig. 9. Linear coefficients for determination of the main components of the B channel.

The linear correlation coefficient between the original features of the histograms obtained from RGB channels and identification of soot in insulators has absolute average equal to 0.148. The absolute average of correction between the main components adopted and identification of soot is 0.224. The increase of approximately 50% on average correlation helps to demonstrate the effectiveness of the technique to reduce the dimension of the problem.

3 Design of artificial neural network for identification of insulation with soot

Based on the histogram of the major components of the RGB channels, a neural network of multilayer-perceptron type was then designed for identifying soot in insulators, as illustrated in Figure 10. The inputs of the network are then formed by 6 inputs, i.e.:

➢ C#1(R) is the component #1 in relation to R channel,

➢ C#2(R) is the component #2 in relation to R channel,

➢ C#1(G) is the component #1 in relation to G channel,

➢ C#2(G) is the component #2 in relation to G channel,

➢ C#1(B) is the component #1 in relation to B channel,

➢ C#2(B) is the component #2 in relation to G channel.
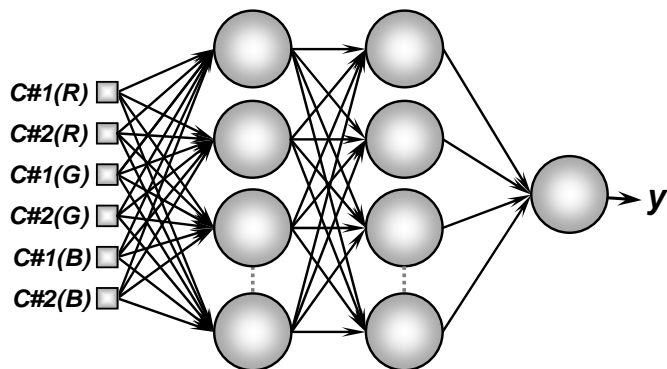
Fig. 10. Multilayer perceptron network topology for identifying soot in power distribution system insulators.

Computational implementations performed with perceptron networks of different topologies have led to the choice of a configuration with two hidden layers [21]. The first hidden layer was composed of 12 neurons and the second layer of 10 neurons. In both layers have neurons with activation function of hyperbolic-tangent type. The final layer, composed of a single neuron (y), makes the linear combination of the outputs of the neurons of the second hidden layer, and the response value of 1 indicates the need for washing of the insulator, while the value 0 indicates the insulator is further under conditions suitable for operation.

The training algorithm adopted was one based on nonlinear optimization technique of Levenberg-Maquardt [22] and the behavior of the mean squared error over the training epochs is shown in Figure 11.
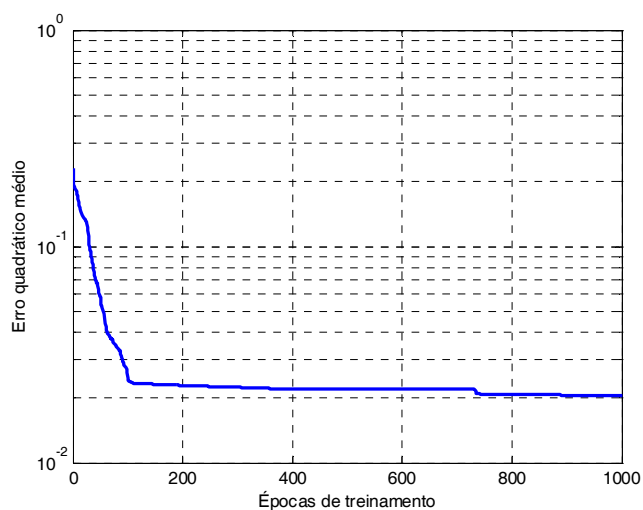
Fig. 11. Behavior of the mean squared error over the training epochs.

A histogram of the absolute error on the soot indication is provided in the graph of Figure 12.
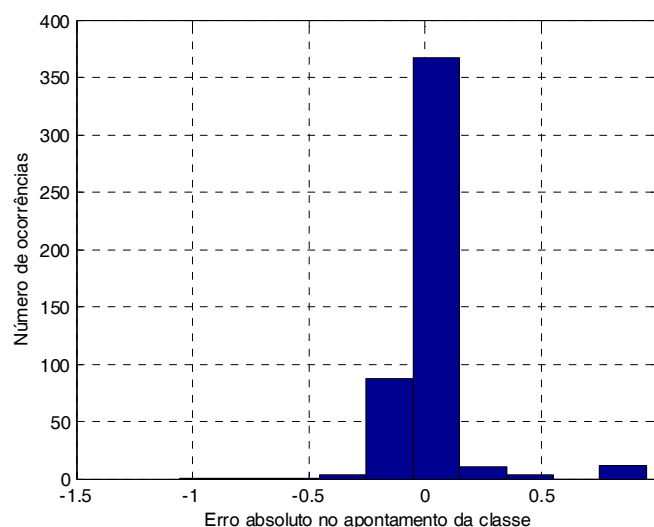
Fig. 12. Histogram of the absolute error for identification of soot.

Through Figure 12 one can observe that the estimation error has an approximately normal distribution, with mean centered in the vicinity of zero. Furthermore, it has a high concentration of occurrences around intervals that are neighbors to zero. The results achieved for pattern classification showed a rate of not identifying the soot of 2.7% and a false-positive rate of 0.6%. Overall, the success rate was 96.7%.

4 Conclusions

This paper presented an approach based on image processing techniques to identify the condition of electrical insulators present in power distribution systems. From the histogram of the RGB components related to the images of such devices, artificial neural networks were implemented to serve as an expert system for pattern classification. The results provided by this technique developed were fully satisfactory, which can be used as a decision support system to distribution utilities, indicating whether there is already a need (or not) to carry out preventive maintenance procedures (wash) in insulators installed in a given region.

5 Acknowledgements

6 References

[1]   C. A. I. Fontanella, A. P. N. Oliveira, D. Hotza, "Recobrimentos poliméricos hidrofóbicos sobre isoladores elétricos de porcelana," Matéria, vol.13, no. 4, pp. 624-635, 2008.

[2]   A. L. Chinelatto, D. P. F. Souza, "Porcelanas elétricas aluminosas: Parte I - Revisão da literatura," Cerâmica, vol. 50, no. 313, pp. 62-68, 2004.

[3]   A. L. Chinelatto, D. P. F. Souza, "Porcelanas elétricas aluminosas: Parte II - Desenvolvimento da microestrutura e sua influência no módulo de ruptura," Cerâmica, vol. 50, no. 315, pp. 172-184, 2004.

[4]   E. O. Abdelaziz, P. Charloman, M. Javoronokov, "Modeling the behavior of the electric insulators in polluted desertic medium," Journal of Engineering and Applied Sciences, vol.1, no. 1, pp. 77-81, 2006.

[5]   K. F. Portella, F. Piazza, P. C. Inone, S. Ribeiro Junior, M. S. Cabussú, D. P. Cerqueira, C. S. S. Chaves, "Efeitos da poluição atmosférica (litorânea e industrial) em isoladores da rede elétrica da região metropolitana de Salvador," Química Nova, vol.31, no. 2, pp. 340-348, 2008.

[6]   I. Swarno, A. Pradana, "Properties of leakage current on 20 kV ceramic insulators and computer simulation based on electrical equivalent circuit," Proceedings of the World Congress on Engineering, vol. II, pp. 1-5, 2010.

[7]   J. Liebermann, "Electric behavior of porcelain insulators in polluted environments," American Ceramic Society Bulletin, vol. 81, no. 6, pp. 31-36, 2002.

[8]   J. Liebermann, "New effective ways toward solving the problem of contamination of porcelain insulators," Refractories and Industrial Ceramics, vol. 43, no. 1-2, pp. 55-64, 2002.

[9]   S. M. Gubanski, A. E. Vlastos, "Wettability of naturally aged silicone and EPDM composite insulators," IEEE Transactions on Power Delivery, vol. 5, no. 3, pp. 1527-1535, 1990.

[10]   S. Kobayashi, Y. Matsuzaki, Y. Arashitani, R. Kimata, "Development of composite insulators for overhead lines (Part 2)," Furukawa Review, no. 21, pp. 56-61, 2002.

[11]   P. T. Tsarabaris, C. G. Karagiannopoulos, P. D. Bourkas, N. J. Theodorou, "An experimental investigation of leakage current on high voltage contaminated insulators," Iranian Journal of Electrical and Computer Engineering, vol. 2, no. 1, pp. 30-34, 2003.

[12]   S. K. A. Suwarno, P. Juniko, "Roles of fog conductivity and humidity on leakage current of ceramic insulators," Journal of Engineering and Applied Sciences, vol. 4, no. 4, pp. 282-287, 2009.

[13]   M. Afendi, M. Piah, A. Darus, "Leakage current analysis of polymer insulating material with variable contaminant flow rate," Australasian Universities Power Engineering Conference, pp. 1-5, 2002.

[14]   M. Amin, S. Amin, M. Ali, "Monitoring of leakage current for composite insulators and electrical devices," Rev. Adv. Mater. Sci., no. 21, pp. 14-34, 2009.

[15]   A. S. Almad, H. Ahmad, M. A. Salam, S. Ahamad, "Regression technique for prediction of salt contamination severity on high voltage insulators," Annual Report Conference on Electrical Insulation and Dielectric Phenomena, vol. 1, pp. 218-221, 2000.

[16]   A. N. Jahromi, A. H. El-hag, S. H. Jayaram, E. A. Cherney, M. Sanaye-Pasand, H. Mohseni, "A neural network based method for leakage current prediction of polymeric insulators," IEEE Transactions on Power Delivery, vol. 21, no. 1, pp. 506-507, 2006.

[17]   J. Li, W. Sima, C. Sun, S. A. Sebo, "Use of leakage currents of insulators to determine the stage characteristics of the flashover process and contamination level prediction," IEEE Transactions on Dielectrics and Electrical Insulation, vol. 17, no. 2, pp. 490-501, 2010.

[18]   X. Jiang, Y. Shi, C. Sun, Z. Zhang, "Evaluating the safety condition of porcelain insulators by the time and frequency characteristics of LC based on artificial pollution tests," IEEE Transactions on Dielectrics and Electrical Insulation, vol. 17, no. 2, pp. 481-489, 2010.

[19]   X. U. Xiaoning, N. I. E. Yixiong, C. Hanxiang, "Application of fuzzy model identification on insulation detection of insulator strings," Asian Journal of Information Technology, vol. 5, no. 1, pp. 24-27, 2006.

[20]   K. Zuiderveld, "Contrast limited adaptive histogram equalization," In: Graphic Gems IV, Academic Press Professional (San Diego), pp. 474-485, 1994.

[21]   I. N. da Silva, D. H. Spatti, R. A. Flauzino, Redes Neurais Artificiais Para Engenharia e Ciências Aplicadas, Editora Artliber (São Paulo), 2010.

[22]   M. T. Hagan, M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," IEEE Transactions on Neural Networks, vol. 5, no. 6, pp. 989-993, 1994.

SESSION

POSTER AND SHORT PAPERS

Chair(s)

TBA

Association inference processing to extract knowledge sentences for question answering

[1]Hirokazu Watabe, [2]Misako Imono, [3]Eriko Yoshimura, and [4]Seiji Tsuchiya
Dept. of Intelligent Information Engineering and Science,
Doshisha University, Kyotanabe, Kyoto, 610-0394, Japan
[1]hwatabe@mail.doshisha.ac.jp, [2] etl1701@mail4.doshisha.ac.jp, [3] sk109716@mail.doshisha.ac.jp,
stsuchiy@mail.doshisha.ac.jp

Abstract - *Much of everyday conversation consists of answering questions, with the respondent commonly exhibiting an innate ability to extract appropriate answers from a vast volume of mentally stored knowledge. In computer-based question-answering systems, knowledge extraction has generally involved searching knowledge representations organized by predicate logic or production rules (if-then rules), to find objects of inference. As it is based on symbol processing, however, it is inherently dependent on word notation. In the present paper, we propose a flexible knowledge extraction method in which words having the same meaning but different notations can become objects of inference through incorporation of an association system[1,2] which comprises a Concept-Base that utilizes a method for calculating the degree of association.*

Keywords: knowledge extraction, degree of association, Concept-Base

1. Techniques
1.1 Common knowledge

In this study, common knowledge is defined as scientific knowledge commonly acquired during compulsory education, such as the knowledge that HCl (i.e., hydrochloric acid) is acidic, rather than its more general definition of commonly acquired linguistic and social knowledge. It is maintained in the following three configurations.

(1) Common-knowledge base

The common-knowledge base consists of knowledge stored in the form of "knowledge sentences" and word strings (including compound terms) that are subjected to morphological analysis, as illustrated in Table 1. In the present study, the base contains 1,622 such sentences relating to science.

Table 1. Common-knowledge base entry (science).

Knowledge sentence	Word strings
HCl is acidic.	HCl, acidic

(2) Common-concept base

The common-concept base consists of common knowledge added to an existing concept base[2], having the structure illustrated in Fig. 1.
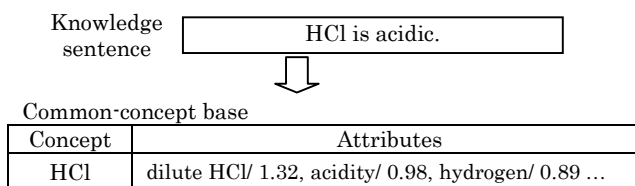
Fig. 1. Common-concept base structure.

In the construction of the common-concept base, the common term "HCl" is thus incorporated as a concept, which enables its use in calculating degrees of association[1].

(3) Common thesaurus

The common thesaurus comprises knowledge that systemizes inclusive and hierarchical relations that cannot be expressed in the common-knowledge base or the common-concept base. The scientific knowledge in the present study is systemized with 190 nodes and 1,462 leaves.

1.2 Question-sentence comprehension system

The question-sentence comprehension system acquires the "question-object" terms (the words that express the answer sought by the question sentence) and their conditions (the conditions placed on the question-object terms) from the question sentence. More specifically, a parsing tool is used to extract the question-object terms and their conditions by combining the words involved in the "?" and interrogative terms in the question sentence and their positional relations.

2 Knowledge-sentence extraction for question answering
2.1 Preconditions

The question sentences treated in the present study were such that they could be answered with term units (e.g., "What (color) is…?") and were moreover limited to those for which the answers could be obtained from one or two knowledge sentences.

2.2 Question-sentence input

In the following description of the proposed processing, the question "To what color does HCl change litmus paper?" will serve as a typical example of the question-sentence input. It is further assumed that the common-knowledge base includes the

knowledge α that "Dilute HCl, carbonated water, and boric acid solution are acidic aqueous solutions" and β that "Acidic aqueous solutions turn litmus paper red," which are necessary for the question answering.

2.3 Question-sentence analysis

Essential information concerning the question sentence is obtained by question-sentence analysis, using morphological analysis and framing as illustrated in Fig. 2. The framing information is used in association-inference processing for inference of multiple knowledge problems, and the information termed A and C in the syllogism is extracted from the question sentence. In the syllogism A⇒B⇒C, B can serve as both a subject and a predicate, and separation of the question sentence into its subject and predicate may therefore be considered effective. On this basis, the question sentence is divided into condition A as the subject and answer-expression C as the predicate.

Q: To what color does HCl turn litmus paper?
Morphological analysis: HCl, litmus paper, color, change

Condition A	Answer-expression C
HCl	litmus paper, color, change

Fig. 2. Typical question-sentence analysis.

2.4 Question typing

The question typing consists of assessing whether it presents a single or a multiple knowledge problem. If it is a single knowledge problem, then knowledge satisfying the criteria is extracted at this stage. Otherwise, further processing is performed as follows.

2.5 Association inference processing

(1) Bidirectional inference

Semantic extension (described in (2) below) of the terms of condition A and answer-expression C is followed by a bidirectional notation search, to obtain knowledge groups comprising knowledge α regarding A ("HCl") and knowledge β regarding C ("litmus paper, color, change"). The term "acidic aqueous solution" is common to both directions and is thus taken as B, which semantically links the two pieces of knowledge, and knowledge α and knowledge β are extracted, as illustrated in Fig. 3.

Question sentence: To what color does HCl acid turn litmus paper?

Extracted knowledge: Dilute HCl, carbonated water, and boric acid solution are acidic aqueous solutions.
  Acidic aqueous solutions turn litmus paper red.
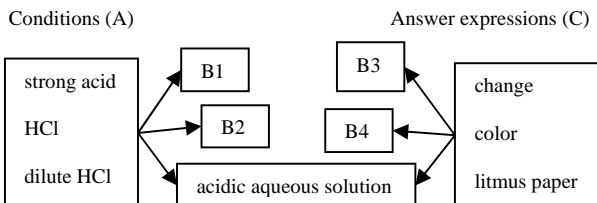
Fig. 3 Inference by bidirectional approach.

(2) Semantic extension of terms

As illustrated in Fig. 3, semantic extension of the term "HCl" is performed to enable inference based on comprehension of its meaning. In this example, "strong acid" and "dilute HCl" are added to the concept "HCl," as their degree of association with "HCl" is at or above a threshold value. In effect, this adds substantial knowledge that is semantically associated with "HCl."

2.6 Extracted knowledge sentence assessment

The extracted knowledge is assessed for its potential validity as part of a correct answer. From the question sentence, we extract the question target terms "color" and its related terms "litmus paper" and "change." They are important terms which represent knowledge that may comprise a correct answer, and for this reason, readily occur in conjunction with the extracted knowledge. Knowledge including these terms is accordingly selected and output. In the example, the knowledge sentence "Acidic aqueous solutions turn litmus paper red," which contains all of the question target terms, is accordingly output.

3. Evaluation

3.1 Evaluation conditions

The proposed method was evaluated using a total of 100 question sentences (50 single knowledge problems and 50 multiple knowledge problems). They were selected for this purpose after confirming that the knowledge necessary to answer them was present in the common knowledge base.

3.2 Evaluation results

The overall ratio of correct answer occurrence among the top three scoring positions was 74%. The accuracy was relatively high for the 50 single knowledge problems, but just 68% were in the top three scoring positions and thus rather low for the 50 multiple knowledge problems.

4. Conclusion

The present study has shown the effectiveness of the method of knowledge extraction using association inference. Further improvement of the semantic extension method and refinement of the range of semantic extension may be expected to yield more human-like and more correct question answering.

References

[1] H. Watabe and T. Kawaoka: "The Degree of Association between Concepts using the Chain of Concepts", Proc. of SMC2001, pp.877-881, 2001.
[2] N. Okumura, E. Yoshimura, H. Watabe, and T. Kawaoka: "An Association Method Using Concept-Base", KES 2007/WIRN2007, Part I, LNAI4692, pp.604–611, 2007.

Voting Dynamics

Neelam Gohar

Department of Computer Science, University of Liverpool, Liverpool, United Kingdom

Abstract— *In AI, multi-agent decision problems are of central importance, in which independent agents aggregate their heterogeneous preference orders among all alternatives and the result of this aggregation can be a single alternative, corresponding to the groups' collective decision, or a complete aggregate ranking of all the alternatives. Voting is a general method for aggregating the preferences of multiple agents. An important technical issue that arises is manipulation of voting schemes: a voter may be to make the outcome most favorable to itself (with respect to his own preferences) by reporting his preferences incorrectly. Unfortunately, the Gibbard-Satterthwaites theorem shows that no reasonable voting rule is completely immune to manipulation, recent literature focused on making the voting schemes computationally hard to manipulate. In contrast to most prior work Meir et al. [3] have studied this phenomenon as a dynamic process in which voters may repeatedly alter their reported preferences until the system converges, or else the system goes into a cycle. The dynamics can be extended for other positional scoring rules like k-approval voting, plurality with runoff and non positional scoring rules like Copeland and Bucklin voting system and cycles exists with different types of manipulation moves. In cases of cycles one can leave the cycle by choosing the correct transitions? Where we ca avoid cyles, one could ask further whether random choices are likely to find equilibrium in a short sequence of steps. Convergence to pure Nash equilibria in plurality voting games can be studied under unweighted setting. We described the concept of tactical voting that is different from manipulative voting and results for the termination of tactical vote dynamics in case of positional scoring rules can be obtained.*

Keywords: Voting protocols, Game Theory (non-cooperative), Modeling the dynamics of MAS

1. Introduction

Sincere voting is voting in accordance with one's true preferences over alternatives. While, Strategic voting is voting over assumed outcomes, in which a voter uses skills to determine an action that secures a best possible outcome in his view. Strategic voting under Plurality rule refers to a voter deserting a more preferred candidate with a poor chance of winning for a less preferred candidate with a better chance of winning [4]. The logic of tactical/strategic voting, of course, is that of Duverger's law, which states that the supporters of a small party would not waste their votes by voting for their most preferred party (candidate) because it does not have a chance to win under a Plurality system with single member districts. Instead, they vote for the major party that is most acceptable to them and that has a chance of winning. Let us suppose a voter believes that her most preferred candidate has little chance of competing for the lead in the election. Voting for such a candidate may be a waste. The voter may decide to switch her vote to the expected leading candidate she most prefers in order to make her vote pivotal in determining a more preferable outcome. This is the trade-off a rational voter faces in an election. Strategic voting is an important component of Duverger's Law, if voters are rational, they end up voting for one of the two leading candidates [5]. The model of tactical voting is a system of states and transitions. Voters have true preferences and declared preferences which can change. A transition occurs from current state to a new state when a voter changes his mind and chooses a different candidate to support. In a state of a system, each voter determines whether it can improve the outcome by altering its own vote while assuming that all other votes remain the same. This model is different than the manipulation dynamics because it simply raises the votes of an expected leading candidate she most prefers. We wanted to capture the behavior (noted in political elections) that a voter switches to another candidate that becomes "more likely to win" as a result, assuming that the voter's favorite candidate was "very unlikely" to win. There are various ways one might try to formalize this behavior. A voter can switch his support to a different according to the following rule: voters consider current state, now each bloc or a single voter determine whether it can improve the outcome by altering its own vote while assuming that all other votes remain the same. In a state of a system, consider all alternative candidates that a voter ranked higher than the current winner. A voter or bloc of voters can then change his support to that alternative candidate who has currently most votes, breaking ties in favor of his own preference. With this mind changing, transition occurs and system enters into a new state. If no voter or bloc can improve the outcome, the current situation is Nash equilibrium. Tactical voting is different than the manipulation dynamics because it simply raises the votes of an expected leading candidate he most prefers. In this kind of voting a voter instead of wasting vote by voting for his most preferred candidate who does not have a chance to win, it's better to vote for a candidate and raise the score of that alternative who is more acceptable to him and has a chance of winning. The sequences of votes can

be analyzed that may result from various voters performing tactical vote.

Definition 1: (Declared preference) A declared preference is the vote that a voter submits to the social choice function in use.

Definition 2: Assume we are using *Plurality*. Fix a state of the system. A *bloc* is a (maximum sized) set of voters who all support the same candidate w.r.t. declared preferences. However, voters belonging to the same bloc may or may not be *like minded*.

Definition 3: (Transition in case of like minded voters under plurality rule for tactical voting). Fix a state S of the system in which a set of like-minded voters $V \in \mathcal{V}$ currently support candidate $j \in \mathcal{A}$. The system can make a transition from current state S to S', if for set of voters V there is a candidate j' such that $j' \succ_V^S j$ and $\mathcal{A}'$ is the subset of $\mathcal{A}$ such that $\mathcal{A}' \succ_V w(S)$, $j' \succ_S \mathcal{A}'$.

2. Significance

This analysis is of particular interest to artificial intelligence as it tackles the problem of multi-agent decision making, where agents are considered to be autonomous entities and they have to a joint plan of action or allocate resources. The study of dynamics in strategic voting is interesting as it helps to understand, control and design mutil-agent decision-making processes.

References

[1] Robert Forsythe, Thomas Rietz, Roger Myerson and Robert Weber, *An experimental work of voting rules and polls in three-candidate elections*, International Journal of Game Theory, 25, 3, 335–383, 1996

[2] Stephane Airiau and Ulle Endriss, *Iterated Majority Voting*. Proceedings of ADT, 2009, 38-49, Springer-Verlag LNAI 5783

[3] R. Meir and M. Polukarov and J. S. Rosenschein and N. R. Jennings, *Convergence to Equilibria in Plurality Voting*. Proceedings of 24th Conference on Artificial Intelligence (AAAI-10), 2010

[4] R. Farquharson, *Theory of Voting*. Yale University Press, New Haven, 1969

[5] A. Bassi *Voting Systems and Strategic Manipulation: an Experimental Study*. Paper presented at the annual meeting of the The Midwest Political Science Association, Palmer House Hilton, Chicago, Illinois, 2006

Preliminary Screening of Graduate School Applicants Using Fuzzy Inference System

T. Sri Kalyan[1], Devarsh Bhonde[1], Hari Sai Krishna Kanth[1] and Pushpal Mazumder[1]
[1]Department of Civil Engineering, Indian Institute of Technology Kharagpur, West Bengal, India

Abstract - *In this paper a fuzzy inference system is proposed as a preliminary screening tool for graduate school applications, in an attempt to make the screening process more efficient and accurate. The system uses important aspects of an application to provide an advisory on whether the application should be considered for further review. The system provides a choice to define weightages for different criteria as per the requirement of the individual graduate school or to use a general model. The fuzzy inference system conveniently establishes relation between multiple input parameters to less output parameters. Thus the fuzzy inference system facilitates an efficient application screening process, free from personal bias of the admissions committee.*

Keywords: Fuzzy inference system; Graduate school applications; Preliminary screening

1   Introduction

The graduate schools receive applications from many students with very high qualifications. A competitive program in some graduate school may also receive upto 500 applications. The current review system adopted by the admission committees in most of the schools is a time consuming and tedious process. It generally is a three/four step process. An initial review is done to check whether the application submitted is complete, followed by a first pass to identify those applications with promise. The applications then undergo a batch review, where sets of applications are sent to 2-3 faculty members for evaluation based on motivation, experience and overall promise. The faculty members meet to make final decision on the admissions. The outcome of this review process can be highly dependent on the personal bias of the admissions committee members.

Hence a fuzzy inference system is proposed in this paper in order to make the preliminary application screening process simpler by providing an advisory on which application should be shortlisted. All the important aspects of an application such as the number of publications, professor recommendation, parent institute rating, grade point average (GPA), GRE score and work experience have been taken into consideration to suggest whether the application should be considered for further review or not. The system also suggests whether the decision on the application should be made based on other factors such as the statement of purpose, motivation and overall promise of the applicant.

2   Fuzzy rule-based approach

Each application has its individual characteristic for e.g. an applicant may have very high GPA but no publications whereas other applicant may have many publications but a comparatively low GRE score. In order to compare independent applications the fuzzy inference system is adopted to make the decision making process easier. Fuzzy based system presents a convenient way for representing the relation between multiple input parameters to less output parameters by means of verbal statements instead of complex mathematical equations. It helps to express the qualitative aspects of parameters in a quantitative way. Fuzzy inference system is used in this paper to suggest whether the application should be shortlisted, waitlisted or rejected after processing the inputs for different parameters on the basis of fuzzy rules, as shown in Fig. 1.

2.1   Evaluating Parameters:

To account for the influence and contribution in governing the status of the application, the following fuzzy logic variables are used for evaluating the graduate applications:

1. Grade Point Average (GPA) of the student.
2. Graduate Record Examination (GRE) Score
3. Professor Recommendation Rating
4. Number of Publications
5. Parent Institute Rating
6. Work Experience
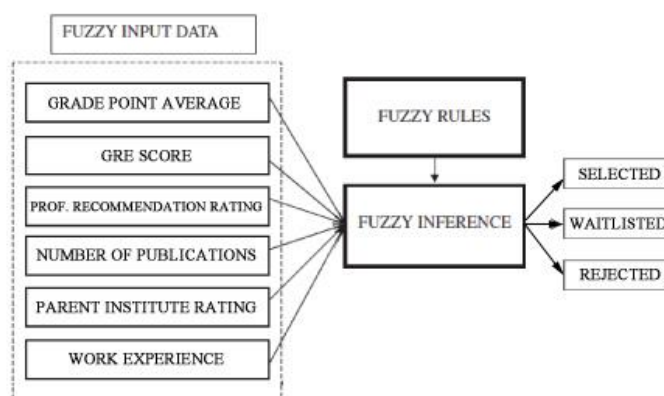
Fig. 1. Fuzzy logic model components

2.2 Input variables description

The detailed explanation of each parameter used in the fuzzy logic modeling is explained in this section. The membership function for each parameter can be seen in Fig. 2.

2.2.1 Grade Point Average (GPA) of student

In evaluating the graduate applications, one of the most important and crucial variable which decides the fate of the application is the GPA of the student. Since there is a strong correlation between the GPA and the application getting selected, three different classes namely bad, average and good along with their corresponding membership functions are defined. It is assumed that the GPA is on the scale of 10 and the membership functions are defined accordingly.

2.2.2 Graduate Record Examination (GRE) Score

The GRE score of the applicant is also given a relatively high importance while evaluating the application. Three classes are defined to represent this parameter namely bad score, average score and good score. The actual GRE score ranges from 260-340, but the universities might ask the students to either retake the test or reject them if the scores are as low as 300. So, a minimum score of 300 has been assumed in this model.

2.2.3 Professor Recommendation Rating

On observation of the recent trends in the graduate selection process it is found that the recommendations from different professors or mentors play a key role on the decision regarding the application. Hence it is decided to include this parameter as one of the governing inputs besides other parameters which are mostly related to applicant's academic performance. It is assumed that the rating is done on a scale of 0 to 1 and three classes namely good, average and bad are defined.

2.2.4 Number of Publications

The number of publications in an applicant's profile shows the interest of the student in their respective specializations. The publications are representative of the applicant's performance in the field of research. Hence a student with a reasonable number of publications is preferred over those with no publications. Thus two different classes are proposed to represent those applications with and without any publications.

2.2.5 Parent Institute Rating

The reputation of the applicant's parent institute should also to be taken into consideration while evaluating the application since it represents the level of the competition and in-depth exposure to their respective fields. To account for this parameter, three classes: normal institute, good Institute and prestigious institute are defined.

2.2.6 Work Experience

An applicant having some work experience in his/her respective field has a more practical understanding of the subject. Hence an applicant having some work experience has an additional advantage. Thus work experience is also considered as an input parameter for evaluation with the range of 0 to 4 years and the respective classes are defined as: no experience, experienced and well experienced.

3 Rule base for fuzzy inference system

Based on the six input parameters defined in the previous section, set of 180 rules are defined to evaluate the application. Sample of 20 rules are tabulated and presented in the Table 1.

4 Evaluating Criteria

The membership value of each rule is computed and the output class corresponding to the rule with maximum membership value is given as the result. The result is represented by three classes: selected for further screening, waitlisted and rejected to depict the application status. Final membership value of a given rule is calculated as:

$$\begin{aligned} \textit{Membership value for a given rule} &= w_1 *(\textit{Membership values of GPA}) + w_2 *(\textit{Membership value of GRE score}) \\ &+ w_3 *(\textit{Membership value of Professor Recommendation}) + w_4 *(\textit{Membership value Number of Publication}) + w_5 *(\textit{Membership value Parent Institute Rating}) + w_6 *(\textit{Membership value Work Experience}) \end{aligned} \tag{1}$$

Where w_1, w_2, w_3, w_4, w_5, w_6 are weightages associated with corresponding input parameters. In this paper the weightages considered for the input parameters are as follows: GPA (w_1): 0.25; GRE Score (w_2): 0.1; Professor Recommendation (w_3): 0.25; Number of Publication (w_4): 0.2; Parent Institute Rating (w_5): 0.1; Work Experience (w_6): 0.1 The weightages given to each aspect can be modified and edited as per the university norms and guidelines in order to provide flexibility of usage by a wide range of universities.

As a result of the above mentioned criteria the applicants can be screened on a preliminary basis into the following categories:

1. Eligible for secondary screening (Selected)
2. Decision to be based on other factors like statement of purpose, motivation and overall promise (Waitlisted)
3. The applicant has inadequate qualifications (Rejected)

Hence the applications can be scrutinized in a more systematic and orderly manner with the help of the proposed system.

Table 1. Sample rule base for fuzzy inference system

GPA	GRE Score	Professor Recommendation	No. of Publications	Parent Institute Rating	Work Experience	Result
Good	Good	Good	Yes	Good	Good	Selected
Good	Good	Good	No	Good	Good	Selected
Good	Good	Average	Yes	Good	Good	Selected
Average	Good	Good	Yes	Good	Good	Selected
Good	Average	Good	Yes	Good	Good	Selected
Good	Average	Good	No	Good	Good	Selected
Good	Bad	Good	Yes	Good	Good	Waiting List
Good	Bad	Good	No	Good	Good	Waiting List
Good	Good	Average	No	Good	Good	Waiting List
Good	Average	Bad	Yes	Good	Good	Waiting List
Average	Good	Good	No	Good	Good	Waiting List
Bad	Average	Good	Yes	Good	Good	Waiting List
Good	Bad	Average	No	Good	Good	Rejected
Good	Bad	Bad	No	Good	Good	Rejected
Good	Average	Bad	No	Good	Good	Rejected
Good	Bad	Bad	Yes	Good	Good	Rejected
Average	Bad	Good	No	Good	Good	Rejected
Bad	Bad	Good	No	Good	Good	Rejected
Bad	Average	Good	No	Good	Good	Rejected
Bad	Bad	Good	Yes	Good	Good	Rejected
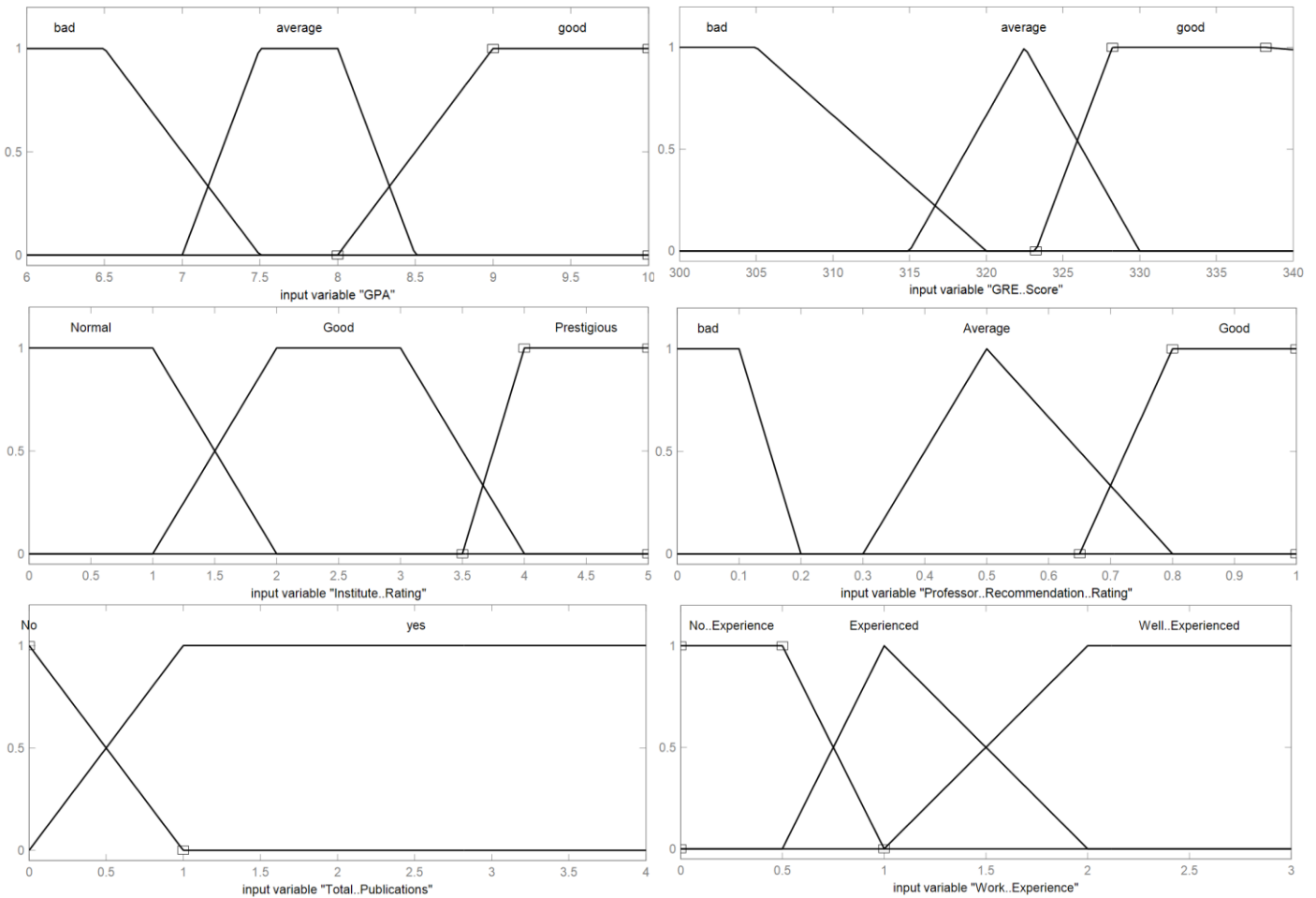
Fig. 2. Input variable membership function

Table 2. Sample example results

GPA	GRE Score	Professor Recommendation Rating	No. of Publications	Parent Institute Rating	Work Experience	Result
9.2	325	0.75	1	3.5	0.5	Selected
8.2	313	0.8	2	3	1	Selected
7.1	321	0.7	3	4	0	Waitlisted
7.3	310	0.7	0	4	1	Rejected
6.6	305	0.6	0	3	0	Rejected

5   Example application

This section presents an example to demonstrate the applicability of the proposed approach in effective screening of the graduate applications for further processing. Five different cases with variation in different input parameters are considered and the output given by the proposed approach is observed. These results are presented in Table 2.

6   Conclusion

The graduate application review process is very time consuming process and based on the personal bias of the admissions committee members. Each application has its individual strong characteristic hence comparing two applications based on a set cut-off is not a very efficient method. Fuzzy based system presents a convenient way for representing the relation between multiple input parameters to less output parameters by means of verbal statements instead of complex mathematical equations. The proposed system takes into consideration all the important aspects of an application such as the number of publications, professor recommendation, parent institute rating, grade point average (GPA), GRE score and work experience in order to provide an advisory on whether the application should be considered for further review or not. The system also suggests whether a further decision on the application should be made based on other factors like statement of purpose, motivation and overall promise. The weightages given to each aspect can be modified and edited as per the university norms and guidelines in order to provide flexibility and the possibility of usage by a wide range of universities. Thus the fuzzy based inference system provides an efficient method to scrutinize the applications in a more systematic and orderly manner to make the process of preliminary screening of applications faster and free from the bias of the reviewers.

7   References

[1]   Santoso Wibowo, Hepu Deng. "A fuzzy rule-based approach for screening international distribution centers." Computers and Mathematics with Applications (Elsevier), vol. 64, pp. pp. 1084-1092, 2012.

[2]   MathWorks (2010). MATLAB 7.9.1 (R2009bSP1), The language of technical computing. Natick, Massachusetts: The MathWorks, Inc.

[3]   Karthik Raghunathan (2010). Demystifying the American Graduate Admissions Process [Online]. Available: http://nlp.stanford.edu/~rkarthik/DAGAP.pdf [Accessed 5 June 2013]

[4]   John Yen, Reza Langari. "Fuzzy Logic: Intelligence, Control and Information" Pearson Education Inc, Second Impression 2007.

SESSION

NLP, MACHINE LEARNING, TEXT MINING, OCR, INTELLIGENT SYSTEMS, FUZZY LOGIC, LEARNING AND DECISION MAKING METHODS, VISUALIZATION, AND APPLICATIONS

Chair(s)

Prof. Hamid Arabnia
University of Georgia

Semantic processor for knowledge extraction from texts in Russian and English

Igor P. Kuznetsov, Elena B. Kozerenko, Nikolay V. Somin

Institute for Informatics Problems of the Russian Academy of Sciences, Moscow, Russia

Abstract *The paper is dedicated to one approach to the automatic extraction of the knowledge from natural language texts (Russian and English) with forming the Knowledge Base. It is used for the solution of the most complex problems of the linguistic processors and logical analytical systems. For this purpose the means of knowledge representation (the extended semantic networks - ESN) and the tools of their processing (the language of logical programming DECL) have been designed. On this basis the universal syntactical semantic rules and ontologies have been proposed which are composed of the universal linguistic knowledge for knowledge extraction and which have been used for construction of many intellectual systems for different applications.*

Keywords Semantics, Natural Language, Linguistic Processor, Knowledge Extraction, Named Entities

1 Introduction

The existing Internet largely consists of unstructured documents. Knowledge contained within these documents can be made more accessible for machine processing by means of transformation into the form to be reasoned with. A more prospective approach consists in using the Knowledge Base. It proposes the development of new technology including the extraction of knowledge structures and organization of their processing in the Knowledge Base [1,2,4].

The distinctive features of our technology are as follows:

1) extraction from texts of knowledge structures (not only separate named entities) that represent the links of named entities and their participation in actions and events;

2) for the knowledge extraction the unique semantic-oriented language processors (LP) are designed; the processor provides deep analysis of NL texts and revealing set of entities together with their structures;

3) the processor LP is controlled by the linguistic knowledge, which are declarative structures (in extended semantic networks - ESN) and which provides the quick tuning of LP to subject areas and languages - Russian and English;

4) linguistic knowledge consists of the rules, which provide the high degree of selectivity in the entities extraction and elimination of collisions during their application; rules provide the minimization of noise and losses, that is the high degree of completeness and accuracy;

5) the knowledge structures and means of their processing (intellectual language DEKL) were designed

as the united tools, oriented at the tasks of linguistic analysis, semantic search, logical analytical processing and the expert solutions. Using these tools considerably facilitates the development of applied intellectual systems.

The technology of knowledge structure extraction and processing has been used for construction of new classes of analytical systems [3,7,12,13]: "Criminal", "Analytic", "AntiTerror", "Resume" etc. (http://IpiranLogos.com/en/Systems/).

2 Tools for intellectual processing

2.1. Extended semantic networks

For knowledge representation the formalism was proposed, i.e. the Extended semantic networks (ESN) [2,3,4]. In the ESN constructions the paradigm is used in which the model of the external world is quantized to the objects and the relationships between them. At the same time the integration of objects is allowed when from simple objects it is possible to build more complex ones. The reverse process is the specification. In each object the parts connected by certain relationships can be selected. This is easily expressed in the natural language (NL) and should be presented in knowledge structures. The extended semantic networks have been designed on the base of this paradigm.

Extended semantic networks (ESN) are composed of fragments of following type:

<Relation name> (<arg1 "," arg2>, ..., <argN> / <code of fragment>)

where *<arg1 "," arg2>, ..., <argN>* are the argument places, which may be occupied by constant, or the number of variables, which may correspond to the named entities (information objects). The code of a fragment corresponds to a complex object, i.e. arguments with their relationship, which are considered as a whole. The "relationship" is considered in the broad sense. A unary relation (with one argument) is a property. A binary relation connects the two objects, and an N-ary relation connects more objects. For example, an N-ary relation can be an action in which N objects took part (with different roles).

Codes of fragments are needed to represent the level of integration. A code of fragment may be a constant, which must be "unique", i.e. it cannot be a code of another fragment.

A fragment code may be missing in the record. Then the fragment will take the simpler form:

<Relation name> (<arg1>,<arg2>, ...,<argN>).

It is not difficult to see that the fragments have the form of named predicates, where the fragment codes are the unique names of predicates. Many fragments are composed of the extended semantic network (ESN). The order of the fragments in the ESN does not matter. Two features should be noted.

The first feature is the use of the so-called intersystem constants. They are written in the form of numbers with a plus sign (N +), when a constant is introduced for the first time, and a minus (N-), when it has already been mentioned. For example, two fragments NAME(IVAN, 1 +) STRONG (1 -) represent that " *a man named Ivan is strong."* In this case, if we again encounter a number 1 +, it would introduce a new (different) constant. For example, the fragments NAME(IVAN, 1 +) STRONG (1 -) NAME (PETER, 1 +) BRAVE (1 -) WEAK(1 -) are presented two people: "*Ivan is strong, and Peter is brave and weak*." Instead of sign 1+ and 1- there can be any integer (N), i.e. 2 +, 2 -, etc. Intersystem constants are needed to refer to objects that are defined by their properties, relations or presented implicitly. If the text has the two objects named *Ivan*, these can be different people and they are represented in ESN by different constants. It is a difficult procedure to choose their different mnemonics.

The second feature is the following. The codes of fragments (usually intersystem constants) can stand in the argument places of other fragments. It is necessary in the cases when some objects are components of others. For example, the fragments NAME(IVAN, 1 +) BUY (1 -, BOOK / 2 +) DECIDE(1 -, 2 -) are represented "*Ivan decided to buy the book"*, where there are two actions. In this one (DECIDE) includes another (BUY). Every named entity (NE) may be the components of actions or other objects. Because every NE is presented as a fragment of ESN with its own code (see 3.3).

Described features (when some code fragments can be in the other argument places) greatly increases the possibilities of language ESN for representation of different types of information, including the semantic components of NL-construction. They are widely used for describing events and actions by forms with verbal nouns, participial and other constructions. It is significant that these features make the possibilities of language ESN reach far beyond the classical language of predicate logic. For example, ESN make possible the representation of various types of paradoxes which are typical for the NL, but impossible in the logic [1]. Constructions of ESN compose the United Knowledge Base which is used for subject and linguistic knowledge representation and which determines the logical analytical decisions and the work of linguistic processor.

2.2. Logical programming language for knowledge processing

For processing the knowledge structures, presented by ESN, a special language of logical programming (DECL) has been constructed [2,3]. The DECL language was used as the base for programming the linguistic processor (LP) which transforms the surface (space)

structures of texts to deep (semantic) structures where named entities and their relationships are represented.

The language DECL consists of the rules IF ... THEN ..., called productions. Productions are applied to the knowledge base (KB) and have the form: *<Name productions> (...): IF <LfP> THEN <RgP>;* where *LfP* is left part of the production and *RgP* is right part of it. Both parts consist of set of ESN fragments, which (in addition to constants) may contain variables. The fragment *<name productions>* (...) is necessary to call the production application.

The left part (LfP) of a production rule sets the conditions for its application. If the conditions are satisfied (i.e. analogical structures are contained in the KB), then the production is considered to be applicable. As a result the variables in LP take the values and are activated in the right part.

The right side of productions (RgP) determines the actions performed for transformation of structures in the KB. If the productions were applicable, then the actions would be initiated. The values of the variables are transferred from the LfP to RgP and are taken into account in actions.

The parts LfP and RgP contain not only fragments, but also special operators to call productions (by name), and the so-called special fragments (or build-in predicates) that define references to external procedures, for example, the interface programs.

The condition of their application consist in comparing the fragments of the LfP with the fragments of KB. If the corresponding structure in the KB is found, the product is considered to be applicable and values of variables from LfP are transmitted to RgP and take into account in the actions. If RgP has a fragment, then it is added to the KB (http://IpiranLogos/en/Tools/).

3. Representation of semantic structures

3.1 Type of entities and links for extraction

Named Entities (NE) are extracted from the documents in Natural Language by linguistic processor (LP) and represented in the Knowledge Base as the fragments of the extended semantic network. The arguments of fragments are the collections of normalized words, numbers and signs, which reflect essence of NE and indicate its type.

In our systems more than 40 types of NE are extracted from NL-texts [1,7,8]. Their quantity depends on the subject area and the tasks of users. Let us note that in the KB some NE can be constitutional components of others. Connections between NE may be complicated [1,6,14]. We consider that actions with their objects and components are the kind of NE, which are connected by special relations (time, space, reason and so on) with other actions. The ESN have been designed for the representation of such information on a homogeneous base which is necessary for deep computer processing of NL texts – Russian and English [1,10].

The set of the entities to be extracted depends on the tasks of a user. At the same time the quality of a linguistic

processor is determined by the possibilities for knowledge extraction. The linguistic processors of the systems "Criminal", "Analytic" and ''Semantix" support more than 40 types of semantic entities which can be extracted automatically.

Standard entities (names, dates, addresses, types of weapons and others) are reduced to one standard form. The identification of entities is performed taking into account brief designations (for example, separate

surnames, patronymics, initials), anaphoric references (indicative and personal pronouns, for example, this person, it...) definitions and explanations (for example, the *Mayor of Moscow Sobyanin* is identified with the subsequent words *mayor, Sobyanin*). An important task is the identification of entities in the entire text, the use for these purposes of indicative pronouns, brief names, anaphoric references.

The Graph presentation of some extracted entities is shown on fig 1.

Fig. 1 Some extracted named entities

3.2. Connections between the entities and participation in actions

Connections and relations between NE, extracted from the NL-texts, can be very diverse. They depend on entity types. For example, one person can be connected with another by relative and friendly relations, and also by the place of living, area of interests and so on. Actions frequently are connected with the time and the place. There can be reason-consequence and other connections between actions. In such a way the complex structures are created. For their formalization special tools of knowledge representation have been designed.

Actions usually are expressed in NL texts by the tensed verb forms, nonfinite verb forms, e.g. verbal nouns, participial and adverbial constructions, gerunds. The actions are also NE, components of which can be another NE. For example, there can be those, who participate in action, or entities, on which the action is

directed. Moreover, some actions may be components of others. For many applications the actions are also the significant information which requires formalization. Because the system is oriented at the deep analysis of text constructions, it extracts all actions and events with NE.

3.3. Meaningful portrait of a document

A meaningful portrait of a document is the formal representation of entities (NE), their properties and the connections, extracted from the text of the document. Such portraits are the structures of knowledge. As means of formalization in our technologies we use the extended semantic networks. Formalization is achieved automatically by the semantics-oriented linguistic processor, which analyzes the texts of NL documents and transforms them into knowledge structures [1,2,9].

A set of meaningful portraits (together with index files) comprise the Knowledge Base (KB) where various types are provided of semantic search and logical analytical functions by comparison and transformation of knowledge structures. We design the technology which provides the processing in the KB distributed within the net of computers.

A meaningful portrait consists of the elementary fragments, arguments of which are words in the normal form (it is necessary for the search and processing). Each elementary fragment has its unique code, which is written in the form of the number with the sign + and is separated by a slash line. For example, in the fragment FIO("ABU - TARIK"," "," "/3+) the sign "3+" is its code (but "3-" is the reference to it). Fragments DOC_(22, TERR_DOC.TXT", "SUMMARY; " /0+) 0-(ENG) indicate that the meaningful portrait is built on the basis of the English-language text of document with number 66 of the file of TERR_DOC.TXT", which was processed as the summary of the incidents (linguistic knowledge depends on this). The following fragments present date DATE_(…/1+), criminal group CRIM_GROUP(…/2+), person's surname (name and patronymic) FIO(… /3+) and so on. The signs "1+", "1-" and "2+", "2-" and "3+", "3-", … are the codes of the fragments, corresponding to the NE. With the aid of the codes the connections and relations of NE are assigned. Actions are represented in the form of fragments of the type DESTROY(ALIEN,3-/4+) 4-(66,ACT_), where it is represented as "alien person (FIO with code "3+"), are destroyed". Fragment 4-(66, ACT_) indicates that the first fragment DESTROY(…./4+) presents the action and relates to the document with the number 66. Fragments PLACE_(CHECHEN,REPUBLIC/5+) WHERE(4-,5-) indicate the place of this action (WHERE). Fragments ORG_(…/6+) INFORM(6-/7+) 7-(66,ACT_) represent that "organization … was informed".

The fragments PREDL_(...), which correspond to the sentences play the special role. They are filled up with the words, which did not enter into the named entities (in this example they are absent), or with the codes of entities themselves. To these fragments the indicators of their position in the text are added. For example, the fragment SENTENSE_(66,12-,16-/20+) 20-(5,254,471) represents the fact that the entities with codes "12-" (corresponding to the action "inform"), "16-" (corresponding the action "destroy") are located in the sentence, which begins from the 5th line of the text of the document and they occupy the place from the 254-th to the 471-th byte. These means of positioning are necessary for the work of the reverse linguistic processor.

A set of meaningful portraits of documents are organized in the Knowledge Base. Logical reference is provided with the aid of the rules IF… THEN (productions) of the language DECL, which are the basis for decision of logical-analytical tasks.

4 Semantic-oriented linguistic processor

Semantics-oriented linguistic processor consist of the following components [9,14].

4.1 The component of lexical and morphological analysis (LMA)

It extracts words and sentences from the text, performs lemmatization of words (normal form establishment) and constructs the semantic network presenting the space structure of text (SpST), which reflects the sequence of words, their basic features, beginnings of sentences and the presence of space character lines. The component LMA uses a two-level general ontology and a special collection of subject dictionaries (the dictionary of countries, regions of Russia, names, forms of weapons, and other items specific for the supported domains). The component performs semantic grouping of the words and assigns them additional semantic attributes [10].

4.2 The component of syntactic-semantic analysis (SSA)

It converts one semantic network (SN) into another which represents the semantic structure of text (SemST), i.e., the relevant semantic entities and their connections [3,8,9]. The SemST is called the meaningful portrait of document. It comprises knowledge structures of the knowledge base which serves the basis for implementing different forms of semantic search : the search by features and connections, the search for the entities connected at different levels, the search for similar persons and incidents, the search by distinctive characteristics (with the use of ontology).

The component SSA is controlled by the linguistic knowledge (LK), which determines the process of text analysis. LK includes the special contextual rules which ensure the high degree of selectivity with the extraction of entities and connections The functions of this component are the following:

- Extraction of entities from the flow of NL documents: persons, organizations, actions, their place and time, and many other relevant types of entities.

 - The establishment of connections between entities. For example, persons are connected with organizations (PLACE_OF_WORK), by addresses (LIVES, REGISTERED). Or figurants of criminal events are connected with such entities as the type of weapon, drugs (TO HAVE).

- The analysis of finite and nonfinite verbal forms with the identification of the participation of entities in the appropriate actions. For example, one figurant gave the drugs to another figurant, and this is the fact linking them.

- The establishment of the connections of actions with the place and time (where and when some action or event occurred).

- The analysis of the reason-consequence and temporary connections between actions and events.

4.3 Expert system component (ES)

On the basis of semantic networks the new knowledge pieces are constructed in the form of additional fragments (ESN). For example, the component

ES extracts the field of a person's activity (in accordance with the assigned classifier) from the text of resume for each autobiography. The person's experience in his field is evaluated. The correlation of a criminal incident to the specific type is accomplished with the analysis of the criminal actions of ES: the following facts are revealed - the nature of crime, the method of its accomplishment, the instrument of crime, and so forth (in accordance with the classifiers of the criminal police) [3,12,13].

5 The base of linguistic and expert knowledge (KB)

It contains the rules of the text analysis and expert solutions in the internal representation. They determine the work of the linguistic processor. Our logicalanalytical systems have several such bases, which are activated depending on subject areas and user tasks.

5.1 Linguistic knowledge

Linguistic knowledge has same structures for various language that give possibilities to tune the processor LP on the text collection in this language, for example, Russian and English. Linguistic knowledge is written in language SSN which has declarative structures. It provides the tuning to new subject field and language for comparatively short time. Procedures part of LP is not changed (excluding blocks of lexical morphological analysis).

5.2. Terminological analysis and transformations

Terminological analysis has as a goal - synonymous transformations, the interpretation of abbreviations and the selection of terms. The fragments of the following form are used for this:

TERMIN (<resulting word>,<word1>,<word2>) or
TERMIN (<resulting word>,<word1>,<word2>,<word3>),

where <word1>,… may be normalized word (in canonical form), or sign, or AND-OR graphs. These graphs are represented as fragments STR_OR (...), where facultative words or their signs are on argument places. For example, fragment

TERMIN (UNEMPLOY,NO,WORK)

indicates the conversion: NO WORK - > UNEMPLOY. Another example: fragments

TERMIN (MO,MOSCOW,3+) STR_OR(REGION,REG.,DISTRICT,…/3-)

perform many conversions: MOSCOW REGION - > MO, MOSCOW REG. - > MO, MOSCOW DISTRICT - > MO… For these fragments the permissible context can be assigned (words, which can stand to the left and to the right). The inadmissible context can be also indicated - word or their signs, which there must not be to the left or to the right. As a result it is possible to extract terms and word combinations, whose values depend on context.

Synonyms are presented by fragments:

SYNON (<resulting word>,<word1>,<word2>).

For example, SYNON (UKRAINIAN,HOHOL) - word HOHOL (specific name of Ukrainian persons) must be substituted on UKRAINIAN. Many synonyms have conditional nature. The permissible or inadmissible context is indicated for them. For example, in the case given above replacements for the words - surnames, the nicknames, names streets and others are not admitted.

The ontology is presented by fragments of ESN with name SUB (class – subclass), NEAR (nearness of meaning) and OR_OR (separate "or"). For example:

SUB(MAN,TERRORIST)
SUB(TERRORIST,SEPARATIST)
SUB(TERRORIST,REBEL)
SUB(TERRORIST,INSURGENT)
SUB(TERRORIST,MERCENARY) . . .
NEAR(ALCOGOL,DRUNK,TIPSY,VODKA) . . .
OR_OR(MALE,FEMALE,CHILD) . . .

5.2. Contextual rules

The block of syntactical-semantic analysis on the basis of context extracts the named entities (NE) and the connected information (for the persons their birthday, sex, address and other) [2,15]. For this contextual rules are used. Syntactical-semantic analysis is necessary for the extraction of addresses, attributes of machines, organizations and other. Usually the entities are the collections of words, which grammatically are not connected. For example, an address can be considered as the collection of letter combinations st. (street), h. (house),.:., words with the capital letter and the numbers. Each such collection can have its boundaries and inadmissible components. For example, in the addresses there can be no verbs and so on. The extraction of such word collections (descriptions of NE) is based on the use of contextual rules of the following form:

CONTEXT (<word1>,<word2>,<word3>,…) - > <resulting fragment>

where <word1>,… - are the normalized word or sign or AND-OR graphs. For every rule a special fragment indicates the position to begin application, and also a permissible or an inadmissible context. This ensures the differentiated application of rules. These rules analyze word group, which describe any entities, and substitute them (in case of application) by one word, with which the resulting fragment is connected.

Contextual rules are applied in the determined sequence. At first they extracted the separated entities, then their properties, word combination, and finally, verbal forms. In process of rules application the meaningful portrait of document will be build.

For example, let us examine rule GG~1:

MUSTBE (GG~1,1) STR_OR(ADJ,PRON/2+)

CONTEXT (2-, NOUN/GG~1) P_P (GG~1,3+) WORD_C (1,2/3-)
  3-(2,MORF) NOTBE (GG~1,2,LETT)

This rule provides the conversions:
ADJECTIVE + NOUN → <word combination> and
PRONOUN + NOUN → <word combination>.

The fragment MUSTBE indicates that application of rule GG~1 must begin from the first position, i.e., from search for words with the signs ADJECTIVE (ADJ) and PRONOUN (PRON), since they are fewer than NOUN (NOUN). Fragment P_P separates the left side from the right side (→), and WORD_C - indicates that the words on the first and second positions must be united into the combination of words, which subsequently will be considered as one word with the morphological properties of the second word. The fragment NOTBE indicates that in the second position there cannot be the separate letters (sign LETT).

This is an example of a simplest rule. The fragments which indicate the context are added to such rules; these help to restore the full name of an entity, for example, on the basis of pronouns or brief descriptions (by the name the surname is restored, if they were somewhere mentioned together), etc.

Each contextual rule is a semantic network (ESN). All linguistic knowledge is written in the ESN language. The application of rules is provided by the productions of the DEKL language. These productions are organized as programs, which play the role of the empty linguistic shell, which supports the language of the record of linguistic knowledge - ESN. As it is shown by experience, this shell can be tuned to different subject fields and languages. In this way different linguistic processors are designed.

5.3. Application of the rules

Application of contextual rules is fulfilled in the strictly defined sequence - each at their level. For example, in system "Criminal" the linguistic processor at first extracts the following named entities - police department, the policemen and others. They can contain surnames, names. It is necessary to facilitate the subsequent analysis. Otherwise the words, which compose these entities, can be captured by other rules and create noise. Further figurants are extracted and so on. The set of rules is introduced for this. Some begin their application from the search of names, surnames (MUSTBE), others - from the search for the birthday, or from the initials. In this way we minimize losses in cases when the block of morphological analysis does not give the necessary data for some words. Then the word combinations are analyzed, and finally, verbal forms. In the process of application of these rules the semantic network (meaningful portrait of document) is built. An example of the levels, which determine the order of the rule application, is given below.

{== levels ==}
LEVEL (LEVEL_E1, LEVEL_E2, LEVEL_E3, LEVEL_E4,…)
LEVEL_E1 (CATALOG) {= extraction of word combinations from the catalogs =}
LEVEL_E2 (MIL~~1, ST~~1) {= extraction of police departments,… =}
LEVEL_E3 (FF~~1, FF~~2) {= extraction of figurants =}
{== grammatical analysis, the extraction of word combinations =}
{== AA~~… - uniform terms, GG~~… - words combination ==}
LEVEL_E4 (AA~~1, AA~~3, AA~~4, GG~~1, GG~~2,…)

{= GG~~1: word combination ADJECTIVE – ADJ or PRONOUNCE - NOUN=}
MUSTBE (GG~~1,1) STR_OR (ADJ, PRON/2+) CONTEXT (2-, OBJ/GG~~1)
P_P (GG~~1,3-) WORD_C (1,2/3+) 3- (2, MORF) NOTBE (GG~~1,2, LETT)
. . .

In the curly braces the commentaries are given. It is an example of the rule GG~1, which reveals word combinations with signs ADJ or PRON and OBJ (i.e. NOUN etc.). The system has a full set of contextual rules, which provide the complete analysis of sentences and construction of meaningful portraits of documents. But in contrast to the standard grammars our LP provides the extraction of all significant (information) entities, including those in which the words are not connected grammatically, for example, addresses, machines with the indication of their numbers and so on. The described processor (LP) is semantics-oriented, because it provides the extraction of entities and supports various kinds of connections between them. These are semantic components. Such LP found their use in the systems of a new class – "Analytic", "Criminal", "AntiTerror" and other.

6 Conclusion

The described semantic-oriented linguistic processors have been used for construction of intellectual analytical systems: "Criminal", "Analytic", "AntiTerror", "Monuments" and others. The distinctive features of these systems are characterized by universality of the processor. It provides automatic extraction of knowledge structures from texts in various language. Now they are Russian and English. The processor can be quickly tuned to other European languages. The result of the processor operation is the Knowledge Base which has common structure for all language and which is used for realization of logical-analytical functions. The ESN apparatus provides powerful representational possibilities for describing all levels of natural language, including the level of deep semantic structures, and cross-lingual correspondences.

The implemented linguistic processors were created on the basis of this approach which made it possible to manufacture design solutions for the basic

problems of extracting meaningful knowledge from the texts in natural languages.

References

[1] Kuznetsov, I.P. Elena B. Kozerenko, Mikhail M. Charnine. Technological peculiarity of knowledge extraction for logical-analytical systems // Proceedings of ICAI'12, WORLDCOMP'12, July 18-21, 2012, Las Vegas, Nevada, USA. - CRSEA Press, USA, 2012.

[2] Kuznetsov I.P. Matskrvich A.G. Semantic oriented systems controlled by knowledge base // University of communications and informatics, Moscow, 2007,173 p.

[3] Kuznetsov I.P. Methods of Processing Reports with the Extraction of Figurants and Events Features // In "Computational Linguistics and its Applications", Vol.2, Tarusa, 1999.

[4] Kuznetsov I.P. Semantic Representations // Moscow: "Nauka", 1986. 290p.

[5] Kozerenko, E.B. Multilingual Processors: a Unified Approach to Semantic and Syntactic Knowledge Presentation. In Proceedings of the International Conference on Artificial Intelligence IC-AI'2001 25-28, 2001. CSREA Press, 2001, pp.1277-1282.

[6] Byrd, R. and Ravin, Y. Identifying and Extracting Relations in Text // 4th International Conference on Applications of Natural Language to Information Systems (NLDB). Klagenfurt, Austria, 1999.

[7] Kuznetsov I.P. Natural Language Texts Processing Employing the Knowledge Base Technology // Sistemy i Sredstva Informatiki, Vol.13, Moscow: Nauka, 2003, pp. 241-250.

[8] Kuznetsov I.P., Matskevich A.G. The English Language Version of Automatic Extraction of Meaningful Information from Natural Language Texts // Proceedings of the Dialog-2005 International Conference "Computational Linguistics and Intelligent Technologies", Zvenigorod, 2005pp. 303-311.

[9] Kuznetsov, I., Kozerenko, E. The system for extracting semantic information from natural language texts // Proceeding of International Conference on Machine Learning. MLMTA-03, Las Vegas US, 23-26 June 2003, p. 75-80.

.

[10] Somin N.V., Solovyova N.S., Charnine M.M The System for Morphological Analysis: the Experience of Employment and Modification // Sistemy i Sredstva Informatiki, Vol. 15 Moscow: Nauka, 2005, pp. 20-30.

[11] Cunningham, H. Automatic Information Extraction // Encyclopedia of Language and Linguistics, 2cnd ed. Elsevier, 2005.

[12] Kuznetsov I.P., Matskevich A.G. Semantics Oriented Linguistic Processor for Automatic Formalization of Autobiographical Data // Proceedings of the Dialog-2006 International Conference "Computational Linguistics and Intelligent Technologies", Bekasovo, 2006, pp. 317-322.

[13] Web site "Knowledge extraction for Analytical Systems": http://Ipiranlogos.com/english/

[14] Kuznetsov I.P., Kozerenko E.B., Matskevich A.G. Deep and Shallow Semantic presentations in Intelligent Fact Extractors // Proceedings of ICAI'2010 Las Vegas, USA, June 14-17, 2010, CRSEA Press, 2010.

[15] Kuznetsov, I.P., Kozerenko E.B. Semantic Approach to Explicit and Implicit Knowledge Extraction // Proceedings of ICAI'11, WORLDCOMP'11, July 18-21, 2011, Las Vegas, Nevada, USA. - CRSEA Press, USA, 2011.

[16] Clark P., P. Harrison, and J. Thompson. A Knowledge-Driven Approach to Text Meaning Processing // Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning, 2007. P. 1–6.

[17] Gildea D. and M. Palmer. The necessity of syntactic parsing for predicate argument recognition. In Proceedings of the 40th Annual Conference of the Association for Computational Linguistics (ACL-02), Philadelphia, PA, 2002. P. 239–246.

[18] Pasca M. and B. Van Durme. What You Seek is What You Get: Extraction of Class Attributes from Query Logs // Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07), 2007. P. 2832–2837.

[19] Punyakanok V., D. Roth, and W. tau Yih. The Importance of Syntactic Parsing and Inference in Semantic Role Labeling // Computational Linguistics 34(2), 2008. P. 257–287.

Social Network Analysis – Network and Link Detection in Overwhelming and Noisy Data Streams

Craig Anken, Pete LaMonica

Air Force Research Laboratory/RIEB

{Craig.Anken, Peter.LaMonica}@rl.af.mil

James Schneider, Tod Hagan, Lee Krause, Bruce McQueary

[1]Securboration, Inc., 1050 W. NASA Blvd, Melbourne, FL 32901

{jschneider,thagan,lkrause,bmcqueary}@securboration.com

Abstract – *The explosion of social media and availability of open source data has created an unprecedented opportunity to expose hidden intelligence. In this work we explore the use of a pipeline of novel algorithms executing within a parallel computing framework to improve SNA performance as compared to baseline approaches. These initial findings show improvements in link detection speed and accuracy (F-measure) on community datasets with ground truth information.*

Keywords: Social Network Analysis, Machine Learning, Non-obvious Link Extraction, Parallel Machine Learning Frameworks, Noise Reduction and Removal of Irrelevant Intelligence

1   Introduction

The explosion of social media and availability of open source data has created an unprecedented opportunity to expose hidden intelligence. Illicit and disruptive activities are being carried out in plain view including protests and social movements, attack planning and coordination, plotting flash riots and robberies, promoting radical ideology and recruiting new members. Uncovering actionable intelligence or manually '*connecting the dots*' in overwhelming, heterogeneous and noisy social media data streams has proven extremely difficult for the U.S. and our coalition partners.

The overall goal of this research is to investigate techniques which improve the accuracy and speed of social network analysis (SNA) while using fewer computing resources as compared to baseline approaches. In this paper, we describe ongoing research performed and results obtained to date by employing a combination of the latest machine learning techniques within a parallel computing framework. We have evaluated our research using community datasets with ground truth information and report encouraging results.

2   Related Work

A significant portion of current SNA research is directed towards analyzing data streams to collect intelligence; whether to analyze buying trends for the commercial market space or detecting nefarious activities for government intelligence entities.

Sitaram Asur and Bernardo A. Huberman from the HP Social Computing Lab in Palo Alto, CA have investigated how social media can be used to predict future events. They have used Twitter data to predict movie revenues. Their approach considers simple metrics such as tweet frequency and sentiment to predict outcomes.

Aditya Krishna Menon and Charles Elkan at the University of California investigated link prediction with an approach of supervised matrix factorization. Their approach uses features based on the graphs topological structure combined with other node and edge attributes. Stochastic gradient descent is used for ranking and class imbalance.

Hila Becker and Luis Gravano from Columbia University and Mor Naaman from Rutgers University have researched using Twitter messages to identify real-time events. Their work is focused on distinguishing between messages about events vs. non-events based on topic clusters over large message sets.

Due to the overwhelming amount of data, many current approaches will be CPU intensive. Our approach differs by employing a method of multi-stage algorithms, each narrowing the result by filtering unrelated information, detecting hidden but relevant relations and networks in a very efficient parallel framework for graph processing.

3   Approach

In this paper we explore the use of a pipeline of functions and algorithms executing within a parallel

computing framework. The pipeline consists of a series of capabilities designed to gather, detect and narrow results based on the user's original intelligence requirement (IR). The pipeline contains functional elements for performing data acquisition, noise reduction, network detection, link detection, and node influence.

3.1 Data Acquisition

In this functional element, social media data streams are queried based on the analyst's IR and parsed for key elements of information.

The system narrows the initial search to targeted topics and data sources based on the IR. The IR is parsed for keyword information including synonyms using a thesaurus. These are used to create a matrix of permutations on the original IR in order to achieve maximum coverage over the data sources.

3.2 Noise Reduction

A key function is reducing the ambient noise encountered in social media data streams. We have implemented an algorithm described in a paper by Dr. Carlos Guestrin and Yison Yue entitled "Linear Sub modular Bandits (LSB) and their Application to Diversified Retrieval". The LSB Greedy algorithm as detailed in this paper takes advantage of sub modular optimization and the upper confidence bounds applied to trees (UCT) in order to describe a noise reduction technique that is both fast and accurate.

The equation $E[r_t^l * A] = W^{*T} \Delta(A^l | A^{1:l-1})$ represents the optimization function for the linear sub modular bandit's problem. The W^{*T} is the unknown weight vector for the users' intelligence request. Optimizing for this weight threshold allows us to tune down the noise and help return only those results related to the IR. The linear sub modular bandits algorithm follows the exploration vs. exploitation tradeoff scheme[1] and allows for the user to tune the exploration and exploitation values for maximum/desired coverage over the dataset.

Due to the overwhelming amount of social media data, approaches such as clustering could take an inordinate completion time on standard computer. Dr. Guestrin's algorithm takes minutes to execute. We are continuing to collaborate with Dr. Guestrin to optimize and tune the algorithm implementation. Implementation of the Dr. Guestrin' s noise reduction algorithm decreased relative noise by 67% as

measured by one user study at CMU which compared the LSB Greedy with competing algorithms.[2]

3.3 Network Detection and Significance

The concept of a network is to represent a graph structure across varying data sources. A network is any structure which is repeated in a statistically significant manner. Within these structures exists well established structures such as the "feed-forward loop" or FFL that represents an AND or an OR gate inside of the network. These structures can be used in a machine learning algorithm as features to determine a network's significant values such as clustering coefficient, network density, and network connectedness. We have examined three possible network finding algorithms in order to assist with the task of finding both structural and content networks. The algorithm chosen for this process is the G-trie algorithm. The G-trie algorithm works by placing random walkers on the network and finding networks of user defined sizes to create networks. These networks will then be used in the statistical calculations in order to determine event classification type.

G-Tries

G-Tries is a very high compression algorithm based on the famous trie data structure. This structure allows for maximum compression of the underlying data and easier computability of the sub-graphs. A consideration when using tries is that starting from different points on a graph will give widely different data structures; to solve this, g-trie uses a common canonical representation where larger vertices appear first. The algorithm is split up into three major sections. The first is the algorithm to insert a sub-graph into a g-trie, the second is to find if the inserted graph is significant over all possible graphs, and the third is designed to stop recompilation of networks that have already been examined.

---

[1] http://homes.cs.washington.edu/~todorov/papers/tradeoff.pdf

[2] http://www.select.cs.cmu.edu/publications/paperdir/nips2011-yue-guestrin.pdf

```
Algorithm 1 Inserting a graph G in a g-trie T
1: procedure INSERT(G, T)
2:     M := canonicalAdjMatrix(G)
3:     insertRecursive(M, T, 0)
4: procedure INSERTRECURSIVE(M, T, k)
5:     if k < numberRows(M) then
6:         for all children c of T do
7:             if c.value = first k + 1 values of M[k] then
8:                 insertRecursive(M, c, k + 1)
9:                 return
10:        nc := new g-trie node
11:        nc.value := first k + 1 values of M[k];
12:        T.insertChild(nc)
13:        insertRecursive(M, nc, k + 1);
```

Figure 1: G-trie inserts a sub-graph

```
Algorithm 2 Census of subgraphs of T in graph G
1: procedure CENSUS(G, t)
2:     for all children c of T.root do
3:         match(c, G, 0, ∅)
4: procedure MATCH(T, G, k, V_{used})
5:     if V_{used} = ∅ then
6:         V_{cand} := V(G)
7:     else
8:         V_{conn} = {V_{used}[i]: T.value[i] = '1'}
9:         m := m ∈ V_{conn} : ∀v ∈ V_{conn}, |N(m)| ≤ |N(v)|
10:        V_{cand} := {v ∈ N(m) : v ∉ V_{used}}
11:    for all v ∈ V_{cand} do
12:        add v to end of V_{used}
13:        if ∀i ∈ [0..k]: T.value[i] = G_{Adj}[v][V_{used}[i]] then
14:            if T.isLeaf() then
15:                reportGraph();
16:            else
17:                for all children c of T do
18:                    match(c, G, k + 1, V_{used})
19:        remove v from V_{used}
```

Figure 2: Finding sub-graph significance

```
Algorithm 3 Symmetry breaking conditions for graph G
1: function FINDCONDITIONS(G)
2:     Conditions := ∅
3:     Aut := setAutomorphisms(G)
4:     while |Aut| > 1 do
5:         m := minimum v : ∃map ∈ Aut, map[v] ≠ v
6:         for all v ≠ m : ∃map ∈ Aut, map[m] = v do
7:             add m < v to Conditions
8:         Aut := {map ∈ Aut : map[m] = m}
9:     return Conditions
```

Figure 3: Avoiding redundant calculations

The ability for g-tires to avoid the redundant calculations induced in the census method of random graphs leads to a much faster algorithm. The ability of the algorithm to include both a computationally efficient data structure and the ability to break symmetries leads to the fastest available algorithm for discovering the random networks.

3.4   Feature Discovery

An important aspect of this research is this identification of connections between two nodes which are important to the analyst. We are investigating the combination of both noise reduction algorithms and multivariate Bayesian scan statistics in order to determine which connections are important and which ones have significant meaning for the analyst. These statistics are common to each network and will be averaged over all networks and categorized for event classification. These features include but are not limited to:

- Average Sentiment: Twitter POS tagger + Opinion Lexicon (English, Spanish)
- Average Tweets per second: *A=Tweets" " /(Event_Begin−Event_End)*
- Event time is measured in seconds from the year 1970
- Average Clustering Coefficient: $C\_i=(2e\_i)/(k\_i (k\_i−1))$
- Average Time active
- Average Influence of Network
- Average number of high influencers
- Topics tweeted via Latent Dirichlet Allocation
- Average network retweets
- Location distance of members (e.g. proximity of one member to another)
- Friends/Followers associates
- Geolocation of members
- Location in relation to event

3.5   Link Detection

A focus of this research is on the ability to find implicit links or relationships that a user might not see due to the overwhelming volume of disparate data. We have researched two detection methods for their efficacy and scalability. Both methods improved precision and recall over the current state of the art, but utilize online learning algorithms that improve performance by four fold over the traditional offline methods without significant loss of precision or recall. Online learning states that all data is not available at processing time and updates the link model incrementally as new data comes into the system. Offline learning processes requires that all data is present and processed at one time. When new data is added, all information must be reprocessed. Online processing is well suited to streaming data sources from large data sources such as Twitter or Facebook.

We have tested performance of the implicit relation detection algorithm on Twitter and Facebook (anonymized) datasets. Using these data sources we

have discovered very interesting characteristics of implicit relations in social networks. First, the "small world" phenomenon is vastly overstated in most of the literature. This phenomena is explained by the statement "No person is more than six degrees away from another". In processing the data we found that people are usually more than the standard six degrees apart if group aggregation is taken into account. We discovered many niche communities connected by separate networks, this contradicts the small world assumption as the lengths of these single chains was found to be around 10 nodes.

Preliminary results of the inferred entity relation algorithm using 10 folded crossover validation shows an 85% accuracy for inferred links without side channel information. Adding this information boosts the algorithm's accuracy to around 91.2% using the same data sets.

The biggest challenge in discovering relationships within the data was not the absence of the small world metric but the degrees of separation within groups. If all the friends/followers from a Twitter or Facebook dataset are graphed, a fully connected graph state is quickly reached. Most of these friends are erroneous and just placed in the list because they are either celebrities, corporate entities, or have a very loose association with the entity over the internet. With this data, we learned that going with a full duplex communications solution solves this issue and provides increased accuracy over more lenient models.

As depicted in the figure below, our link detection approach uses a multi-dimensional feature analysis with values plotted for Singular Value Decomposition, Modified Katz Index and Side Info associated with each node. The red circles represent nodes within the social network which are not connected. Blue circles represent nodes which are related to one another.
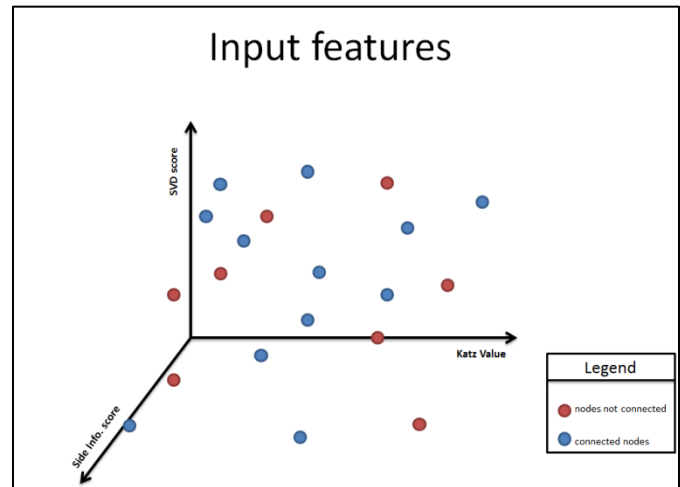
Figure 4. The link detection algorithm output showing social media nodes plotted in multi-dimensional space according to SVD score, Katz value and side information.

As shown in the next diagram, the multi-dimensional plane is processed using the Vowpal Wabbit to determine best fit of each node within a given hyper plane. The significance of this result is that nodes on same hyper plane share a relationship. Vowpal Wabbit is an extremely fast online learning algorithm out of Yahoo and Microsoft Research.
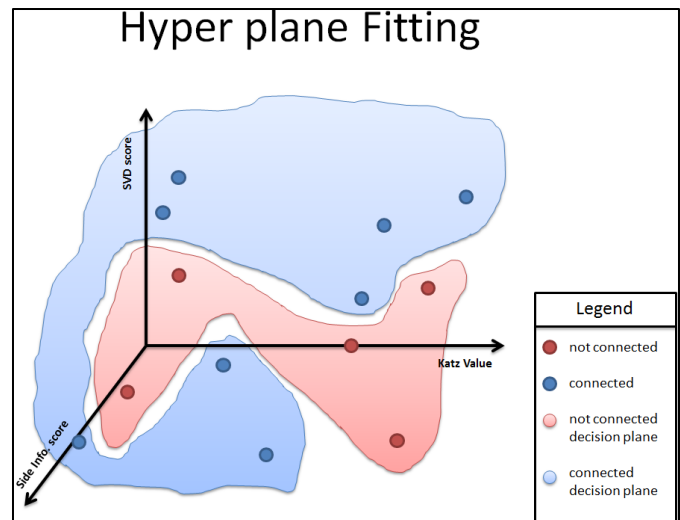
Figure 5. Hyper plane fitting of social network nodes/users using the Vowpal Wabbit online learning algorithm.

One of the most significant aspects of this approach is that as nodes change (e.g. new tweets or Facebook updates) or new social media users are added, the graph is quickly updated (i.e. online processing). Additionally, the implementation within GraphLab makes this efficient to add new nodes and allows

scalability from the single computer up to a cloud based solution.
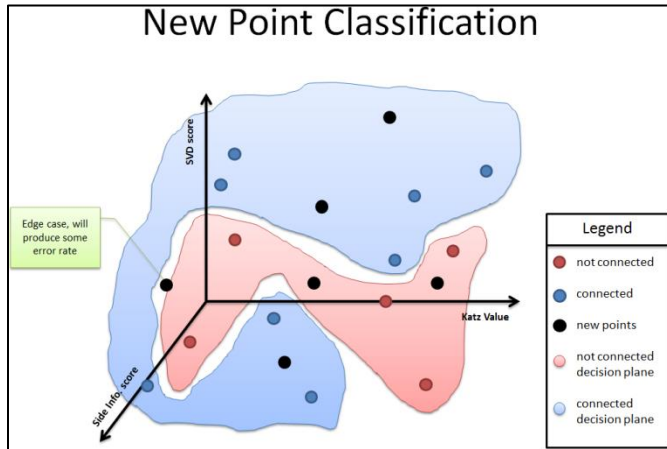
Figure 6. Response times are significantly reduced as new information becomes available within the social network. New points or nodes are quickly classified and placed on their appropriate hyper plane according to relations with other nodes on the same plane.

The final outcome of the pipeline is that groups of related nodes can be shown as well as relations between networks of networks as shown below. Network neighborhoods allow for discovery of who is more likely to discuss certain topics and allow the user to quickly discover new and interesting connections present in the data.
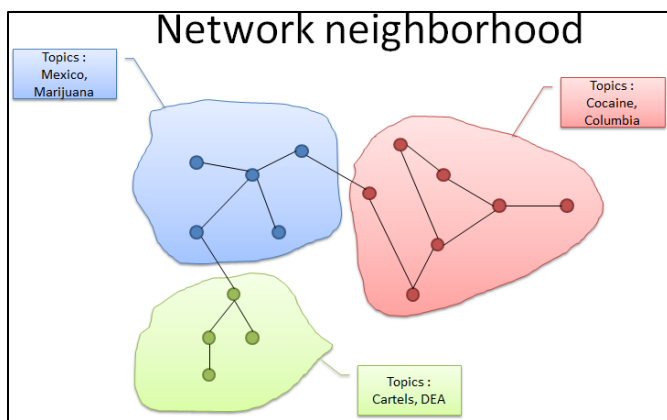
Figure7. The network neighborhoods graph where nodes are grouped according to their relationship with other nodes based on topics of interest and the original intelligence request.

Network neighborhoods are particularly important for many analytical reasons. Displaying the various groups and their connections provides explicit data of how different cells are related. These relations can

depict a multitude of relation types such as communication, transaction, familial, and contagion. Typically when analysts search, they focus on very specific cliques that are usually homogenous. However, the network neighborhood feature allows the analysts to gain both local and global intelligence across a plethora of node and edge types.

The network neighborhood feature is derived from the Layered Network Analysis model that was defined by LaMonica and Waskiewicz (2011) and refers to the fusion of previously disconnected data in a common operating picture allowing the user to explore and exploit interconnected data in a unified state. Layered Network Analysis can be used to discover patterns within and across layers, discover previously unknown nodes and edges, and allow users to focus on multiple data types within a specific time and space.

The intent is to understand the structure and dynamics of network data as this data is of vital importance to decision makers. Current analysis of network data occurs primarily within one or two particular domains and relies on the user to infer and derive links between data sources. However, to fully understand the network environment, decision makers must be able to investigate interconnected relationships of many diverse network types simultaneously as they evolve spatially and temporally. No single data network exists in a vacuum as networks are interconnected through space and time. Each layer represents a diverse network data type and these layers can be connected through nodal and edge similarities providing both local and global situation awareness.

3.6 Node Influence

We use a Modified Decreasing Cascade Model within PageRank statistics to determine which node has the most influence within a given network (e.g. # of re-tweets, # of connections, time until topic is tweeted, etc.). This influence score may allow the analyst to decide who should be monitored for further study or followed more closely. Influence scores are updateable and amendable by the analyst; the user can change the weights of the scores and recalculate influence based on this prior information. As shown in the graphic below, color intensity could indicate relative influence of individuals as compared to other members of the same neighborhood.
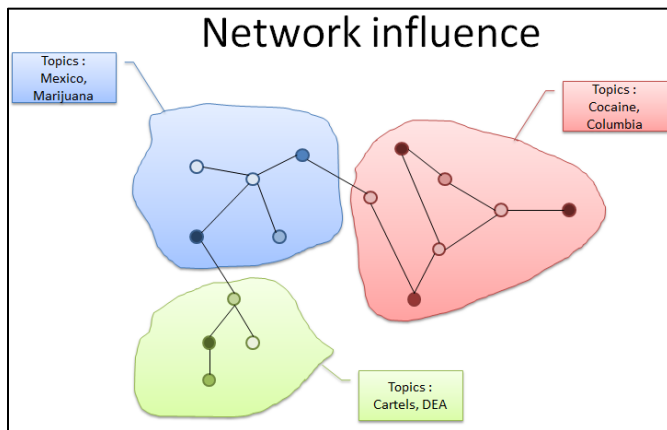
Figure 8. Identify the most influential nodes within a social network of users with similar interests. The strength of influence can be indicated by color intensity.

4 Results

Thus far, we have tested the link detection capability on two datasets:

- The co-author network of astro-physicists. SNAP provided 18,772 nodes with 396,160 edges.
- Co-author network of NIPS. Over 10,000 nodes and 600,000 edges. Each node has a 14035 word index and bag-of-words feature vector.

The output was compared against other approaches to classify nodes within a graph:

- Majority Classifier –Every point in the classifier is assigned to whichever the majority class is in the dataset (baseline).
- Random Classifier – Random flip if the node is connected or not (baseline).
- Perceptron Learner – Simple gradient descent learning algorithm with 2 layers of hidden nodes and trained with back propagation.
- Bayesian Network – Network of probabilities trained from the input data using EM algorithm to discover unknown values.
- SVM (kernel gradient optimization) – Using sub gradients to find the proper kernel metric and applying a SVM to the problem to generalize.

The link detection speed and accuracy exceed those of current baseline approaches. As compared to a Support Vector Machine, Securboration's approach is 322% faster with 5% better accuracy; compared to the Bayesian approaches testing showed 222% speed and 15% accuracy improvements. This particular test was based on a Twitter dataset with 81,306 nodes and 1,768,149 edges.
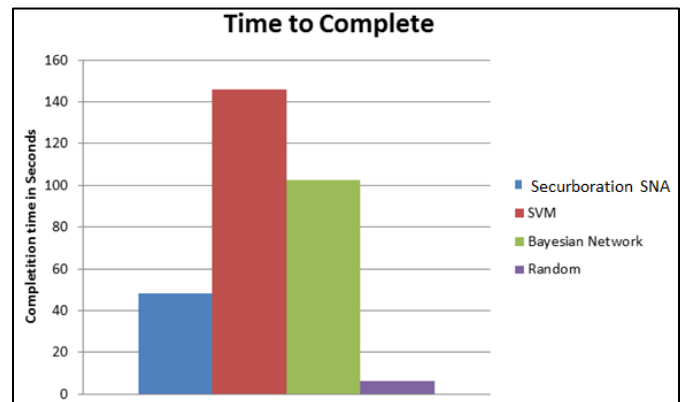
Figure 9: Securboration SNA Classifier Time to Complete
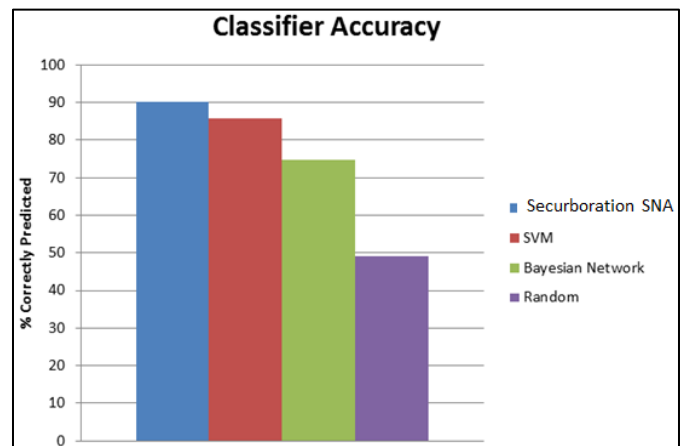
Figure10: Securboration SNA Classifier Accuracy

Figure 11 below shows a social network graph output from LAKE. The purple nodes in the graph are topics extracted from social media streams based on IR inputs. The red nodes are people with links to the topics and other individuals as detected by LAKE. The numbers sequences in black are unique ids for actual social media messages posted by the user and related to the IR. Relation strength is indicated by the edge thickness. The user can right click on graph nodes to display more information about the user and relevant messages as indicated by the green arrow.
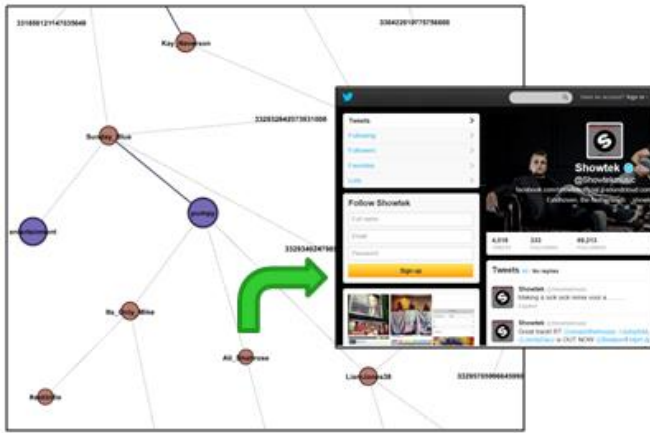
Figure11: LAKE generated social network graph showing topics, people and messages linked together and based on the original intelligence requirement

5   Conclusion and Future Work

This research represents a step forward in the development of tools which automate the analysis of social media data streams. The research uses the latest advances in the field of parallel processing, machine learning and data mining to increase speed, accuracy and precision of intelligence processing. This initial research proved that select algorithms could be configured in a pipeline and executed on standard computers to produce results which are superior to the current baseline. One objective of this research is to build a capability which can be deployed on standard hardware environments to increase adoption of the system.   Future work will focus on extending this research and the development of a service based suite of analytical tools to support the defense, intelligence, and law enforcement communities.

6   References

Asur, S., Huberman, B., Predicting the Future with Social Media, HP Social Computing Lab
http://www.hpl.hp.com/research/scl/papers/socialmedia/socialmedia.pdf

Menon, A. K., Elkan, C., Link Prediction via Matrix Factorization, University of California
http://cseweb.ucsd.edu/~elkan/ECML2011LinkPrediction.pdf

Becker, H., Gravano, L., Beyond Trending Topics: Real-World Event Identification on Twitter, Columbia University and Naaman, M., Rutgers University
http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/viewFile/2745/3207

LaMonica, P.M. and Waskiewicz, T.V. (2011). Layered Multi-Modal Network Analysis of Textual Data for Improved Situation Awareness. http://www.lidi.info.unlp.edu.ar/WorldComp2011-Mirror/IKE5160.pdf

Robustness of Centrality Measures for Small-World Networks Containing Systematic Error

Amanda Lannie

Analytical Systems Branch, Air Force Research Laboratory, NY, USA

Abstract—Social network analysis is a necessary aspect of learning terrorist/criminal networks, or of how people interact through social networking sites like Facebook and Twitter. The metrics used to evaluate these networks are equally as important, centrality measures being the most common. Being the most routinely used metrics, centrality measures have been tested under various conditions. However, there is a lacking focus on systematic error applied to computer generated connected undirected small-world networks. The goal of this paper is to test the centrality measures (betweenness, closeness, and degree) in such networks and test their robustness after the introduction of systematic error. The experiment found that the metrics performed poorly and their use would have to be determined based on the situation and consequences.

Keywords: centrality; systematic error; small-world networks

1 INTRODUCTION

The centrality measures betweenness, closeness, and degree are common metrics for measuring aspects of a social network. Betweenness is the measure of the number of nodes between a pair of nodes. In [1], Freeman claims that to calculate betweenness one must compute:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}. \qquad (1)$$

σ_{st} is the total number of paths from node s to node t. $\sigma_{st}(v)$ is the number of σ_{st} paths that contain node v. Explicitly, Equation 1 is the sum of the quotients ($\sigma_{st}(v)$ divided by σ_{st}) for all pairs of nodes. The value computed represents a node's ability to control the flow of information. Closeness measures a node's number of contacts including both direct and indirect ones; thus, the measure is a representation of how well connected a node is. The formula for calculating closeness from [2] is:

$$C_C(v) = \frac{1}{\sum_j^n d_{vj}}, \qquad (2)$$

where d_{vj} is the shortest path from v to j. Degree is the number of direct contacts a node has. According to [2], the equation to compute degree can be defined as:

$$C_D(v) = \frac{\sum_j^n a_{vj}}{n-1}. \qquad (3)$$

The a_{ij} is representative of an element in an adjacency matrix representation of the graph. The more direct connections a node has the more possibilities the node has to send/receive information.

Betweenness, closeness, and degree are sensitive to changes made to a social network. Consider Figure 1 and assume the network shown is complete. Now look at Figure 2; it is missing the black node in the previous figure. That is four missing connections and now the node with the most connections is missing. In addition, the betweenness and closeness values for Figure 2 would differ from Figure 1. The values would be misleading and possibly detrimental to someone using the measures to focus on important nodes (people) like someone tracking a criminal or terrorist network. Therefore, in the case of Figure 2, the most important person according to the degree measure is missing.

Besides real world scenarios, it is crucial to test the robustness of centrality measures since they are the most frequently used metrics for social network analysis. Our contribution is to determine the robustness of betweenness, closeness, and degree when it comes to small-world networks with systematic error introduced which will create bias in the subgraphs. According to [3], systematic error is error that is not determined by chance but is introduced by an inaccuracy, by observation or measurement, inherent in the system. This research uses connected small-world networks for the ground truths to simulate real world data as closely as possible. A ground truth network represents a complete network, one without missing information like a person or relationship. To mimic cases of missing information and bias (introducing systematic error), we use subgraphs of the ground truths. The creation of subgraphs involves choosing one node as the starting point and like real world scenarios the further away a node is from the starting point the less likely it will be discovered. The setup is analogous to tracking a criminal/terrorist network, and the goal was to study the performance of the metrics used on the subgraphs, specifically their robustness.
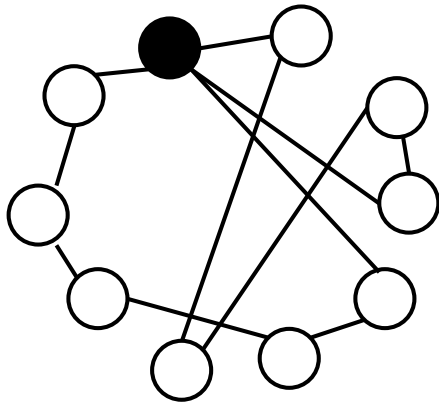
Fig. 1. A connected undirected small-world network where the number of neighbors was originally two. The black node is the most important person in terms of degree.

2   RELATED WORK

In 1988, John M. Bolland published a paper that explores the robustness of the centrality measures (betweenness, closeness, degree, and eigenvector). He used a real world network and manipulated versions of it. Bolland then inserted random systematic variation into the data from which he was able to make several conclusions. Betweenness is usually unstable when up against random error and is sensitive to systematic variation. Secondly, closeness is susceptible to both random and systematic variation on the outer edges of a network. Bolland also concluded that degree is useful in its ability to concurrently ignore random variation while reacting to systematic variation [4].

Costenbader and Valente also dealt with real data, specifically 59 networks. Their approach uses bootstrap sampling processes to establish how sampling influences the stability of 11 different network centrality measures [5]. Of the 11, among them were betweenness, closeness, and degree. From their research, Constenbader and Valente were able to conclude that betweenness and degree had lower correlations for an undirected graph than a directed one where the correlation is between actual and sampled measures. Additionally, closeness has superior performance for directed graphs when compared to undirected ones.

In 2006, Borgatti, Carley, and Krackhardt tested the

Fig. 2. Figure 1 with missing information, the black node and its connections.

robustness of degree, betweenness, closeness, and eigenvector centralities when it came to a large sample of random graphs. The authors of [6] control the amount of error in the observed graphs by adding or removing edges or nodes. They concluded that the measures are robust if the error is less than or equal to 10%. They were of the mind that it would be difficult to tell if such accuracy (61% or better depending on the amount of error of 10% or less) would be acceptable since it would depend on the situation and consequences.

Unlike [6], this research focuses on the robustness of centrality measures under systematic instead of random

TABLE I. WHY THERE ARE 171 GROUND TRUTHS WHEN THERE IS ONE SET FOR EVERY NODE LEVEL.

Number of Neighbors to Control Density (#) means the actual density of the graph						
	8% Density	*16% Density*	*24% Density*	*32% Density*	*40% Density*	*48% Density*
Node Level						
10			2 (22)		4 (44)	
25	2	4	6 (25)	8 (33)	10 (41)	
50	4	8	12	16	20	24
100	8	16	24	32	40	48
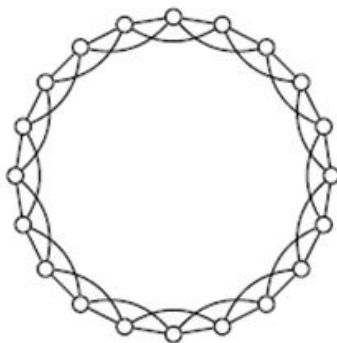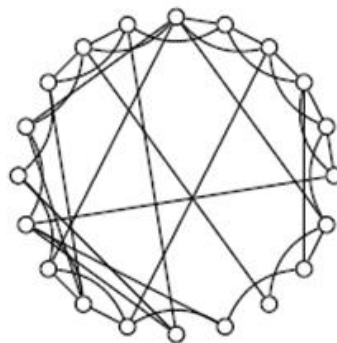
(a) $p = 0$, k-circulant   (b) $p = 0.3$, small-world   (c) $p = 1$, random
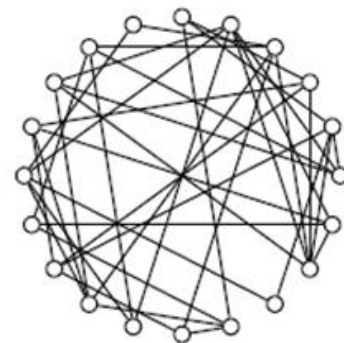
Fig. 3. With increasing randomness, k-circulant graphs evolve to exhibit properties of random graphs in G(n, p). Small-world networks are intermediate between k-circulant graphs and random graphs in G(n, p) [7].

Algorithm 1 Altered Breadth First Search – 1st pass

/* origGraph is a graph and a member to the class to which these methods belong */
Input: start node
Output: hops
Create hashtable visited, queue current, queue nextIt
Set hops to 0
Add start node to visited
origGraph.getNeighbors(start node)
for each neighbor of the start node
    Add neighbor to current
end for
while current and nextIt aren't empty
    while current is not empty
        Remove node from current
        Add node to visited
        origGraph.getNeighbors(node)
        for each neighbor of the node
            if current, nextIt, and visited doesn't contain the neighbor **then**
                Add neighbor to nextIt
            end if
        end for
    end while
     Increment hops
    Copy contents of nextIt to current
end while

---

error. Additionally, different from [4] and [5], the networks used are not based on real data, but are created from an algorithm to generate small-world graphs. Furthermore, [4] introduces systematic variation into the ground truth network by adding randomly-selected links to four different nodes. In contrast, we introduce a bias during the creation of subgraphs that comes from a focus on one specific node and nodes further away from the start node are less likely to be added to the subgraph.

3 METHODOLOGY

Nine hundred sixty-three ground truth networks (connected undirected small-world graphs) were made using Python 2.7 and NetworkX ([7]). A ground truth network has 10, 25, 50, or 100 nodes. For each of those possibilities, a graph was generated with the probability of rewiring edges of 0.1 - 0.9 with increments of 0.1. The higher the probability the more likely edges are to change thus creating a more random graph instead of maintaining its circulant form, see Figure 3. When rewiring an edge, one end remains the same and the other receives a new node; this introduces randomness into the network. For each generated subgraph, there would be a density of approximately 8%, 16%, 24%, 32%, 40%, and 48%, constraints permitting. NetworkX's method uses a third parameter, number of neighbors, to control the density of the graph. Neighbors are the number of nodes every node has a connection to. The algorithm in [8] considers the basis for a small-world graph is a k-circulant graph. According to [8], k is the number of neighbors every node has and must be even. Choosing four neighbors would

---

Algorithm 2 Altered Breadth First Search – 2nd pass

/* origGraph is a graph and a member to the class to which these methods belong */
Input: start node, hops, thresholdIncrements
Output: subgraphNodes
Create hashtable subgraphNodes, hashtable visited, queue current, queue nextIt
Set hops to number from 1st pass
Add start node to visited
origGraph.getNeighbors(start node)
for each neighbor of the start node
    Add neighbor to current
end for
while current and nextIt aren't empty
    while current is not empty
        Remove node from current
        Add node to visited
        origGraph.getNeighbors(node)
        for each neighbor of the node
            if current, nextIt, and visited doesn't contain the neighbor **then**
                Add neighbor to nextIt
            end if
        end for
    end while
    Generate a random number between 0 and 1
    if generated number >= (1 – (thresholdIncrements * (hops + 1)) **then**
        Add node to subgraphNodes
    end if
    Decrement hops
    Copy contents of nextIt to current
end while

---

mean that for each node, the node has connections to the two nodes on its left and the two on its right. The number of neighbors also has the constraint that they be less than Equation 4.

$$k < \frac{|\text{nodes in the graph}|}{2} \qquad (4)$$

Any additional neighbors than that and the graph will contain duplicate edges. After the generation of the k-circulant graph, the method would randomly choose edges to rewire. One full set is 171 ground truths; for a view of the setup to determine this number see Table I. However, there was concern of a bias towards the subgraphs generated from the 100 node cases.

The experiment considers every node in a ground truth as a starting point from which to create a subgraph. This is why there is a need to generate more cases for ten, twenty-five, and fifty node cases. As Table II shows, the experiment has thirty-four sets of ground truths for 10 node cases, five for 25 nodes, and two for 50. Thus, the number of instances with a ground truth of 10 nodes became 20,533; 18,606 for 25 nodes; 21,595 for 50 nodes; and 21,600 for 100 nodes. The number of samples does not include cases where a subgraph had no nodes or was exactly like a ground truth network. The additional instances allows for a more even distribution among the node levels, which will remove the bias when computing overall measures. For instance,

without the correction, if we compute the betweenness at every percent error value, then we would get values closer to the results of the 100 nodes cases rather than ones that represent all four node levels equally. Writing a plug-in for Gephi, open source software for graphs that allows the user to visualize and manipulate them, we were able to create all of the subgraphs.

The creation process of a subgraph inserts the systematic error. Each node from the ground truth is chosen as the focal point for the subgraph. As the focal point, the other nodes surrounding it are considered to have less visibility and thereby less probability of being included in the subgraph, introducing bias. If a node is not included, then neither is its connections. This is different than [6]'s approach of determining the robustness of centrality measures on imperfect data where random nodes and edges were added or removed to create a new graph.

The algorithm to create a subgraph uses a slightly altered breadth first search (BFS) for connected graphs; see Algorithms 1 and 2. Instead of using one queue, the modified breadth first search uses two. One queue is for the current level of nodes for the ground truth and the second is for the next level out. A subgraph generation requires two passes through the amended BFS.

The first pass (Algorithm 1) determines the number of hops (steps from the start node to the outer most node(s)). Hops or steps are the maximum number of edges it takes to visit all nodes. In Algorithm 1, every time the queue for the current level is empty, the number of hops is increased by one. Figuring out the number of steps determines the possibility of keeping (visibility) of each node. If there were four hops and the initial visibility was one, then the probability of keeping a node decreases by 0.2 at each step. Dividing the initial visibility (0.25, 0.5, 0.75, and 1, each is applied to all ground truths) by number of steps plus one allows for the possibility of keeping the outer most node(s) instead of removing them all together. Additionally, the start node has the possibility of not being included in the final subgraph.

During the second pass (Algorithm 2), all nodes receive a visibility based on the number of steps it is from the start node. In addition, each node receives a number that represents its actual visibility. If the actual number of steps is greater than or equal to one minus the visibility the node is included in the subgraph. After deciding for every node in the ground truth to keep it or not, the edges that were connected to the removed nodes are also removed. The leftover nodes and edges make up the resulting subgraph.

For every ground truth, the Gephi plug-in generates two output files. One contains the centrality values (betweenness, closeness, and degree) for each node and the other has the absolute difference of the subgraph values and ground truth values. The centrality values come from a statistics plug-in that came with the Gephi code. Production of the betweenness values is different than Equation 1. Instead, the creators of Gephi decided to use the algorithm in [2] to improve the speed of the calculations; the authors claim to compute the exact answer produced by Equation 1 unlike other methods that reduce computation time which only produce an approximate answer. Like betweenness, Gephi's closeness computations come to the same answers, but use an alternate algorithm. Gephi uses the algorithm from [2] that determines shortest paths. For degree, Gephi uses a node's degree. In addition to the new centrality files, a delta file contains the metrics "Top 1," "Top 3," and "Top 10%" from [6].

- Top 1: Proportion of times that the most central node in the true network is also the most central node in the observed network.
- Top 3: Proportion of times that the most central node in the observed network is among the top three most central nodes in the true network.
- Top 10%: Proportion of times that the most central node in the observed network is among the top ten percent of nodes in the true network.

The measures were a way of determining if the top person in a subgraph was in the set of top people in the corresponding ground truth. For the cases where multiple nodes had the same value, all nodes are included in the set of "top" nodes. The "top" measures consider this case during computation.

4 RESULTS

A great deal of data was generated for and from this experiment. As mentioned before, 963 ground truths were created using NetworkX. From those, Gephi was able to create 82,334 subgraphs. The breakdown based on node level is in Table II.

According to Figure 4, by increasing the number of nodes in a subgraph, it reduces the amount of error within it. We use

$$\% \text{ Error} = \frac{|\text{nodes in ground truth}| - |\text{nodes in subgraph}|}{|\text{nodes in ground truth}|} \quad (5)$$

to compute the amount of error in a subgraph. Additionally, the more nodes in a subgraph, the more accurate the centrality values are.

According to Figure 5, regardless of centrality measure, the more error in a subgraph the less accurate the centrality values were. Inaccuracy in Figures 5, 6, and 7 is calculated as

$$\text{Inaccuracy} = \frac{|\text{ground truth value} - \text{subgraph value}|}{\text{ground truth value}} \quad (6)$$
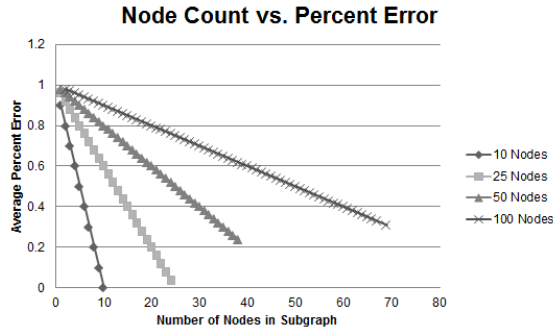
Fig. 4. The average percent error for each number of nodes in a subgraph for every node level.

Fig. 5. The average amount of inaccuracy for percent error shown for each centrality measure.
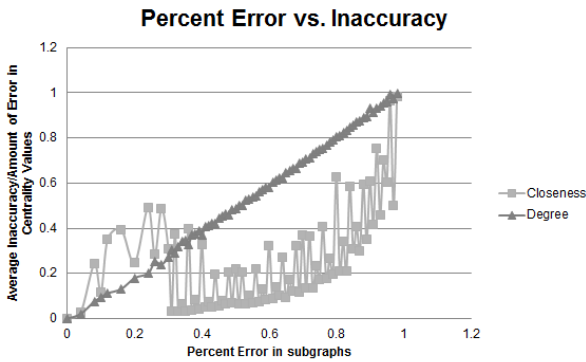
Fig. 6. A closer look at Figure 5's closeness and degree.

and is a representation of how different the subgraph's value was to the ground truth's and how that difference compares to the ground truth's value. Betweenness had the worst performance because at 60% error the node levels started to have different amounts of inaccuracy, see Figure 7. The nodes in subgraphs, generated from ground truths with 50 and 100 nodes, had vastly different values in comparison to the ground truth values. Those cases had to compensate for a great deal of missing nodes. A subgraph made from a 50 nodes ground truth, according to the data and as seen in Figure 4, could contain 1 to 38 nodes. Figure 4 also shows that for 100 node cases, the subgraphs had anywhere from 2 to 69 nodes. We suspect the gap comes from how the subgraphs are generated and how the hops played out.

Fig. 7. The average amount of inaccuracy for each percent error shown for every node level.

TABLE II. NUMBER OF SUBGRAPHS GENERATED FROM 963 GROUND TRUTHS. IT DOES NOT INCLUDE CASES WHERE THERE WERE NO NODES OR THE SUBGRAPH WAS THE SAME AS THE GROUND TRUTH.

Number of Nodes in Ground Truth	Number of Subgraphs Generated for Node Level	Number of Sets Created to Avoid Bias
10	20,533	34
25	18,606	5
50	21,595	2
100	21,600	1
Total	82,334	

By observing Table III, we can see that the top person in the subgraphs is hardly ever in the top portion in the ground truths. Table IV compares our results to Borgatti et al's results for the 100 nodes, 50% density, and 50% error removed nodes case. Our values are from 100 nodes, 50% error, and 48% density; it is the case closest to Borgatti et al.'s. In comparison to Borgatti et al.'s case, our numbers are lower, significantly so for Top 3 and Top 10%. They concluded that the measures are robust for networks that contain error of 10% or less, even then whether the levels of accuracy are sufficient for any given purpose depends on external factors such as the consequences of error [6]. Since the accuracy levels for this experiment are lower, this indicates that betweenness, closeness, and degree are not robust when used on small-world networks containing systematic error.

5   CONCLUSION

Based on Table III, it appears that the information about the "top" nodes in the ground truths is lost when looking at subgraphs. Not only that, according to Figure 5, the centrality values for the individual nodes in the subgraphs vary from the ground truth values, particularly betweenness.

The accuracies in Table III prove that the centrality measures of betweenness, closeness, and degree are not robust enough to withstand missing information (systematic error) in connected undirected small-world networks. At least, they should not be used if the situation and consequences could not handle such low accuracy. Additionally, while Borgatti et al. found the measures to be

robust for small measures of error, based on Figures 5 and 6 this is not the case for this experiment.

6 FUTURE WORK

Despite all the research done on the robustness of betweenness, closeness, and degree, it could be furthered still in the following ways. Firstly, an experiment could be removing edges only instead of nodes. This could then be compared to Borgatti et al.'s case for removed edges. Such an experiment would test the robustness of the centrality measures for the case where all people are known but not all of the relationships between them are. Additionally, another way to expand upon this paper's research is to always keep the start node. By doing this, one could use Bolland's metrics of local and configural sensitivity and be able to compare those results with his data. This study would determine how sensitive biased graphs (social networks where their creation starts with a focus on one person) are and possibly where. Finally, investigating the robustness of centrality measures when focusing on the most and least important nodes could further this study. We could determine how well those measures perform if the network was created from the most or least important person.

REFERENCES

[1] L. C. Freeman, "A Set of Measures of Centrality Based on Betweenness," Sociometry, vol. 40, 1977, pp. 35-41.

[2] U. Brandes, "A Faster Algorithm for Betweenness Centrality," Journal of Mathematical Sociology, vol. 25, 2001, pp. 163-177.

[3] (2013, May 28). *Systematic error* [Online]. Available: http://www.merriam-webster.com/dictionary/systematic%20error

[4] J. M. Bolland, "Sorting Out Centrality: An Analysis of the Performance of Four Centrality Models in Real and Simulated Networks," Social Networks, vol. 10, Sept. 1988, pp. 233-253, doi:10.1016/0378-8733(88)90014-7.

[5] E. Costenbader and T. W. Valente, "The Stability of Centrality Measures When Networks are Sampled," Social Networks, vol. 25, Oct. 2003, pp. 283-307, doi: 10.1016/S0378-8733(03)00012-1.

[6] S. P. Borgatti, K. M. Carley, and D. Krackhardt, "On the Robustness of Centrality Measures Under Conditions of Imperfect Data," Social Networks, vol. 28, May 2006, pp. 124-136, doi:10.1016/j.socnet.2005.05.001.

[7] (2011, Oct. 21). *NetworkX* [Online]. Available: http://networkx.github.com/

[8] D. Joyner, M. Van Nguyen, and N. Cohen, Algorithmic Graph Theory. 2011. http://code.google.com/p/graphbook/

TABLE III.   AVERAGE ACCURACY

	10 Nodes	25 Nodes	50 Nodes	100 Nodes	All Node Levels
Top 1	0.33	0.18	0.10	0.06	0.17
Top 3	0.63	0.36	0.23	0.14	0.34
Top 10%	0.34	0.36	0.32	0.33	0.34

TABLE IV.   ACCURACY RESULTS FOR GRAPHS OF 100 NODES AND 50% DENSITY AND 50% ERROR

	Betweenness		Closeness		Degree	
	Borgatti et al.'s Results	*Our Results*	*Borgatti et al.'s Results*	*Our Results*	*Borgatti et al.'s Results*	*Our Results*
Top 1	0.15	0.1	0.16	0.13	0.16	0.13
Top 3	0.33	0.21	0.34	0.19	0.34	0.19
Top 10%	0.63	0.45	0.64	0.46	0.64	0.46

Arabic Data Analysis using Classification Techniques

[1]Malik Abu Odeh, [2]Naser Abudalbouh

[1]Philadelphia University, Amman, Jordan, malekaboodeh@ymail.com

[2]Ministry of Interior, Management Dept, Amman, Jordan, nassir73a@yahoo.com

Abstract- *Text classification is one of the popular problems in classification data mining. It aims to mapping text documents into one or more predefined class or category based on its contents of keywords. This problem has recently attracted many scholars in the data mining and machine learning communities since the numbers of online documents that hold useful information for decision makers, are numerous. However, the majority of the research works conducted on text categorization is mainly related to English corpuses and little works have focused on Arabic text collections. Thus, this paper investigates the problem of Arabic text categorization using different rule-based classification approaches in data mining. Precisely, this research works attempts to evaluate the performance of different classification approaches that produce simple "If-Then" knowledge in order to decide the most applicable one to Arabic text classification problem. The rule-based classification algorithms that the paper investigates are: OneRule, rule induction (RIPPER), decision trees (C4.5), and hybrid (PART). The results indicate that the hybrid approach of PART achieved better performance that the rest of the algorithms.*

Keywords: Classification, Data Mining, Text Mining, Associative Classifiers

1. Introduction

The rapid developments in computer hardware have made the process of collecting and storing large quantities of data in all application's domains possible [10]. Nowadays, the number of documents scattered online by private and public sectors are in the orders of millions. These online documents may contain valuable information that decision makers can benefit from in decision making. The process of discovering and producing the hidden and useful information from these documents manually by domain experts is extremely hard and time consuming. This is since the numbers of online textual data are numerous and these data have large dimensionality. Therefore, using intelligent techniques to discover essential information automatically from textual documents may give companies the right decisions that work for improving their competitive advantages [16].

Text categorisation (TC) is the task in which texts are categorised into predefined categories based on their contents [14]. For example, if texts are represented as a research paper, categories may represent "Computer Science", "Mathematics", "Medicine", etc. The task of TC has various applications such as automatic email classification, web-page categorisation and indexing [6]. These applications are becoming increasingly important in today's information-oriented society especially with the rapid growth of online information, and therefore TC has become one of the key areas for handling and organizing textual data. As mentioned earlier, the goal of TC is the classification of documents into a fixed number of pre-defined categories in which each document can be in multiple, exactly one, or no category at all. Generally, TC task goes through three main steps: Text pre-processing, text classification and evaluation. Text pre-processing phase is to make the text documents suitable to train the classifier. Then, the classifier is constructed and tuned using a learning technique against the training data set. Finally, the classifier gets evaluated by some evaluation measures i.e. error rate, recall, precision, etc. Detailed description of these steps can be found in [15]. Different data mining and machine learning algorithms have been applied successfully to English text corpuses including decision trees [13], Support Vector Machines (SVM) [8], rule induction [3], probabilistic approaches [5], and others. However, little research works have been conducted on Arabic corpuses mainly since Arabic language is highly rich and requires special treatments such as order verbs, morphological analysis, etc . Particularly, in Arabic morphology, words have affluent meanings and contain a great deal of grammatical and lexical information [16]. Also, in syntactic structure, Arabic sentence formation differs from English. In this regard, the Arabic text documents are required significant processing to build accurate classification mode [9]. Therefore, few scholars have applied a number of classification approaches on the problem of Arabic text classification, i.e. Naïve bayes [7], SVM [8] and Decision tree [9]. However, researchers conclude that Arabic text classification is a very challenging task due to language complexity.

Most of the previous works on Arabic text mining attempt only to achieve the prediction accuracy from the above mentioned learning approaches. Some of domain experts have also interests in understanding the produced models that are formed as "IF-Then" patterns [16]. This is since such models are easy to interpret and can be modified by users manually [15]. Thus, this paper investigates different classification rule mining approaches related to the problem of Arabic text classification. Primarily, OneRule [19], rule induction (RIPPER) [3], decision trees (C4.5) [13], and hybrid (PART) [19] learning methods are applied to SPA Arabic data collection to measure their performance and effectiveness with reference to different text evaluation metrics such as error rate, and the number of derived rules.

The rest of the paper is organized as follows: Text classification and Arabic text mining related works are surveyed in Section 2. Section 3 is about the research methodology used, and Section 4 describes the Arabic data collection as well as the experimentations and results analysis, and finally the conclusions and further research are given in Section 5.

2. Text Categorization

TC is one of fundamental tasks of text mining in analyzing complex and unstructured data which is concerned about 'assigning of natural language texts to one or more predefined category based on their content. The concept of text classification has been firstly anticipated in early sixties [6] and it focused on indexing scientific journals using the vocabulary. Latterly, this research field has got more interest due to the fast growth of online documents that holds important and useful knowledge. Therefore, automatic text classification has turned into one of key domains for organizing and handling textual data Currently, there are many applications that are based on the text categorization including: document filtering, spam filtering, automatic metadata generation, classifying web resources under hierarchical catalogues and others [14].

2.1 Arabic Text Mining Related Works

EL-Kourdi et al. 2004) have used Naïve Bayesian classifier to classify an in-house collection of Arabic documents. The collections consist of five categories and 300 web documents for each category, and have used several partitions of the data set. They have concluded that there is some indication that the performance of Naïve Bayesian algorithm in classifying Arabic documents is not sensitive to the Arabic root extraction algorithm, in addition to their own root extraction algorithm; they used other root extraction algorithms. The average accuracy reported was about 68.78%.

[4] compared the performance of Naive Bayes, K-Nearest-Neighbors (KNN), and distance-based classifiers for Arabic text categorization. The collected corpus contains 1000 documents that vary in length and writing styles and fall into ten categories each category consists of 100 documents. The author used stemming to reduce the number of features extracted from documents. The recall, precision, error rate and fallout measures were used to compare the accuracy of classifiers. The results showed that the performance of Naïve Bayes classifier outperformed the other two classifiers.

[5] has used maximum entropy to classify Arabic documents. The data set collected from Aljazeera Arabic news channel and the documents categorized into sex domains: politics, sports, culture and arts, science and technology, economy and health. The results showed that using the preprocessing techniques (normalization, tokenizing and part-of speech) increases the F-measure from 68.13% to 80.41%.

[12] has implemented the SVM algorithm with the uses Chi square method as a feature selection method to classify Arabic documents. He has used an in-house collected corpus from online Arabic newspaper archives, including Al-Jazeera, Al- Nahar, Al-hayat, Al-Ahram, and Al-Dostor as well as a few other specialized websites. The collected corpus contains 1445 documents that vary in length. These documents fall into nine classification categories. The results showed that the SVM algorithm with the Chi-square method has outperformed Naïve Bayes and the KNN classifiers in term of F-measure. [2] have evaluated the performance of tow popular classification algorithms C5.0 decision tree [13] and SVM on classifying Arabic text using the seven different Arabic corpora such as (Saudi News Papers, WEB Sites, Arabic Poems). They have implemented a tool for Arabic text classification to accomplish feature extraction and selection. They have concluded that the C5.0 decision tree algorithm outperformed SVM in term of accuracy where as the SVM average accuracy was 68.65%, while the average accuracy for the C5.0 was 78.42%.

[18] have investigated different variations of vector space models and term weighting approaches using KNN algorithm, these variations are Cosine coefficient, Dice coefficient and Jaccard coefficient. A corpus of Arabic text documents was collected from online Arabic newspaper archives, including Al-Jazeera, Al-Nahar, Al-hayat, Al-Ahram, and Al Dostor. The average F1 results obtained against six Arabic data sets indicated that Dice based TF.IDF and Jaccard based TF.IDF outperformed Cosine based TF.IDF, Cosine based WIDF, Cosine based ITF, Cosine based log(1+tf), Dice based WIDF, Dice based ITF, Dice based log(1+tf), Jaccard based WIDF, Jaccard based ITF, and Jaccard based log(1+tf).

[1] have implemented and compared the KNN classifier one time by using N-Gram (word-level unigrams and bigrams) in documents indexing, and another time by using traditional single terms indexing method to classify Arabic documents. A corpus of Arabic text documents contains 1445 documents that vary in length. These documents fall into nine categories. They also used document frequency for feature reduction and Cosine measure to calculate the similarity of each document to be classified with training documents. The experimental results showed that the use of N-gram to represent each document produces better performance than using single terms. The average accuracy in case of using N-Gram is 74% while with Single terms indexing is 66%. [17] have evaluated the performance of Naïve Bayesian and SVM classifiers in Islamic Arabic data sets. The data set consist of 2244 Arabic documents of different lengths that belong to five categories. The recall, precision and F1 were used to compare the accuracy of classifiers. The results showed that the performance of SVM classifier outperformed the Naïve Bayesian classifier.

[17] have investigated Naïve Bayesian algorithm based on Chi Square features selection method to Categorize Arabic

Data. The data set consist of 1562 Arabic documents of different lengths that belong to sex categories. The experimental results provided evidence that feature selection (Chi Square) often increases classification accuracy by removing rare terms.

3. Research Methodology

The research is systematic process in which defining the objective, controlling the data, and communicating the findings take place within recognized frameworks and associated with existing guidelines. The research success depends on selecting an appropriate research methodology. Employing both qualitative and quantitative methods meets the purposes of this research. Quantitative approach is concerned about data collecting in quantitative form which can be subject to extensive quantitative analysis in a formal method [11]. This research strategy is applied for analyzing the experimentation results namely; precision, recall, error rate, and the number of generated rules derived from different classification approaches. Qualitative research involves subjective evaluation of approaches, opinions, and behaviour in which results are gained either in non-quantitative type or in the type which are not subjected to detailed quantitative analysis [11].

Since one of this paper's objectives is to conduct a literature review on Arabic text classification, the qualitative approach is considered as a suitable research approach to carry out this task effectively. Integrating both of these approaches into a single study is known as a mixed research methodology. The importance of employing this type of methodological approach is to achieve strong points and to reduce weak points of quantitative and qualitative research approaches.

4. Data and Results
4.1 SPA Data Set Description

SPA Arabic data collection is collected from (SPA site, 2008), and it consists of 1526 Arabic documents of different lengths that belong to six different categories (Economic "اقتصادية" , Cultural "ثقافية" , Political " سياسية" , Social "إجتماعية" , Sports " رياضية " , General " عامة"). Table 1 shows the number of documents for each category. Arabic text is different than English one since it is highly inflectional and derivational language which makes monophonical analysis a complex task. Also, in Arabic scripts, some of the vowels are represented by diacritics which usually left out in the text and it does use capitalization for proper nouns that creates ambiguity in the text. In this Arabic data set, each document file was saved in a separate file within the corresponding category's directory, i.e. this data documents are single-labeled.

Table 1: Number of Documents per Category (SPA)

Category Name	Number of Documents
Cultural News	258
Sports News	255
Economic News	250
Social News	258
Political News	250
General News	255
Total	**1526**

4.2 Experiments Details

The rule-based classification algorithms are depicted in Table 2. The selection of these algorithms is based on the different rule learning strategies they employ during the training step.

Table 2 The selected algorithms used in the experiments

Classification Approach	Classification Learning Algorithm	Algorithm Implementation Name in WEKA
Decision Trees	C4.5	Trees.J48
Simple Rule	One Rule	Rules.OneR
Rule Induction	RIPPER	Rules.JRip
Hybrid	PART	Rules.Part

All the experiments in this paper have been conducted using the WEKA software system on a Windows Vista Pentium IV 1.7 Ghz, 2 GB RAM PC.

We used the classification accuracy, precision, recall, and the number of rules as the base measures of comparison. Further, thorough analysis of the results produced is conducted in order to determine the applicable rule based classification algorithm to the problem of Arabic text mining. Ten-fold cross validation was utilized to derive the classifiers and their performance measure results in the experiments. This is a known testing method in data mining that works as follow: The input data set is randomly partitioned into ten parts where the class is represented in the same proportions as in each data part. Each part is tested in turn and the learning algorithm trained on the remaining parts; then the error rate is computed on the holdout part. The learning algorithm is executed ten times on different data parts, and the error rates are averaged in order to produce the final error rate.

4.3 Results and Analysis

Table 4 shows the average recall and precision results for the chosen classification algorithms. After analyzing the figures in that table, it is clear that PART algorithm outperformed the remaining classification algorithm (C4.5, RIPPER, OneRule) with reference to recall on the selected data set on average by 1.5%, 6.3%, and 35.9% respectively. In addition, it achieved higher precision than C4.5, RIPPER, and OneRule algorithms by 1.0%, 2.0%, and 53.0% respectively.

Table 4 the average recall and precision results

Classification Algorithm	Average Precision	Average Recall
OneRule	0.089	0.264
C4.5	0.609	0.608
RIPPER	0.599	0.56
PART	0.619	0.623

It is also obvious from Table 4 that OneRule algorithm is the least applicable classification method to the SPA data set due to the low numbers of recall and precision. This is since OneRule algorithms seeks for the rule that is associated with the largest frequency class in the training data set and derive it. From the SPA corpus, and for the OneRule classification algorithm, most of the categories are associated with no documents at except two categories, i.e. "Politics" and "Social", and therefore most of the documents belonging to the other categories have been misclassified to "Politics" and "Social" class labels. On the other hand, PART algorithm for instance derived 119 rules that represent most of the document categories in the SPA data set and correctly covered 855 out of 1526 documents. Overall, PART and C4.5 algorithms show competitive performance with reference to

precision and recall on the data set we considered since their generated results are very close to each other.

Figure 1 displays the accuracy results generated by the chosen algorithms against the SPA data set. This accuracy figures show that C4.5 and PART have close performance and that PART achieved a slightly better performance than C4.5 and the rest of the algorithms. Furthermore, again the classification accuracy of OneRule algorithm on the SPA data set is very low, e.g. 26.4%, which makes it the least applicable algorithm to such data. The reason behind the high error rate generated by the OneRule algorithm is that its outputted classifier represents very small portion of the data set that belong to just two categories ("Politics" , "Social") from a total of six categories. Thus when a new document is to be classified this algorithm often assigns to it either category ("Politics", "Social") which often leads to incorrect classification.
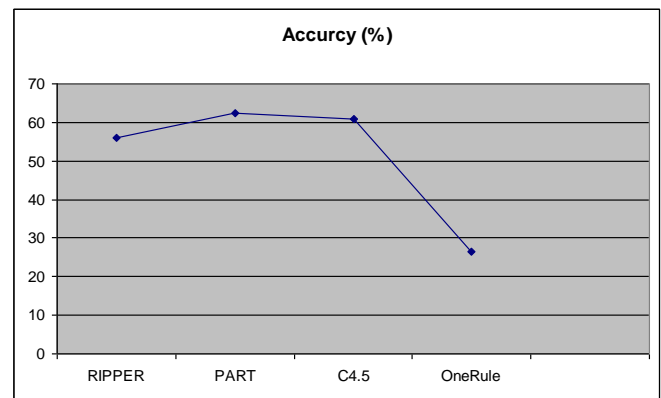
Figure 1: Accuracy (%) results produced by the contrasted classifiers

Figure 2 depicts the classifier size (number of rules) derived by the algorithms and it shows that C4.5 decision tree algorithm generated the largest number of rules whereas OneRule algorithm produced the least. In fact, the size of the classifiers generated by C4.5 is massive (417) and (PART, RIPPER) have derived moderate size classifiers competitive with respect to accuracy to C4.5. On the other hand, OneRule derived a very small classifier in size that lacks the fundamental feature for classification algorithm which is the ability to predict test documents and this expose its use for at least the SPA Arabic text collection.

5. Conclusions

In this paper, different classification data mining algorithms (C4.5, PART, RIPPER, OneRule) have been contrasted on the problem of Arabic text classification. Selecting these approaches is due to the fact that 1) they produce simple chunk of knowledge that end-user may interpret and understand easily, and 2) they employ different rule learning

strategies. Comprehensive experiments are conducted against known Arabic text collection called SPA with respect to different evaluation measures such as error rate, number of rules, etc, to determine the applicable classification approaches to Arabic text classification problem. The results revealed that the least applicable learning algorithm towards the chosen Arabic data set is OneRule. This is since OneRule algorithm produced only very limited numbers of rules and most of Arabic documents which belong to other rules have been misclassified. . Moreover, the most applicable algorithm to the Arabic data set is PART in which it derived higher results in all evaluation criteria than RIPPER, and PART, respectively. Moreover, the results of documents per category is generated for all the considered algorithms, and showed that the "Sport" category achieved the best results with respects to precision and recall on the SPA data set. we intend in near future to develop an language specific classification algorithm that can overcome the main challenges faced by the current classification algorithms.
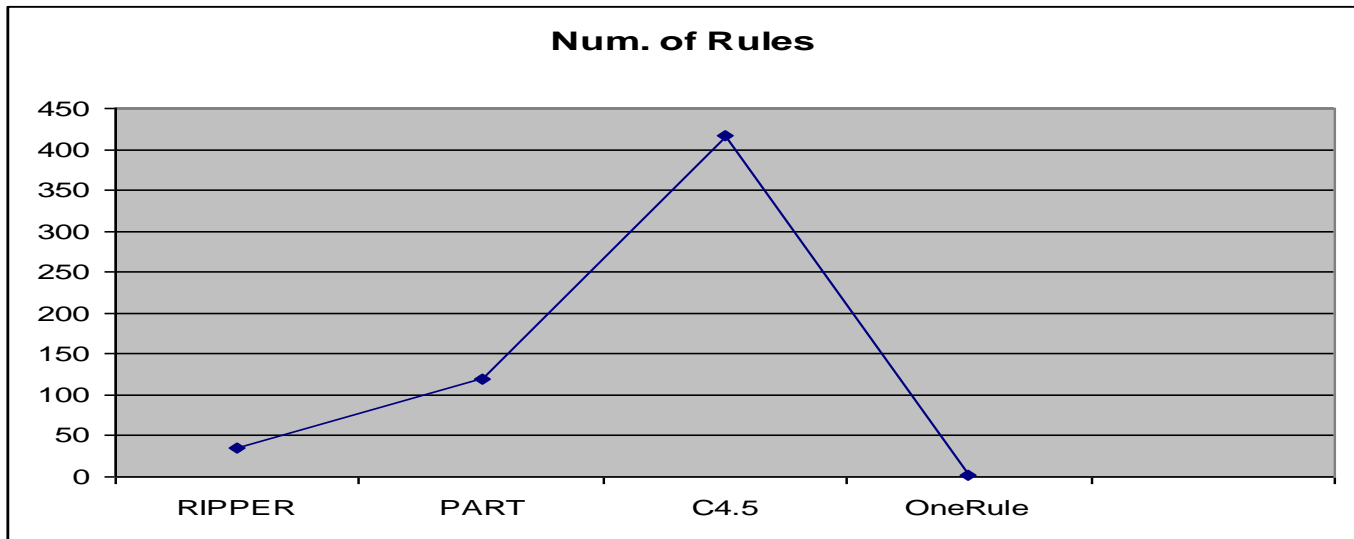
Figure 2: number of derived rules from the classifiers

References

1. Al-Shalabi R., Obeidat R., (2008). Improving KNN Arabic Text Classification with N-Grams Based Document Indexing. Proceedings of the Sixth International Conference on Informatics and Systems. March 27-29, 2008. Cairo, Egypt.

2. Al-Harbi, S. (2008) 'Automatic Arabic Text Classification', JADT'08: 9es Journées internationales d'Analyse statistique des Données Textuelles., pp. 77-83

3. Cohen, W. (1995) Fast effective rule induction. Proceedings of the 12th International Conference on Machine Learning, (pp. 115-123). CA, USA.

4. Duwairi, R. (2007) 'Arabic Text Categorization', International Arab Journal of Information Technology, Vol.4, No.2,

5. El-Halees, A. M.(2007) 'Arabic Text Classification Using Maximum Entropy', The Islamic University Journal, Vol. 15, No.1, pp 157-167,

6. Feldman, R. & Sanger, J., (2007) the Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data, NY: Cambridge University Press

7. Hadi W., Thabtah F., ALHawari S. and Ababneh J. (2008b). Naive Bayesian and K-Nearest Neighbour to Categorize Arabic Text Data. Proceedings of the European Simulation and Modelling Conference. Le Havre, France,(pp. 196-200).

8. Harrag, F. & El-Qawasmeh, E. (2009) 'Neural Network for Arabic Text Classification', IEEE

9. Harrag, F. El-Qawasmeh, E. & Pichappan, P. (2009) 'Improving arabic text categorization using decision trees', IEEE, NDT '09, pp. 110 – 115

10. Kantardzic, M. (2003) Data Mining: Concepts, Models, Methods, and Algorithms, John Wiley & Sons

11. Kothari, C. R. (2009) Research Methodology : Methods And Techniques, New Age Publication

12. Mesleh A. A. (2007). Chi Square Feature Extraction Based Svms Arabic Language Text

13. Categorization System. Journal of Computer Science 3(6): 430-435.

14. Quinlan, J. (1993) C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann.

15. Sebastiani, F. (2002) 'Machine learning in automated text categorization' ACM Publication: ACM Computing Surveys. Vol. 3(1) PP.1-47.

16.   Song, M. (2009) Handbook of research on text and web mining technologies, information science reference, IGI global

17.   Thabtah, F., Gharaibeh O., Alzubaidi R. (2011): Arabic text mining using rule based classification. Journal of Information and Knowledge Management, to be published in December 2011.

18.   Thabtah F., Eljinini M., Zamzeer M., Hadi W. (2009) Naïve Bayesian based on Chi Square to Categorize Arabic Data. Communications of IBIMA, volume 10, (PP. 158-163). CIBIMA.

19.   Thabtah F., Mahazah M., Hadi W. (2008) VSMs with K-Nearest Neighbour to Categorise Arabic Text Data. Proceedings of the International Conference on Machine Learning and Data Analysis 2008 (ICMLDA). Oct. 2008, San Francisco, CA, USA.

20.   Witten, I., H., & Frank, E. (2000) Data Mining: Practical Machine Learning Tools and Techniques, San Francisco: Elsevier Inc

Problems of the Arabic OCR: New Attitudes

Prof. O.Redkin, Dr. O.Bernikova

Department of Asian and African Studies, St. Petersburg State University, St Petersburg, Russia

Abstract - *his paper revie s e isting problems in the process of the Arabic . y analy ing the traditional approaches to te t recognition authors propose a model optimally considering the specificity of Arabic. is a phased process and must include stages of initial analysis of the picture files including possible positions of graphemes sequence in accordance ith the verbal or noun morphological paradigm, selection of the most li ely variants and decision on each of them in certain position. Ideally all the available databases should be used in the course of process, among them corpuses of te ts, e dictionaries, the Internet resources. he problem lies in the selection of the most relevant of them for the solution of a certain linguistic problem.*

.

Keywords: Optical Character Recognition (OCR), Arabic

1 Introduction

Optical Character Recognition (OCR) is among the most important problems of computer programming and mathematical linguistics. Most of the computer solutions designed for the Arabic OCR are based on the linear and vertical segmentation of the text with a subsequent identification of its separate components. The initial linear segmentation (lines) and the vertical segmentation (sentences, words and letters) is based on the use of the available information related to the nature of pixels of the *. bmp files.

Although there is a sufficient amount of research on the problems of clusterization[1], the problem of OCR is far from its final solution. One of the reasons for it is that focusing on pure mathematical analysis restricts if not oversimplifies the

actual process of recognizing of the written text, and hence, limits the effectiveness of the software. Besides that it may also lead to errors, especially when the high level of random unwanted noises takes place (for example, insufficient quality of the handwritten or printed document - defects of paper or script, notes between the lines or on the original text, etc.). So it is necessary to understand the process of OCR broader than just a formal analysis and clusterization of the pixels.

The initial stage of the optical recognition must be followed by the analysis of the possible positions of graphemes sequence in accordance with the verbal or noun morphological paradigm, selection of the most likely variants and decision on each of them in a certain position. The selection of the correct characters in the range of the possible variants also depends on the results of lexical and syntax analysis within the framework of a syntagm.

The ideal software for recognizing acoustic or written text must simulate the actual processes of one's reading and decoding of the written symbols (characters). However, there is no universal program so far which simulates the entire model of the native speaker reading process.

Thus, the aim is to simulate the cognitive process of reading character recognition and its subsequent stages.

.

2 Peculiarities of the Arabic Text

The problem of the character recognition is twofold, and, along with the purely mathematical part there is a linguistic one. When solving such kind of problems collaboration of linguists and computer programmers gives good results in the creation of linguistically oriented computer programs if the project participants are well aware of difficulties and opportunities in both sectors of the task.

Today, there is the effective OCR software developed for Latin or Cyrillic texts. As for the Arabic script, the problem seems to be more complex than for the two cases mentioned above. To a certain extent it is caused by the peculiarities of the Arabic script, among which one can mention a big number of morphological and graphic derivatives, cursive stile of writing, which presents a continuous flowing line, joined-up writing of many prepositions, particles, etc. Sometimes the elements of the characters (the dots) may be presented in disperse, and may be located apart from the main element of the letters.

---

[1] Avros R., Granichin O., Shalymov D., Volkovich Z., Weber G.-W. Randomized algorithm of finding the true number of clusters based on Chebychev polynomial approximation (Chapter 6) // Data Mining: Found. & Intell. Paradigms, D.E. Holmes, L.C. Jain (Eds.), Berlin Heidelberg: Springer-Verlag, ISRL 23, 2012. Vol. 1, pp. 131–155.
Granichin O.N., Shalymov D.S., Avros R., Volkovich Z. A randomized algorithm for estimating the number of clusters // Automation and Remote Control, 2011. Vol. 72, No. 4, pp. 754-765.

The account of the peculiarities and difficulties may be followed by such factors as the variation of the certain written elements of the letters, size of the components, the presence of ligatures and the ability to write letters on the homepage, as well as differences in the shape of the elements of writing letters, slope, size, elements, etc. In addition, there are regional variants of the letters, for example ٻ instead of regular ف , and ڢ or ڧ instead of ق in Maghreb and letters used sporadically in dialectal or loanwords: چ , ڤ , پش.

Besides that, as it is well known in the dialects the interdental spirants can be realized as the corresponding explosive such, instead of ذ, ت instead of ث, and spirants instead of affricates, for example, چ instead of ج, ض or ط instead of ظ, - the facts which are reflected in written texts.

The problem of recognizing handwritten documents, especially manuscripts, which include the individual characteristics of the authors' handwriting, is even more complex, not to mention the extra noises – notes of the scribes, defects of the written material, and lacunae and gaps in the text, notes and additions to the original text. All this makes the correct identification of the Arabic written texts extremely difficult.

So, despite the abundance of works and research materials dedicated to the problem of the Arabic OCR made up today, the problem is still far from to be resolved.

In order to provide accuracy in the Arabic text recognition a set of special methods should be used along with the classical attitudes for optical character recognition. Among them there are such as developing of the grammatical and lexical analyzers, consideration of such useful data as the relative frequency of use of letters or words, the comparison with existing databases, such as e-dictionaries, software lemmatization, etc.

3 Formal Indicators for OCR

Among the formal markers that facilitate the task of optical character recognition - are the spaces between words which allow to define the beginning and the end of the words. Besides that such characters as tā'-marbūtah ة or 'alif lām ال of the definite article also indicate the beginning or the end of the word respectively. The minor spaces may help to identify such characters as ا 'alif, د dāl, ذ dhāl, ر rā', ز zāyn in medial position within the word.

The frequency of characters or words use may be also regarded as one of the clues which affects the identification of the written elements. Thus the most frequent characters are more likely in certain position. As for the account of the frequency of letters, the analysis of the text of the Qur'an gives the following results:

ab. . requency of letters

Character		Count
'alif	ا	62381
lām	ل	38102
nūn	ن	27268
mīm	م	26735
wāw	و	25676
yā'	ي	19143
hā'	ه	14850
rā'	ر	12403
bā'	ب	11491
tā'	ت	10520
kāf	ك	10497
'ayn	ع	9405
fā'	ف	8747
qāf	ق	7034
'alif maksūra	ى	6605
sīn	س	6010
dāl	د	5991
dhāl	ذ	4932
ḥā'	ح	4140
djīm	ج	3317
hamzah	ء	3059
khā'	خ	2497
tā' marbūta	ة	2344
šīn	ش	2124
Sād	ص	2074
Dād	ض	1686
zayn	ز	1599
thā'	ث	1414
Tā'	ط	1273
ghayn	غ	1221
Dthā'	ظ	853

3.1 The most frequent words

As for the most frequent words the authors have carried on analysis of the texts including some one million words and the most frequent words are:

ab. . he most frequent ords in Arabic

و	conj. "and"
في	prep. "in"
من	prep. "from"
على	prep. "and"
أن	conj. "that"
التي	"which"
إلى	prep. "to"

2

http://corpus.quran.com/java/example/characterfrequencyexample.jsp

هذا	"this" (masc.)
ما	"what"
عن	prep. "about"
شمس	"sun"
هذه	"this" (fem.)
عام	"year"
لا	"no"
مع	prep. "with"
ذلك	"that"
قمر	"moon"
هو	"he"
كان	"to be"
كل	"each, all"

د	ذ
ذ	ض، ط، ظ ص،
ر	ل
ز	ض ،ص، ظ
س	ص ،ض
ش	ض ،ل
ص	ض، ط، ظ
ض	ط ،ظ
ط	ظ ،ك
ظ	غ ،ق
غ	ق ،ك
ق	ك ،غ
ل	ن

Statistical analysis is also one of the most effective methods used for the processing of the material in other languages, for example, Vietnamese.[3]

3.2 Linguistic Context

As it was mentioned above in addition to the methods of optical character recognition, in addition to form and grammatical and lexical analyzers, and consideration of other useful data, such as the relative frequency of use of letters or words, it is important to use available databases, such as e-dictionaries, e-translators, morphological and lexical analyzers, etc.

A decision on the choice of a letter can be made subject to the rules of compatibility and incompatibility of letters in Arabic.[4]

ab. . ompatibility and incompatibility of letters in Arabic

Character 1	Characters incompatible with Character 1
غ	خ ،ح ع،
ع	خ ،ح غ،
خ	ع، ح غ،
ح	ع، خ غ،
ب	م ف،
ت	ث
ث	س ،ص ،ض، ط
ج	ف، ق ،ك
خ	ق، ك ظ،

Moreover, unlike in most of the Indo-European languages where **the text** has linear character and represents the sequence of graphemes which correspond to certain phonemes, in Arabic the degree of hieroglyphic style of the text increases and, due to consonantal type of writing each word may represent various grammatical forms.

The lower is the clarity and sharpness of writing is, the greater is the variability of the recognition and comprehension of its characters. Consonantal script without diacritics, which existed at the early stage, allowed a vast variety of characters identifications. Thus the same letter or word form could be recognized, understood and read in different ways. All this relates to old types of Arabic script, for example, ancient *ufi* script and the first written texts of the Qur'an.

Picture . he ur an ufi script .

---

[3] Hieu L. T., Granichin O. N. Using application of statistics for word extraction from Vietnamese documents // Vestnik Sankt-Petersb. Univ. App. Math., 2009. No. 3, pp. 162-170.
[4] Yushmanov N.V. Grammatika literaturnogo arabskogo yazyka (The Grammal of Literary Arabic). St. Petersburg, 1999. P. 21.

[5] http://religionnerd.com/2010/10/05/traces-of-the-calligrapher/.
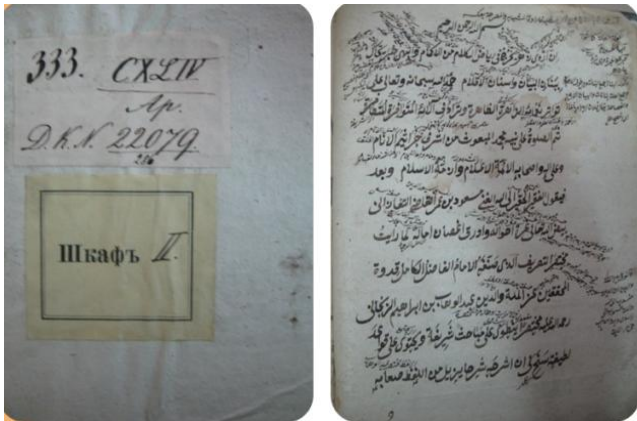
Picture 2. A fragment of the manuscript from the Manuscripts ollection at the epartment of riental and African tudies, aint Petersburg tate niversity. ommentaries to A injani rammar by At anta ani.
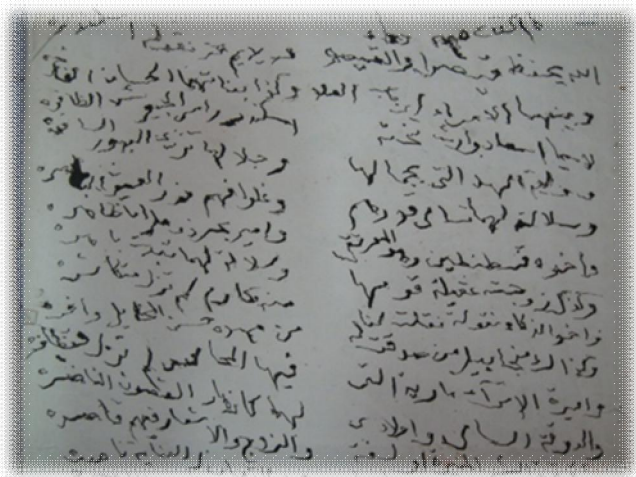
Picture . A fragment of the manuscript from the Manuscripts ollection at the epartment of riental and African tudies, aint Petersburg tate niversity. At anta i s ollection.

Meanwhile variability of reading and comprehension of sacred texts, such as the text of the Qur'an is unacceptable, so diacritics (dots) and special signs for vowels were introduced later on.

At the time of writing is not only omission of vowels, but the point above or below, making it difficult to adequately understand and read the text simultaneity of such a process.

Thus the results of the first phase of the optical character recognition should be analyzed regarding the available databases and considering all possible derivates of the certain root and variants of word forms, which are possible

in this position. So, another important condition of successful and comprehensive OCR is the analysis of grammatical and lexical context, including the semantics of the words, and, hence, modeling of all possible linguistic environments.

Thus the OCR process along with the analysis of the linear sequence of its constituent units (sentences, words and, finally, letters) must include grammatical and lexical analyzers, i.e. the available lexical and morphological databases as well as take into consideration relevant linguistic and extralinguistic context clues.

3.3 Extralinguistic Context

In many respects the OCR software must resemble the cognitive process of the native speakers which is based on the basis of his/her prior linguistic and practical knowledge and experiences, which affects recognition and comprehension of the written text and have options for self learning.[6]

Thus, the linguistic experience of a native speaker of the texts makes it possible to understand the words written unclear or in the defective way, for instance, including the omission of vowels, misspellings, etc., which, however, has only a negligible impact on their perception of the text or adequacy of reading. In a certain sense, the native speakers reading is heuristic by its nature, and is the result of many years of linguistic practice and, ultimately, fluent reading.

Thus, in the course of the dialogue the extralinguistic context gives grammatical information related to such grammatical categories as person, gender, number, along with the information related to the choice of the most appropriate communication style, etc.

In addition, the situation context and information related to the subject of the text gives clues related to the vocabulary used in it and thus limits the search fields in the e-dictionaries and databases.

4   Conclusions

OCR is a phased process and must include stages of initial analysis of the picture files including possible positions of graphemes sequence in accordance with the verbal or noun morphological paradigm, selection of the most likely variants and decision on each of them in a certain position. Ideally all the available databases should be used in the course of OCR process, among them - corpuses of texts, e-dictionaries, the Internet resources. The problem is in the selection of the most

relevant of them for the solution of the certain linguistic problem.

The software system must have access to extensive databases used for the identification of the characters, and finally, the word forms on the basis of the comparative analysis, and, if possible, be suitable for simulation of the model of actual text reading and comprehension.

4. References

[1]　　Avros R., Granichin O., Shalymov D., Volkovich Z., Weber G.-W. "Randomized algorithm of finding the true number of clusters based on Chebychev polynomial approximation (Chapter 6)", Data Mining: Found. & Intell. Paradigms, D.E. Holmes, L.C. Jain (Eds.), Berlin Heidelberg:, ISRL 23, Vol. 1, 131–155, Springer-Verlag 2012

[2]　　Redkin O.I. , Bernikova O.A., " On the Optical Character Recognition and Machine Translation Technology in Arabic"; Proceedings of the 2011 International Conference on Artificial Intelligence. Las Vegas, USA, 2011. Pp.861-867.

[3]　　Granichin O.N., Shalymov D.S., Avros R., Volkovich Z. A randomized algorithm for estimating the number of clusters // Automation and Remote Control, 2011. Vol. 72, No. 4, pp. 754-765.

[4]　　Hieu L. T., Granichin O. N. "Using application of statistics for word extraction from Vietnamese documents", Vestnik Sankt-Petersb. No. 3, 162-170, 2009.

[5]　　Yushmanov N.V. "Grammatika literaturnogo arabskogo yazyka" (The Grammal of Literary Arabic). St. Petersburg, 1999.

[6]　　Granichin, O.N.; Izmakova, O.A. "A randomized stochastic approximation algorithm for self-learning", Automation and Remote Control. Vol.66, Issue 8, 1239-1248, Aug. 2005.

[7]　　Redkin O.I., Bernikova O.A., Shalymov D.S. "Software for Arabic text e-learning, translation and recognition"; St.Petersburg, Russia, April 2007.

A Grey Prediction Based TOPSIS Decision Making for Product Form Design

Yang-Cheng Lin

Department of Arts and Design, National Dong Hwa University, 974, Hualien, Taiwan

Abstract – *The grey prediction (GP) has been developed to examine the relationship among factors with the uncertain and incomplete information. In addition, the technique for order preference by similarity to ideal solution (TOPSIS) has been highlighted by its wide use in multi-attribute decision making (MADM). As such, this paper is based on the process of kansei engineering in conjunction with GP and TOPSIS. Three GP models are used a basis to build a GP-based TOPSIS decision support database, which helps product designers obtain the optimal design alternatives that best meet consumers' preferences for a new characteristic toy design. This approach provides an effective mechanism for facilitating the process of the new product form design.*

Keywords: Product Form Design; TOPSIS; Grey Prediction; Decision Making; Kansei Engineering

1   Introduction

Aesthetics plays an important role in new product development, marketing strategies, and the retail environment [3, 8]. The Apple product has showed the visual appearance has become a major factor in consumers' purchase decisions [16]. The aesthetically pleasing properties have a positive influence on consumers' preferences of a product and their decision processes when consumers purchase it [13, 17]. In the field of product design, the "visual appearance/ aesthetics" is usually concerned with "product form" [5]. The product form is defined as the collection of design features that consumers will appreciate [4-6].

In order to explore the relationship between consumers' emotional feelings and product form elements, we conduct an experimental study on characteristic toys [11, 12], using kansei engineering (KE). KE is as an ergonomic consumer-oriented methodology and design strategies for affective design to satisfy consumers' emotional feelings or preferences [14, 15]. In subsequent sections, we first present the methodology proposed in this study, including the Grey Prediction (GP) due to its powerful learning and prediction abilities [1, 2], and the technique for order preference by similarity to ideal solution (TOPSIS) due to its wide use in multi-attribute decision making (MADM) [12, 18]. Then an empirical application on characteristic toys [11] is conducted to show how GP can be used a basis to construct the GP-based TOPSIS decision support database. It is built to help product designers get the optimal alternatives (ideal solutions)

that best meet consumers' preferences for the new product form design [11].

2   Methodology

This section presents the concept of GP and TOPSIS used in this study.

2.1   Grey prediction (GP)

The GP model uses a grey differential model (GM) to generate data series from the original data series of a dynamic system [1, 2]. The data series generated by the GM are converted back to the original data series by a reverse procedure to predict the performance of the system. Since the generated data series are more coherent than the original, the accuracy of the modeling is enhanced. The GM has three basic operations [1, 2]: (1) accumulated generation, (2) inverse accumulated generation, and (3) grey modeling. The accumulated generation operation (AGO) is used to build differential equations. The GM is usually represented as GM(M,N) for dealing with Mth-order differential equations with N variables [4, 10].

2.2   The technique for order preference by similarity to ideal solution (TOPSIS)

Based on the concept of the degree of optimality, the overall preference value of an alternative is determined by its distance to the positive ideal solution and to the negative ideal solution. This concept has been implemented by a widely used MADM method called the technique for order preference by similarity to ideal solution (TOPSIS) [9, 12, 18]. The advantages of using this concept have been highlighted by (a) its intuitively appealing logic, (b) its simplicity and comprehensibility, (c) its computational efficiency, (d) its ability to measure the relative performance of the alternatives with respect to individual or all evaluation criteria in a simple mathematical form, and (e) its applicability in solving various MADM problems [18].

3   An application of kansei engineering

In this section, we present the primary procedure of kansei engineering in the context of characteristic toys, including how to extract the representative experimental sample, how to conduct the morphological analysis of product form elements, and how to assess the preferences of

consumers. First, we collect about 179 characteristic toys and then classify them based on their similarity degree by a focus group which is formed by 6 product experts/designers with at least two years' experience of product design (please refer [7-9, 11, 12] for details). The focus group eliminates some highly similar samples through discussions. Then the hierarchy cluster analysis is used to extract the representative samples of characteristic toys. The 35 representative characteristic toy samples are selected by the cluster tree diagram, including 28 samples as the training set and 7 samples as the test set for building the GP model.

Furthermore, the morphological analysis is used to extract the product form elements of the 35 representative characteristic toy samples. The focus group is asked to decompose the representative samples into several dominant form elements and form types according to their knowledge and experience. Table 1 shows the result of the morphological analysis, with 7 product form elements and 24 associated product form types being identified (refer [7, 8] for details). Various design alternatives can be generated by different combinations of morphological elements.

Table 1 The result of morphological analysis

Form Elements	Form Types				
	Type 1	Type 2	Type 3	Type 4	Type 5
Length ratio of head and body (X_1)	>1:1	1:1~1:2	<1:2		
Width ratio of head and body (X_2)	Head> body	head =body	Head< body		
Costume style (X_3)	one-piece	two-pieces	robe		
Costume pattern (X_4)	simple	striped	geomet-ric	mixed	
Headdress (X_5)	tribal	ordinary	flowered	feathered	arc-shaped
Appearance of facial features (X_6)	eyes only	partial features	entire features		
Overall appearance (X_7)	cute style	semi-person. style*	person. style**		

* semi-personified style

** personified style

On the other hand, we collect about 110 image words which are used to describe the characteristic toys. Then we apply factor analysis and cluster analysis according to the result of the semantic differential method. Finally, 3 representative image words, i.e. "cute (CU)", "artistic (AR)", and "attractive (AT)", are determined (refer to [7, 8] for details). To obtain the assessed values for the emotional preferences of 35 representative characteristic toy samples, a 100-point scale (1-100) of the semantic differential method is used. 150 subjects (70 males and 80 females with ages ranging from 15 to 50) are asked to assess the form (look) of characteristic toy samples on an image word scale of 1 to 100, for example, where 100 is cutest on the CU scale (as showed in Table 2).

Table 2 The assessment of consumers' preferences

No.	X_1	X_2	X_3	X_4	X_5	X_6	X_7	CU	AR	AT
1	3	2	1	1	4	3	3	73	61	64
2	1	1	1	1	1	2	1	72	45	43
3	2	2	1	3	3	1	1	70	64	71
4*	2	3	2	4	2	2	2	63	52	54
5	2	2	1	1	4	2	1	68	59	55
6	2	2	2	4	3	2	2	65	66	69
7*	2	2	2	4	5	2	2	52	66	61
8	2	3	2	4	4	2	2	53	61	60
9	2	2	3	2	2	2	2	63	59	59
10	2	2	1	3	2	2	2	55	63	65
11	1	1	2	3	4	2	1	70	69	67
12	1	1	3	2	2	2	1	57	54	61
13	3	3	2	4	4	3	3	48	69	76
14	3	3	1	4	4	3	3	62	68	78
15	3	3	2	2	2	3	3	54	63	68
16*	3	3	1	2	3	3	3	62	74	72
17	3	3	2	4	2	3	3	55	68	66
18	2	3	3	2	2	2	2	71	65	61
19	2	2	1	1	2	3	3	41	52	75
20	2	2	2	1	1	3	3	39	53	63
21*	2	2	2	2	3	3	3	41	50	58
22	2	2	2	3	2	3	2	44	74	62
23	2	2	2	1	2	3	3	43	59	74
24	2	2	1	3	2	3	1	54	60	62
25	2	2	2	2	2	1	1	63	52	62
26*	1	2	2	2	4	3	2	58	71	68
27	1	2	1	2	4	3	2	57	61	66
28	1	1	2	2	1	1	1	62	56	73
29	1	1	1	3	5	3	2	76	67	74
30*	1	1	1	3	3	3	2	68	59	65
31	1	1	3	2	2	3	2	71	60	70
32*	1	1	1	4	4	1	1	61	49	51
33	1	1	1	4	5	1	1	72	59	57
34	2	3	2	4	2	2	2	38	48	49
35	1	1	1	3	5	2	1	78	59	79

The last 3 columns of Table 2 show the 3 assessed values of the 35 samples, including 28 samples in the training set and 7 samples in the test set (asterisked). For each selected characteristic toy in Table 2, the first column shows the characteristic toy number and Columns 2-8 show the corresponding type number for each of its 7 product form elements, as given in Table 1. Table 2 provides a numerical data source for building the GP model mentioned in Section 4, which can be used to develop a design decision support database.

4 The GP-based TOPSIS decision support database

This section presents the analysis of the GP model and the TOPSIS decision support model.

4.1 Analysis of the GP model

The GP is used as a technique for determining the optimal combination of product form elements for matching a desirable emotional preference. The 28 samples in the training set, given in Table 2, are used as the data set for building the GP model. As a GM(1, 8), the GP model uses the 7 form elements as the comparison series X_i and the average CU, AR, and AT values as the reference series X_0, respectively. As such, there are 3 GP models built, called CU-GP, AR-GP, and AT-GP model respectively, for representing 3 representative image words in this study (refer [11] for details).

For building each GP model, taking CU-GP model as an example, it can be described in the following steps:

Step 1: Denote the original sequence.

Step 2: Generate a new sequence for each series.

Step 3: Define the first-order differential equation with N variables and then obtain the parameters.

Step 4: Obtain the CU-GP model for predicting the CU value based on the 7 form elements is built by (1) as

The CU-GP model:

$$\hat{X}_0^{(1)}(k+1) = [0.73+0.95X_1^{(1)}(k+1)-0.75X_2^{(1)}(k+1)$$
$$-0.3X_3^{(1)}(k+1)-0.25X_4^{(1)}(k+1)+0.15X_5^{(1)}(k+1)+0.3X_6^{(1)}(k+1)-$$
$$0.4X_7^{(1)}(k+1)] \ e^{0.2k} -0.95X_1^{(1)}(k+1)+0.75X_2^{(1)}(k+1)+$$
$$0.3X_3^{(1)}(k+1)+0.25X_4^{(1)}(k+1)-0.15X_5^{(1)}(k+1)-0.3X_6^{(1)}(k+1)$$
$$+0.4X_7^{(1)}(k+1) \tag{1}$$

To follow the same process, the other two GP models are as follows:

The AR-GP model:

$$\hat{X}_0^{(1)}(k+1) =[0.61-0.75X_1^{(1)}(k+1)-0.46X_2^{(1)}(k+1)+$$
$$0.04X_3^{(1)}(k+1)+ 0.25X_4^{(1)}(k+1)-0.33X_5^{(1)}(k+1)-0.08X_6^{(1)}(k+1)$$

$$+0.17X_7^{(1)}(k+1)] \ e^{-0.24k} +0.75X_1^{(1)}(k+1)-0.46X_2^{(1)}(k+1)-$$
$$0.04X_3^{(1)}(k+1)-0.25X_4^{(1)}(k+1)+0.33X_5^{(1)}(k+1)+0.08X_6^{(1)}(k+1)-$$
$$0.17X_7^{(1)}(k+1) \tag{2}$$

The AT-GP model:

$$\hat{X}_0^{(1)}(k+1) = [0.64-0.94X_1^{(1)}(k+1)+0.56X_2^{(1)}(k+1)-$$
$$0.06X_3^{(1)}(k+1) +0.44X_4^{(1)}(k+1)-0.5X_5^{(1)}(k+1)-0.06X_6^{(1)}(k+1)$$
$$+0.06X_7^{(1)}(k+1)] \ e^{-0.16k} +0.94X_1^{(1)}(k+1)-0.56X_2^{(1)}(k+1)$$
$$+0.06X_3^{(1)}(k+1)-0.44X_4^{(1)}(k+1)+0.5X_5^{(1)}(k+1)+0.06X_6^{(1)}(k+1)-$$
$$0.06X_7^{(1)}(k+1) \tag{3}$$

With the GP models in (1), (2), and (3), product designers can input the values of the corresponding form elements, and then obtain the prediction values of the CU, AR, and AT emotional preferences, respectively.

4.2 The TOPSIS decision support model

The three GP models enable us to build a design decision support database that can be used to help determine the optimal product form for best matching specific consumers' preferences. The decision support database can be generated by inputting each of all possible combinations of form types on each form element to the three GP models individually for generating the associated preferences' values. The resultant design decision support database for characteristic toy consists of 4,860 ($=3\times3\times3\times4\times5\times3\times3$) different combinations of product form elements, together with their associated CU, AR, and AT preference values. In other words, there are 4,860 design alternatives generated by the three GP models [11].

The TOPSIS method is used to determine the optimal alternative (ideal solution) if the specific design requirement or concept is proposed by consumers or product designers. For example, if consumers prefer a new characteristic toy with "moderately cute", "moderately artistic", and "extremely attractive", the GP-based TOPSIS decision support method can be described in the following main steps [9, 12]:

Step 1: Obtain the decision matrix C, i.e. the CU, AR, and AT values of 4,860 design alternatives generated by the three GP models.

Step 2: Normalize the CU, AR, and AT values to allow a comparable scale for all criteria by

$$r_{ij} = C_{ij} / \sqrt{\sum_{i=1}^{4860} C_{ij}} \ , \ i = 1, 2, \ldots, 4860; j = 1, 2, 3 \tag{4}$$

Step 3: Calculate the weighted normalized decision matrix. The weighted normalized value of v_{ij} can be calculated by

$$v_{ij} = w_j r_{ij} \ , \ \sum_{j=1}^{3} w_j = 1 \tag{5}$$

where w_j is the normalized weight of the jth attribute. As an illustration, we can assign the value of 1, 3, and 5 for the "slightly", "moderately", and "extremely", respectively. Hence, the normalized weights of the "moderately cute", "moderately artistic", and "extremely attractive" are 3/(3+3+5), 3/(3+3+5), and 5/(3+3+5), respectively.

Step 4: Determine the positive and negative ideal alternatives. The positive ideal alternative is a hypothetical alternative in which all attribute values correspond to the best level. On the contrary, the negative ideal solution is also a hypothetical alternative in which all attribute values correspond to the worst level. Denote the positive ideal alternative, A^+, and the negative ideal alternative, A^-, as

$$A^+=\{v_1^+, v_2^+, v_3^+\} \text{ and } A^-=\{v_1^-, v_2^-, v_3^-\} \quad (6)$$

In this illustration, we obtain $A^+=$ (1.09004, 0.49541, 1.23770), and $A^-=$ (0.00002, 0.07729, 0.00893) in all 4,860 design alternatives, respectively.

Step 5: Calculate the separation measures. The separation (distance) between alternatives can be measured by the n-dimensional Euclidean distance. The positive ideal alternative, S_i^+, is given as

$$S_i^+ = \sqrt{\sum_{j=1}^{3}(v_{ij} - v_j^+)^2} \quad (7)$$

Similarly, the negative ideal alternative, S_i^-, is given as

$$S_i^- = \sqrt{\sum_{j=1}^{3}(v_{ij} - v_j^-)^2} \quad (8)$$

Step 6: Obtain an overall preference value P_i for each design alternative C_i, relative to other alternatives, by

$$P_i = S_i^- / (S_i^+ + S_i^-) \quad (9)$$

The larger the preference value, the more preferred the alternatives.

Step 7: Rank 4,860 design alternatives by their P_i value to best match the desirable consumers' preferences. As an illustration, Table 3 shows the top 20 ranking design alternatives with "moderately cute", "moderately artistic", and "extremely attractive".

Table 4 shows the corresponding combination of form elements of the top 20 ranking design alternatives. To facilitate the product form design in the new characteristic toy development process, the product designer can use a computer aided design system. Table 5 shows the optimal combinations of form elements for the "moderately cute", "moderately artistic", and "extremely attractive". Product designers can follow the optimal combinations to best meet

specific consumers' preferences for a new characteristic toy design.

Table 3 The TOP 20 ranking design alternatives

Ranking	No.	S_i^+	S_i^-	P_i
1	3283	0.1104	1.6431	0.9370
2	3463	0.1748	1.5704	0.8999
3	3280	0.1862	1.5390	0.8921
4	3284	0.2250	1.5105	0.8704
5	3274	0.2531	1.4752	0.8536
6	3328	0.2510	1.4594	0.8532
7	3643	0.2867	1.5120	0.8406
8	3460	0.2911	1.4785	0.8355
9	3277	0.3008	1.4459	0.8278
10	3454	0.3063	1.4002	0.8205
11	3464	0.3316	1.4544	0.8143
12	3508	0.3221	1.3900	0.8119
13	3281	0.3403	1.4213	0.8068
14	3271	0.3316	1.3718	0.8054
15	3325	0.3443	1.3606	0.7980
16	3275	0.3623	1.3432	0.7876
17	3285	0.3792	1.3980	0.7866
18	3640	0.4013	1.4318	0.7811
19	3329	0.3778	1.3336	0.7792
20	3457	0.4065	1.3974	0.7746

Table 4 The corresponding combinations of form elements of the TOP 20 alternatives

Ranking	No.	Form Elements						
		X_1	X_2	X_3	X_4	X_5	X_6	X_7
1	3283	3	1	1	1	5	3	1
2	3463	3	1	2	1	5	3	1
3	3280	3	1	1	1	5	1	3
4	3284	3	1	1	1	5	3	1
5	3274	3	1	1	1	4	3	1
6	3328	3	1	1	2	5	3	1
7	3643	3	1	3	1	5	3	1
8	3460	3	1	2	1	5	2	1
9	3277	3	1	1	1	5	1	1
10	3454	3	1	2	1	4	3	1
11	3464	3	1	2	1	5	3	2
12	3508	3	1	2	2	5	3	1
13	3281	3	1	1	1	5	2	1
14	3271	3	1	1	1	4	2	1
15	3325	3	1	1	2	5	2	1
16	3275	3	1	1	1	4	3	2
17	3285	3	1	1	1	5	3	3
18	3640	3	1	3	1	5	2	1
19	3329	3	1	1	2	5	3	2
20	3457	3	1	2	1	5	1	1

Table 5 The optimal combination of form elements

Form Element	Optimal Combination		
	1st choice	2nd choice	3rd choice
Length ratio of head and body (X_1)	$<1:2$		
Width ratio of head and body (X_2)	head$>$body		
Costume style (X_3)	one-piece	two-pieces	robe
Costume pattern (X_4)	simple	striped	
Headdress (X_5)	arc-shaped	feathered	
Appearance of facial features (X_6)	entire features	partial features	eyes only
Overall appearance (X_7)	cute style	semi-personified style	personified style

5   Conclusions

In this paper, we have strictly followed the primary procedure of kansei engineering in the context of characteristic toys, including extracting the representative experimental sample, conducting the morphological analysis, and assessing consumers' preferences. In addition, we have built a GP-based TOPSIS decision support model for assisting product designers to obtain the optimal alternatives (ideal solutions) that can best meet consumers' preferences. The result has showed the GP-based TOPSIS decision support model can support the new product development process. This approach presented can be applied to other consumer products with a wide variety of product form elements and design factors.

6   Acknowledgements

7   References

[1]   J.-L. Deng, 1982. Control problems of grey system. System and Control Letters, vol. 1, 288-294.

[2]   J.-L. Deng, 1989. Introduction to grey system theory. The Journal of Grey System, vol. 1, 1-7.

[3]   C. Jonathan, and M. V. Craig, 2002. Creating breakthrough products- Innovation from product planning to program approval. Prentice Hall, New Jersey, 1-31.

[4]   H.-H. Lai, Y.-C. Lin, and C.-H. Yeh, 2005. Form design of product image using grey relational analysis and neural network models. Computers and Operations Research, vol. 32, 2689-2711.

[5]   H.-H. Lai, Y.-C. Lin, C.-H. Yeh, and C.-H. Wei, 2006. User oriented design for the optimal combination on product design. International Journal of Production Economics, vol. 100, 253-267.

[6]   Y.-C. Lin, H.-H. Lai, and C.-H. Yeh, 2007. Consumer-oriented product form design based on fuzzy logic: A case study of mobile phones. International Journal of Industrial Ergonomics, vol. 37, 531-543.

[7]   Y.-C. Lin, and C.-C. Wei, 2011. How to transfer the innovation knowledge from craft art into product design- A case study of character toys. Smart Innovation, Systems and Technologies, vol. 9, 265-274.

[8]   Y.-C. Lin, and C.-C. Wei, 2011. An intelligent decision support model for product design. Smart Innovation, Systems and Technologies, vol. 10, 833-842.

[9]   Y.-C. Lin, and C.-C. Wei, 2012. How to best meet consumers' preferences for new product design: An application of art and design. IEEE Computer Society, 610-612.

[10] Y.-C. Lin, C.-H. Yeh, C.-C. Wang, and C.-C. Wei, 2012. Is the linear modeling technique good enough for optimal form design? A comparison of quantitative analysis models. The Scientific World Journal 2012, Article ID 689842, 13 pages. DOI= 10.1100/2012/689842.

[11] Y.-C. Lin, 2013. Applying grey prediction approach to product form design based on multiple affective responses. ACM ICPS, 359-363.

[12] Y.-C. Lin, C.-C. Chen, and C.-H. Yeh, 2013. Intelligent decision support for new product development: A consumer-oriented approach. Applied Mathematics and Information Sciences, No. 7-2S, 375-382.

[13] Y.-C. Lin, C.-H. Yeh, and C.-C. Wei, 2013. How will the use of graphics affect visual aesthetics? A user-centered approach for web page design. International Journal of Human-Computer Studies, vol. 71, 217-227.

[14] M. Nagamachi, 1995. Kansei engineering: A new ergonomics consumer-oriented technology for product development. International Journal of Industrial Ergonomics, vol. 15, 3-10.

[15] M. Nagamachi, 2002. Kansei engineering as a powerful consumer-oriented technology for product development. Applied Ergonomics, vol. 33, 289-294.

[16] G. H. Walker, N. A. Stanton, D. P. Jenkins, and P. M. Salmon, 2009. From telephones to iPhones: Applying systems thinking to networked, interoperable products. Applied Ergonomics, vol. 40, 206-215.

[17] M. Yamamoto, and D. R. Lambert, 1994. The impact of product aesthetics on the evaluation of industrial products. Journal of Product Innovation Management, vol. 11, 309-324.

[18] C.-H. Yeh, and Y.-H. Chang, 2009. Modeling subjective evaluation for fuzzy group multicriteria decision making. European Journal of Operational Research, vol. 194, 464-473.

Expert System Solution for Fault Detection Purposes in Power Distribution Systems

N.R. Santos[1], D.H. Spatti[2], L. Zamboni[1], I.N. da Silva[2], L.N. Soares[3], and R.A. Flauzino[2]

[1]EDP – Bandeirante, São Paulo, SP, Brazil
[2]University of São Paulo, USP/EESC/SEL, São Carlos, SP, Brazil
[3]EDP – ESCELSA, Vitória, ES, Brazil

Abstract—*The main purpose of this paper is to present architecture of automated system that allows monitoring and tracking in real time (online) the possible occurrence of faults and electromagnetic transients observed in primary power distribution networks. Through the interconnection of this automated system to the utility operation center, it will be possible to provide an efficient tool that will assist in decision-making by the Operation Center. In short, the desired purpose aims to have all tools necessary to identify, almost instantaneously, the occurrence of faults and transient disturbances in the primary power distribution system, as well as to determine its origin and probable location. The compilations of results from the application of this automated system show that the developed techniques provide accurate results, identifying and locating a batch of several occurrences of faults observed in the distribution system.*

Keywords: *fault detection, fault location, intelligent systems.*

1 Introduction

The detection, classification and location of faults in power systems are the target of transmission and distribution utilities. Thus, it is observed by the related literature, propositions of works that make use of a diversity of tools in a variety of conjunctions.

According to [1,2] the major limitation of the some approaches is the need to make measurement in at least two points of the system. It can be verified that this kind of method is more suitable for transmission systems than distribution systems.

Another successful combination about fault identification is to use the set of measurement and its decompositions into symmetrical components, because that the faulted section can be determined through the temporal dynamic of impedance matrix [3].

Simulations are indispensable for this purpose in order to validate any approach developed. So, case studies are conducted and the results provided by the methodology are compared with those who actually denote the reality of simulated faults.

Despite the correct identification of faults in most cases, the authors of [3] point out that the efficiency some techniques is conditioned to the accuracy of distribution lines impedance. Hence, the simple and constant changes in load and ambient temperature would be able to compromise the efficiency of this approach.

Computational-based models can create great amount of data of impedance and the limitations of the feeder modeling should be decreased, as in [4] where authors use calculated impedances obtained by means of the power system topology and by voltage and current measurements. A computational model of the distribution line is used to create faults conditions and these models are constantly updated by measurements realized on the substation secondary. However, errors can up to 25% if the system is unbalanced and no smart system is used to compensate this condition.

To suppress that limitations intelligent systems emerge as a new proposal to the tractability of problems whose solutions are inherently complex, such as identification a location of faults on power distribution systems. Besides the non-linearity of the problem, unbalanced load and huge types of faults, good results have been achieved using combination of tools.

An example of this new trend is the paper presented in [5], where techniques related to genetic algorithms and sparse areas are combined to form a system capable of locating faults in power systems.

In [6] a approach was developed to classify faults in electric power systems via multi-level analysis provided by the wavelet decomposition of the waveforms of voltage and current. This type of analysis is known for providing information on disturbances in electric power systems and has broad applicability in the power quality context. However, analysis of results from this tool for signal processing does not proceed naturally and much experience is necessary so that the findings are feasible to the actual facts.

On the other hand, fuzzy inference systems have the goal of emulating the approximate shape of human reasoning. In this way, the works [6,7] presents a method for fault classification in power distribution systems capable to distinguish real faults of programmed tasks into the system, for example, the energization of transformers.

In the study reported in [8] authors conduct their research using a vector decomposition of voltage and current waveforms and artificial neural networks for determining the location of fault occurrence in the system.

However, despite the use of intelligent systems, evidential results may not possess all desired requirements, or, these results could be improved. With such features as a premise, it

is increasingly common to develop systems for detection, classification and location of fault that do not use only one expert system, but several of these systems arranged in an orderly manner [9].

In parallel, integration and communication systems directed to the operation of an electric power system is a worldwide tonic since the mid-70s, however, that over those 30 years of development of systems to control and monitor, many concepts were created, as well as new alternatives to reduce costs with such systems were implemented, as outlined in [10,11].

Following the trends for a modern system of supervision and control of power distribution systems, this paper presents a complete solution for faults identification and location, working as a decision making tool that helps operators.

This approach is called solution because uses a great number of methods, algorithm's and techniques combined with hardware equipment to produce a very accurate fault identification and locations. Expert systems are the kernel of this purpose solution, due to handle a great variety of faults types, non-linearity's, signal contaminations and external events. In Table 1 is shown a little statistical history about interruption events registered in the case study feeder of this paper.

Table 1 – Interruption statistics for the feeder used in the study.

Parameter	Value
Processed events	1384
Interruptions events found	363
Cases involving switch breaker (%)	26.83
Cases involving reclosers (%)	73.17
3 phase faults (%)	26.82
1 phase faults (%)	7.31
Phase-phase faults (%)	14.63
Other types of faults or not identified (%)	51.24
Occurrence with the highest incidence (%) *Branch Tree touching line*	29.26
Occurrence involving climatic conditions (%) *Atmospheric Discharge + Strong Wind*	9.75
Weather conditions with the highest incidence (%) *Good weather*	24.39

It can be seen from Table 1 that a huge number of faults type is unknown. In other hand, in most of cases involving interruptions the weather conditions were considered good and same analysis could also extended from atmospheric discharges impacts on the distribution network.

These statistics data must be used to adjust and refine feeder simulations models, because it is intrinsically fulfill of non-linearity and imprecision information.

The case study feeder can be seen in Fig. 1, with the highlighted substation at the begging. This feeder possesses 1255 primary branches.

In the follow sections will be presented the solution approach proposed on this paper, to solve the fault identification and location problem, using the feeder of Fig. 1 as case study.
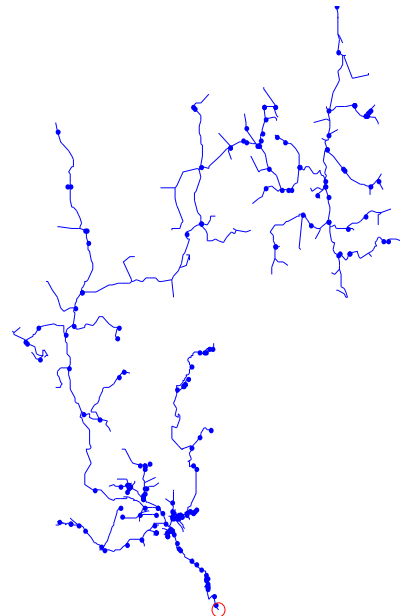
Fig. 1. Case study feeder.

2 Automated system for identification and location of faults

The development of any automated system can be divided into several objectives. Thus, by means of Fig. 2, there is a schematic illustration of the modularization of the proposed identification and fault location system. One can see that the system consists of the data acquisition module, pre-processing module, transients' identification module, identification system for phases participating of the fault, fault classifier system and fault locator system. Briefly, this system operates by using data acquired in the distribution system substation. In principle, these data consists of three phase voltages and the three line currents. Considering that Current Transform (CT) is not commonly used to measure ground current, this value is calculated resulting in a residual current of CTs allocated in phase.

The data acquisition module has the functionality to adequate levels of voltage and current of the distribution system so that they can be digitalized and acquired. These digitized data are, in turn, processed and its main parameters are determined by the pre-processing module. By using these parameters, the transient's identification module detects when a disturbance starts, and because of its identification, this module classifies it as arising from load changes or fault condition. The identification of a fault condition, in turn, triggers the operation of the identification system of the phases participating in the fault. Finally, with the data of the phases participating in the fault, the fault classifier system informs what kind of fault has been identified, while the fault locator system indicates the location where it occurred. The main feature of the pre-processing module is to group or compress the waveform data so that the excessive amount of data coming from these signals can be represented by a smaller number of parameters. The transient's identification module

identifies the existence of transients of current and voltage in distribution systems, regardless of its origin. This result will follow up the identification process of participating phases and their classification. More specifically, the schematic representation shown in Fig. 2 can be complemented by the block diagram shown in Fig. 3, in which the steps for the identification of faults, discrimination of participating phases and estimation of distance and resistance of fault can be observed.
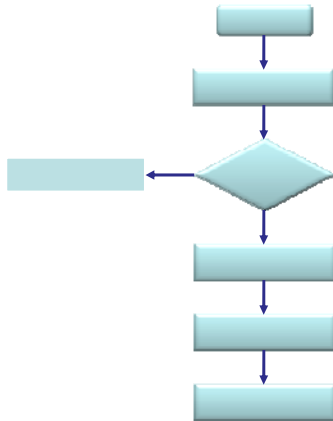
Fig. 2. Diagram representing the identification and location of faults.
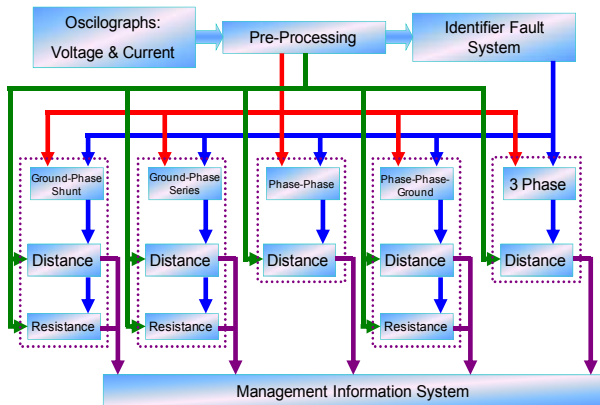
Fig. 3. Diagram of the fault identifier.

Through this schematic diagram it is possible to see that, by the identification of a fault, there is the consequent determination of the type of this fault. The identification of the fault type is very important, since this characteristic determines the way in which the disturbance data will be processed in order to determine the participating phases of the fault, so as to proceed with the estimation of the distance of incidence and fault resistance.

The operation of the Fault Classifier system assigns to each kind of fault a probability ratio which, if close to the unity, indicates a strong tendency of the disturbance to be associated with this kind of fault.

If, on the other hand, this rate is nearing zero, the opposite applies. Thus, the fault classification module indicates the kind of fault based on the highest calculated probability, in which FFF label refers to three-phase faults, DLF to the phase-phase faults, FFT to the phase-to-phase-earth faults, FFP to

parallel phase-earth faults and FFS to phase-series earth faults.

Once the kind of fault is identified, the participating phases of the fault must be identified.

The identification of type of fault as well as the discrimination process of the participating phases of the fault used data coming from the technique of decomposition in orthogonal components [12,13]. The number of components used in each of these tasks is five. Moreover, receiving such components as inputs, there is artificial neural networks of multilayer perceptron architecture, which were trained by compound samples sets for each kind of fault, for a total of 18 thousand cases of faults computationally simulated. Finally, the Fault location system can be shown in Fig. 4.
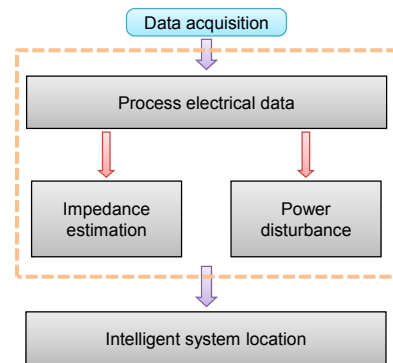
Fig. 4. Diagram of the fault location module.

One of the most important points related to the faulty section selection for any location method is the choice of the candidate branch, which most represents the electrical characteristics observed during the fault. Methods that analyze purely the impedance end up across multiple sections choice to the same universe search values.

The analysis proposed in this paper consists in investigating the maps for resistance R0 and R1, and maps of X0 and X1 on the distance. The Impedance estimation process uses a great amount of simulation data involving the case study feeder, which database is composed using all feeder models. The map for reactance distribution along the feeder could be seen in Fig. 5. In Fig. 6 is shown the map for resistance distribution along the feeder.
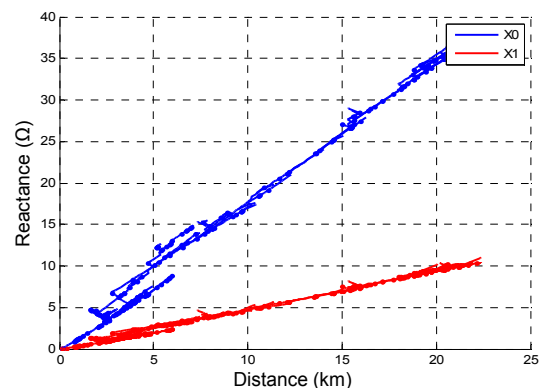
Fig. 5. Map from reactance distribution along the feeder.
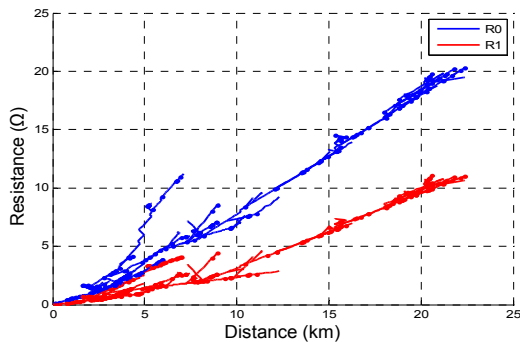
Fig. 6. Map from resistance distribution along the feeder.

The disturbance occurred during a fault could be used as a filter for more accurate faulty section determination. By observing oscillography records was possible create a power disturbance analysis in time, such as shown in Fig. 7.

It is observed from this figure that the power from phase green is considerable high at 0.04 seconds in opposition the normal operation for other phases (blue and red). This disturbance can also help for identify candidate branches with same electrical response when experiment similar fault conditions.

For all process described on this section it is import mention the relevance of the feeder real data simulation, responsible for create a high fidelity electrical condition database from this circuit.
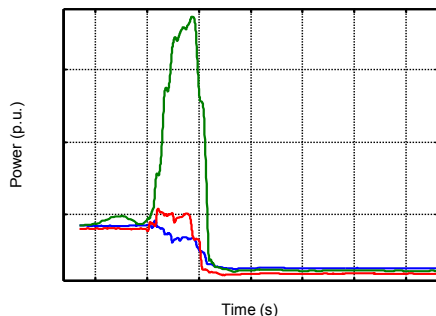
Fig. 7. Power disturbance from a fault example.

3 Supervisory of the data acquisition system

The acquisition of voltage and current signals related to an electric power system, as well as a power distribution system, use not only specific hardware but also software that are also able to identify the phenomena to be monitored. Thus, the need to identify disturbances arising from a possible fault condition in the monitored feeder becomes the first requirement for the software controlling the hardware. Besides being able to identify the occurrence of disturbances, the software must be provided with facilities for adequate storage of acquired signals, aiming the following processes:

- Identification of whether the oscillographed disturbances is caused by a fault condition;
- Classification of type of fault that occurred;
- Discrimination of participating phases of fault;
- Estimation of distance of fault occurrence;

- Estimation of fault resistance in order to punctuate possible triggers of the same events;
- Election and consideration of possible extensions where the fault was found.

Thus, the software made to monitor signals acquired by the hardware was implemented on LabView platform, which is known for performing specific tasks such as those relevant to the project in question. This implementation contemplates, in a unified way, both the identification of disturbances as well as the supervision of recording of the oscillographies which were made.

The identification of the disturbances is performed based on instantaneous values of currents and on instantaneous values of zero sequence of current and voltage. Therefore, if those signs exceed the predetermined threshold, the software identifies this as a disturbance, thus, resulting in the recording of the acquired signals.

In this first version of the system, an acquisition rate of 96 samples per cycle, i.e., a total of 5,760 samples per second is being used. But, if necessary, the developed system can reach up to 30,720 samples per second, equivalent to 512 samples per cycle. Still, the system considers when the acquired signals are stored, a time of 5 cycles for pre-disturbance, by recording a total time which takes a second.

Fig. 8 is presented with the purpose of illustrating some of man-machine interface of the developed supervisory. Subsequently, by means of Fig. 9, the scanning module of records made is also pictured.
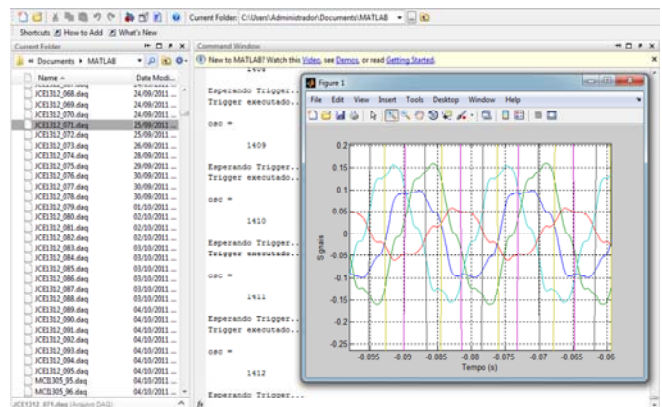
Fig. 8. Interface of supervisory system.
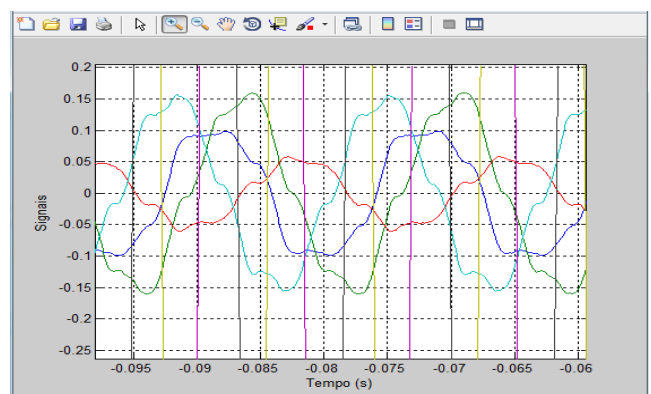
Fig 9. Interface for visualization of oscillographed events.

4 Results related to the system of identification and location of faults

The computer simulations sets used in this research were each composed by a population of 18,000 simulations. Figure 8 shows the histogram of percentage distribution of simulations depending on distance.

The histogram in Fig. 10, besides showing the percentage distribution of the number of simulations based on distance, also allows inferring that the number of bars at a distance equal to or less than one kilometer is the highest one. Moreover, the concentration of bars at a distance between 5 and 10 km is considered more important for this specific feeder.
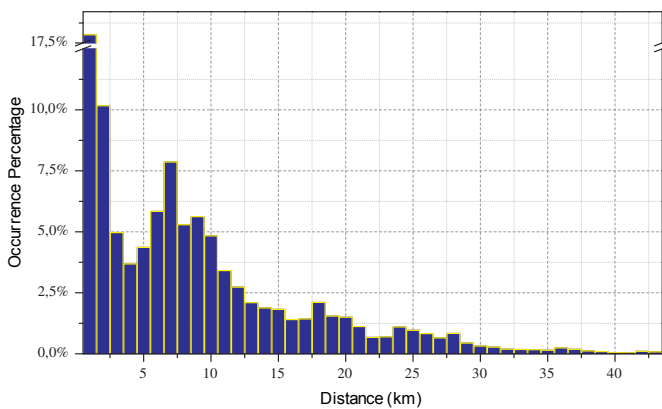
Fig. 10. Histogram of percentage distribution of the quantity of simulations in function of distance.

This section emphasized the importance of having a discrimination of the participating phase of fault as an additional process. This importance can be justified by the graph shown in Fig. 11. In this figure we have the histogram of error to estimate the distance of fault occurrence, disregarding, for such purposes, the phases involved.
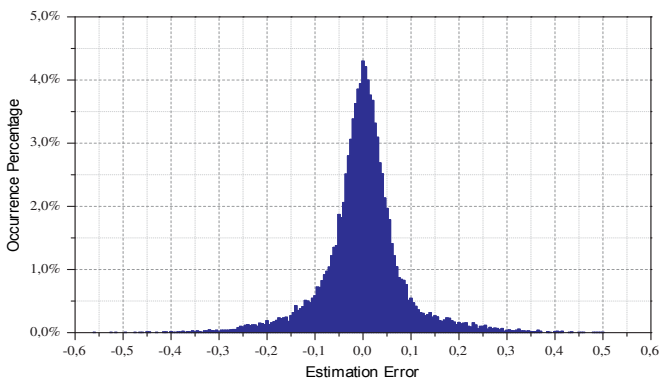
Fig. 11. Histogram of error for distance estimation without considers the participant phases.

Through the graph in Fig. 11, one can then verify that the distance estimation error, not considering the participating phase of the fault, has (in these conditions) concentration

within the range of -10% to 10%.

If the participating phase information of the fault is considered, the error behavior changes and appears more suitable to the purposes of the project. This aspect is illustrated in Fig. 12, which highlights the error behavior considering the faults where Phase A participated. It is noted here that the error in the estimation of distance is much more accurate than the cases of Fig. 10.
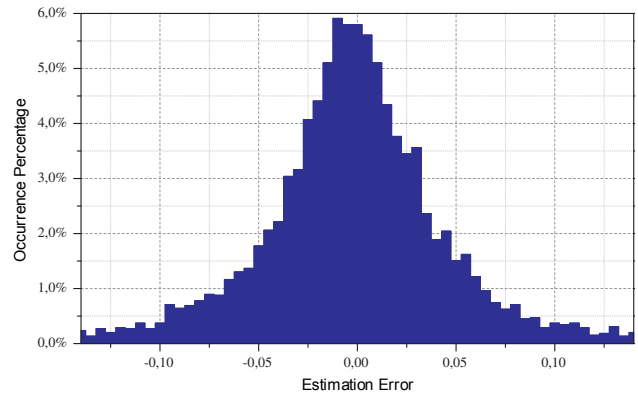
Fig. 12. Histogram of error for distance estimation considering faults involving phase A.

For all other phases, the results produced by the automated system for identification and location of fault were also similar to those presented in Fig. 12.

Real location fault condition can be seen in Fig. 13. An artificial condition of fault was create do validate the approach, at substation. The system was adjusted to generate 10 possible solutions.

Due to the proximity of the "occurrence" of the fault the accuracy of the method can be observed.
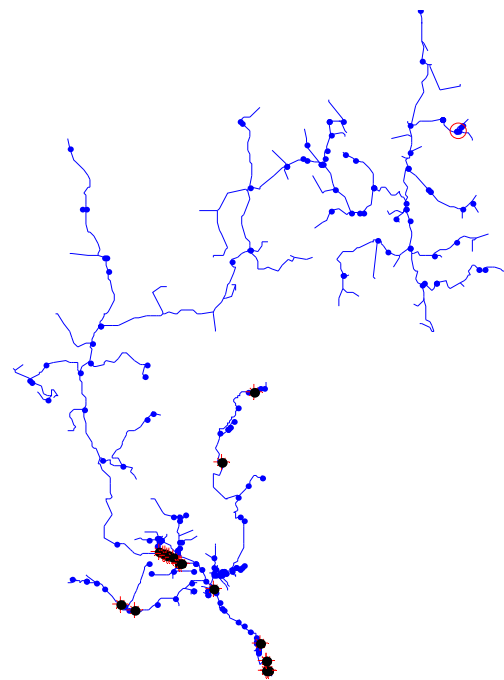
Fig. 13. Fault location with real candidates branches.

5 Conclusions

This paper presented a scheme based on the use of intelligent systems for efficient identification and location of faults in distribution systems, taking into account the voltages of phase and line currents observed at the substation.

The proposed architecture was designed in a modular way in order to provide a higher degree of redundancy procedures for identifying and locating faults. The sequencing of tasks of transient identification with the discrimination of fault and, finally, with the task of identifying the participating phases of fault brings a greater robustness to the system as a whole.

In addition to a modular and robust architecture from the faults occurrence point of view, the intelligent system for identifying faults was implemented by using tools, proven to be effective, dedicated to the achievement of pre-processing the voltage signals and current sampled at the substation. These tools operate on these signals in order to extract features that individualize and distinguish fault conditions from other normal operating conditions.

Thus, the efficiency of these pre-processing tools, which are based on decomposition technique for orthogonal components, give the system a robustness and accuracy with regards to the identification of phase-to-earth faults, as well as for determining the participating phase. The answers helped confirm how correlated the orthogonal components can be to the execution of both tasks.

The final results derived from the implementation of the proposed system were fully satisfactory, and they were tested and validated by using data from both simulations and fault oscillography acquired at the substation.

6 Acknowledgements

7 References

[1] F. Chunju, C. Huarong and Y. Weiyong, "Application of six-sequence fault components in fault location for joint parallel transmission line", *Tsinghu Science and Technology*, vol. 10, no. 2, pp. 247-253, 2005.

[2] G. Yaozhong, *New Types of Protective Relaying and Fault Location - Theory and Technique*, Xi'an Jiao Tong University Press, 1996.

[3] W. N. Mahmood, R. N. Mukerjee and V. K. Ramachandaramurthy, "Fault point identification in a power network using single-point measurement", *IEEE TENCON Conference*, pp. 381-384, 2004.

[4] M. S. Choi, S. J. Lee and D. S. Lee, "A new fault location algorithm using direct circuit analysis for distribution systems", *IEEE Transactions on Power Delivery*, vol. 19, no. 1, pp. 35-41, 2004.

[5] S. Luo, M. Kezunovic and D. R. Sevick, "Locating faults in the transmission network using field measurements, simulation data and genetic algorithm", *Electric Power Systems Research*, vol. 71, pp. 169-177, 2004.

[6] O. A. S. Youssef, "Combined fuzzy-logic wavelet-based fault classification technique for power system relaying", *IEEE Transactions on Power Delivery*, vol. 19, no. 2, pp. 582-589, 2004.

[7] O. A. S. Youssef, "Applications of fuzzy-logic-wavelet-based techniques for transformers inrush currents identification and power system faults classification", *IEEE PES Power Systems Conference and Exposition*, pp. 553-559, 2004.

[8] L. S. Martins, J. F. Martins, V. Fernão Pires and C. M. Alegria, "A neural space vector fault location for parallel doublé-circuit distribution lines", *Electrical Power & Energy Systems*, vol. 27, 225-231, 2005.

[9] R. N. Mahanty and P. B. D. Gupta, "Application of RBF neural network to fault classification and location in transmission lines", *IEE Proceedings on Generation, Transmission and Distribution*, vol. 151, no. 2, pp. 201-212, 2004.

[10] J. Zaborzsky, K. Whang and K. Prasad, "Fast contingency evaluation using concentric relaxation", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-99, no. 1, pp. 28-36, 1980.

[11] K. Tomsovic, D. E. Bakken, V. Venkatasubramanian and A. Bose, "Designing the next generation of real-time control, communication, and computations for large power systems", *Proceedings of the IEEE*, vol. 93, no. 5, pp. 965-979, 2005.

[12] R. A. Flauzino, I. N. Silva, "Uma abordagem baseada na técnica de decomposição em componentes ortogonais e sistemas de inferência fuzzy para localização de faltas de alta-impedância em sistemas de distribuição de energia elétrica", *Anais do VIII Simpósio Brasileiro de Automação Inteligente*, pp. 1-6, 2007.

[13] R. A. Flauzino, *Identificação e Localização de Faltas de Alta Impedância em Sistemas de Distribuição Baseadas em Decomposição por Componentes Ortogonais e Inferência Fuzzy*, Tese de Doutorado, Universidade de São (USP), Escola de Engenharia de São Carlos, 2007.

Telecommunication Error Resilient Reinforcement Learning and Adaptive Control Algorithm

N. Taylor[1], B. Yang[2]
[1]Computer Science Department, Bowie State University, Bowie, MD, USA
[2]Computer Science Department, Bowie State University, Bowie, MD, USA

Abstract— *In this paper, a new telecommunication error resilient reinforcement learning and adaptive control algorithm is proposed which is implemented by an intelligent agent and knowledge discovery engine .Error Resilience is a crucial component to a reliable, healthy system life cycle and it determines how high the system rise above the failures or errors that threatens to reduce its performance or shut it down. The algorithm view is characterized by accurate and flexible system thinking, and the capacity to see other node failure points and to reconfigure or redirect traffic in another node success point, and the ability to move on with daily operations despite system errors and failures. Most importantly, the algorithm learns the system environment over time and adapts to errors and failures but not just an ability to resolve one error or failure at a time in order for the system to survive.*

Keywords— **algorithm, error, resilience, communication, networks**

I. INTRODUCTION

Connecting two or more nodes with a suitable channel is the first step in telecommunication. Before the nodes can successfully exchange messages, the bits must be encoded so that they can be understood by the receiving entity [13]. This is followed by delineating the sequence of bits transmitted over the link into complete messages that can be delivered to the end node. These messages are sometimes corrupted during transmission, making it necessary to detect and correct errors to make the link appear reliable, in spite of the fact that message corruption may occur occasionally [14]. In a multichannel shared environment, it is also necessary to mediate access to the channel to enhance performance by easing congestion. With competition and a high demand for real-time processing of service activation commands, overcoming provisional limitations, enhancement of broader network coverage and the demand for high bandwidth within multichannel telecommunication networks is challenging [25]. Though proven to be a challenge, many enterprises are relying heavily on service automation to keep their competitive advantage [18]. In order to follow up rapidly changing environment, Telecommunication network service providers (TNSP) have to develop the flexible and cost effective operations support systems (OSS).

II. LITERATURE REVIEW

This section highlights related work and groups them into two types of literature review on telecommunication systems network. The first group concentrates on telecommunication networks services and the second focuses on error resilience algorithm. With development of distributed networks, network management has become an annoyance for administration staff and also poses the challenge of managing each device separately, even if they are of the same vendor. Moreover, configuration overhead, as well as misconfiguration possibility is quite high in telecommunication environment. The provisioning process for a basic service required complex configurations of numerous network devices and end user optional service request [28]. It is also hard to store all network devices data and connections information in a plain list hence network structuring loading approach was a natural solution. According to the United States Department of Labor Bureau of Labor Statistics, the telecommunications industry is divided into four main sectors: wired, wireless, satellite, and other telecommunications establishments [10]. The largest sector of the telecommunication industry continues to be made up of wired telecommunication carriers. Establishments in this sector mainly provide telecommunication services such as such as wired (landline) telephone, digital subscriber line (DSL) Internet, and cable TV and Internet services. These organizations route TV, voice, Internet, data, and other content over a network of wires, wireless and cables, and control access to this content. They may own and maintain networks, share networks with other organizations, or lease network capacity from other companies [19]. Telecommunication network by definition is a system of interconnected elements linked by facilities (i.e., physical connections) over which traffic will flow. The traffic may be conversations, information, or complex video or audio services. The telecommunication network must also be able to control the interconnected elements [4]. In modern telecommunication networks, whether an analogy, digital or hybrid network, there is one main aim of the network, which is the transmission of information over a

significant distance between two end points to communicate. This requires transmitters, receivers, and communication channels. Martin describes ways to hasten the advent of the so-called information age based on a service-rich ubiquitous telecommunication network handling integrated voice, data, and video [3]. The research also noted that telecommunications services have historically been introduced slowly and heterogeneously. The author then addresses an approach that reverses this trend through a centralized service control intelligent network [8].

III. ALGORITHM

In the reinforcement learning frame work, the agent algorithm will be provided with reward and bonus function which will indicate to the agent when it's doing well, load balancing well or doing poorly [28]. The Re-Del agent will receive positive rewards for successfully provisioning and activating a service request and negative rewards for either failing to select a node with high success rate likelihood probability and not load balancing [22]. It will then be the learning algorithm's job to figure out how to choose nodes over time so as to obtain large rewards and bonuses. The use of reinforcement learning and adaptive control will begin with a definition of the Markov decision processes (MDP), which provides the formalism in which RL problem will be posed.

Notations and Problem Formulation
This section introduces some notations, definitions and basic theorems for Markov decision process and packet scheduling [108]. The version of the Error resilience problem to be addressed will be defined.

Markov Decision Processes and Packet Scheduling

A Markov Decision Process is a tuple $(\mathcal{S}, \mathcal{A}, \{\mathcal{P}_{sa}\}, \gamma, \mathcal{R})$. It provides a mathematical formalism in which RL problems is modeled in a sequential decision problem. Where:

- $\mathcal{S}$ is a fine set of $\mathcal{N}$ **state** space. (Network Status, $\mathcal{S}$ is the set of all possible network status attribute: Link Cost, Link Capacity, Response Time, Throughput Rate and Delivery Performance)

- $\mathcal{A}$ is a set of k **action** space. $\mathcal{A} = \{a_1, ..., a_k\}$ (The selection of a possible destination network element in which the service order can be completed successfully)

- $\mathcal{P}_{sa}(\cdot)$ is the likelihood success state of **transition provision and activation probability** function upon taking action a in state s. Given that the state transition probabilities are not known in this case, the MDP consists of a number of trials by the agent. $\mathcal{P}_{sa}$ is derived for the maximum likelihood success state

transition probability for each network element.

- $\gamma \in [0,1)$ is called the provision and activation **discount factor**.

- $\mathcal{R}: \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the provision and activation **reinforcement function** which depends on the action and outcome, bounded in absolute value by $\mathcal{R}_{max}$

The assumption here is that the reward is known, to simplify the notation. The rewards $\mathcal{R}(s)$ is written as a function of a state s only in case where $\mathcal{R}: \mathcal{S} \mapsto \mathbb{R}$. The goal of MDP is to find the best policy under the given optimality criteria by solving the Bellman's equation [113].
A **policy** is defined with respect to **provision and activation** as the mapping $\pi: \mathcal{S} \mapsto \mathcal{A}$ [107]. The agent must learn to choose actions over time so as to maximize the expected value of the total discounted rewards noting that the reward at time t is discounted by a factor of γ^t . The value function for a policy π, evaluated at any state s_1 is given by

$$V^\pi(s_1) = E[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \gamma^3 R(s_3) + ... | s_t = s, \pi] \qquad (1)$$

Where the expectation is over the distribution of the state sequence $(s_1, s_2, s_3 ......)$ which the agent passes through when executing the policy π starting from s_1. Also, **Q-function** is defined according to

$$Q^\pi(s, a) = R(s) + \gamma E_{s' \sim \mathcal{P}_{sa}(\cdot)} [V^\pi(s_1)] \qquad (2)$$

(where the notation $s' \sim \mathcal{P}_{sa}(\cdot)$ means the expectation is with respect to s' distribution according to $\mathcal{P}_{sa}(\cdot)$). The **optimal provision and activation value function** is

$$V^*(s) = max_\pi V^\pi(s) \qquad (3)$$

And **optimal provision and activation Q-function** is

$$Q^*(s, a) = max_\pi Q^\pi(s, a) \qquad (4)$$

For discrete, finite spaces, all these functions can be represented as vectors indexed by state, fixing some enumeration from 1 to $\mathcal{N}$ of the finite state space $\mathcal{S}$. The rewards may be written as an $\mathcal{N}$-dimensional vector $\mathcal{R}$, whose ith element is the rewards at the ith state of the MDP. V^π is also a vector whose ith element is the value function for the π evaluated at state i . For each action $\alpha = \pi(s)$, let $\mathcal{P}_{sa}$ be $\mathcal{N}$ by $\mathcal{N}$ matrix in which (i, j) gives the probability of transitioning to state j upon taking action in state i [110]. The goal of reinforcement learning and adaptive control is to find the policy π such that $V^\pi(s)$ is maximized. It has been shown that there exists at least one optimal policy π^* such that $V^\pi(s)$ is simultaneous maximized for $s \in \mathcal{S}$ by $\pi = \pi^*$. With provision

and activation discounted rewards, the utility of an infinite sequence is finite with the rewards bounded by R_{max} and $\gamma < 1$.
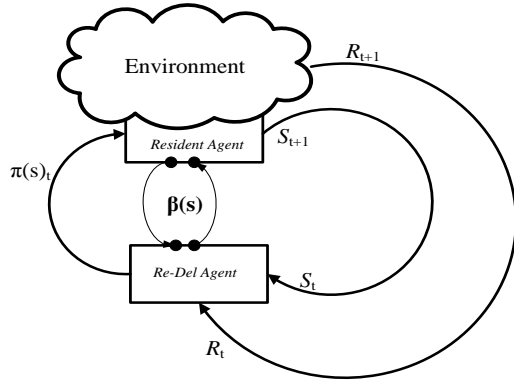
Fig. 1   Resident and Re-Del Agent-wired environment interaction in a RL&A model

$$P_h([S_0, S_1, S_{2,\dots}])$$
$$= \sum_{t=0}^{\infty} \gamma t R(s_t) \leq \sum_{t=0}^{\infty} \gamma t R_{max} = R_{max}/(1-\gamma) \qquad (5)$$

Included in the goal of the *Re-Del* Agent is to understand the urgency and sensitivity surrounding the error provisional and activation request. Understanding the structural limitations and utilizing the packet scheduling algorithm at the MAC layer and the routing algorithm by the agent at the network layer level are tightly coupled on the basis of the urgency metrics to effectively deliver delay-sensitive service request data over multi-rate carrier networks [10]. The objective is to deliver the maximum error delay-sensitive service request data within the end-to-end delay requirement and to load balance traffic across the entire network simultaneously. The urgency metric and network traffic metrics is defined before proposing the bonus function.

Definition of Packet Urgency: The packet urgency ($urg_{packet}(t)$) at the m_{th} node at time t is defined as

$$urg_{packet}(t) = (\quad 1 - \frac{dly_{residual}(t)}{dly_{max}}\quad)^2 \qquad (6)$$

$$dly_{residual}(t) = dly_{max} - dly^2_{cumulative}(t)$$

Where dly_{max} is the maximum tolerable end-to-end delay, $dly^2_{cumulative}(t)$ is the cumulative delay from the source node to m_{th} node along a route at time t, and $dly_{residual}(t)$ is the residual delay satisfying the end-to-end delay requirement over the remaining hops at time t. In general, $urg_{packet}(t)$ should be inversely proportional to $dly_{residual}(t)$. A packet with smaller $dly_{residual}(t)$ should be transmitted more urgently for one-time delivery to the destination node [13].

Definition of Node Urgency: The node urgency ($urg_{node}(t)$) at time t is defined as the sum of the packet urgency of all the packets in the buffer that is

$$urg_{node}(t) = \sum_{n=1}^{Npacket} urg_{packet(n)}(t) \qquad (7)$$

Where N_{packet} is the number of packets in the buffer and $urg_{packet}(t)$ is the packet urgency of the N_{th} packet in the buffer at time t. A larger amount of node urgency implies that more urgent packets are in the buffer [17].

Definition of Route Urgency: The route urgency ($urg_{route}(t)$) at time t is defined as the sum of the node urgency along route R that is

$$Urg_{route}(t) = \sum_{m \ni R} urg_{route(m)}(t) \qquad (9)$$

Where R denotes the route including all intermediate nodes and $Urg_{node}(t)$ is the node urgency of the m_{th} along the route at time t. As the route urgency increases, the route may become a congestion route [24].
A larger amount of node urgency implies that more urgent packets are in the buffer, using the standard formula for the sum of infinite geometrics series [6]. The state sequence depends on π that is executed if, whenever the agent is in state S, it take action $\alpha = \pi(s)$. So if s_t is the state the agent is in after executing π for t then by defining the provisioning function (equals the value function) for the policy π according to

$$P^\pi(s) = E[\ \sum_{t=0}^{\infty} \gamma t R(s_t)\ |\ \pi, S_{0=}S]$$

Basic Properties of MDPs and Networks Packet Scheduling
For the solution of the RLAC problem where the adaptive controller adjusts parameters of the network infrastructure in real time, two of the classical MDP's will be used. Also the controller optimize the path/cost/quality trade-off on the basis of specified marginal costs without sticking strictly to the set points originally defined during service provisioning [1],[2].

Theorem 1 (Bellman Equations) *let an MDP $M = (S, A, \{P_a\}, \gamma, R)$ and a policy $\pi: S \mapsto A$ be given. Then for all $s \in S$, $a \in A$, V^π and Q^π satisfy*

$$V^\pi(s) = R(s) + \gamma \sum P_{\pi(\cdot)}(s')\ V^\pi(s') \qquad (10)$$

$$Q^\pi(s,a) = R(s) + \gamma \sum P_a(s')\ V^\pi(s') \qquad (11)$$

Theorem 2 (Bellman Optimality) *let an MDP $M = (S, A, \{P_a\}, \gamma, R)$ and a policy $\pi: S \mapsto A$ be given. Then π is an optimal policy for M if for all $s \in S$, $a \in A$, V^π and Q^π satisfy*

$$\pi(s) \in arg\ max_{a \in A}\ Q^\pi(s,a) \qquad (12)$$

Theorem 3 *let the finite state space S, a set of actions $A = \{a_1, \dots a_k\}$, transition probability matrices, $\{P_a\}$, and a discount factor $\gamma \in [0,1)$ be given. Then the policy π given by $\pi(s) = a_i$ is optimal if and only if for all $a_1, \dots a_k$, the reward R satisfies*

$$(P_{ai} - P_a)(i - \gamma P_a)^{-1}\ R \geq 0 \qquad (13)$$

Proof: Since a policy π is executing whenever the agent is in a state S an action $\pi(s) = a_i$ is taken Equation (1)

$$V^\pi = R + \gamma P_a V^\pi$$

$$V^{\pi} = (i - \gamma \mathcal{R})^{-1} \mathcal{R} \tag{14}$$

Hence, $i - \gamma \mathcal{R}$ is always invertible. $\mathcal{R}$ being a transition matrix has all eigenvalues in the unit circle in the complex plan since $\gamma < 1$. This implies that the matrix $\gamma \mathcal{R}$ has all eigenvalues in the interior of the unit circle [66]. The eigenvalue equation for a matrix $\mathcal{A}$ is

$$\mathcal{A}v - \gamma v = 0,$$

Which is equivalent to:

$$(\mathcal{A} - \gamma i) = 0,$$

Where i is the $n \, x \, n$ identify matrix. It is a fundamental result of linear algebra that an equation $\mathcal{M}v = 0$ has a non-zero solution v if and only if the determinant $det(\mathcal{M})$ of the matrix $\mathcal{M}$ is zero. It follows that the eigenvalues of $\mathcal{A}$ are precisely the real numbers γ that satisfy the equation

$$det(\mathcal{A} - \gamma i) = 0 \tag{15}$$

The left-hand side of this Equation(15) can be seen (using Leibniz' rule for the determinant) to be a polynomial function of the variable γ [105]. The degree of this polynomial is n, the order of the matrix. Its coefficients depend on the entries of $\mathcal{A}$, except that its terms of n is always $(-1)^n \gamma^n$. This polynomial is also called the characteristic polynomial of $\mathcal{A}$; and Equation(15) is called the characteristic equation of $\mathcal{A}$. This means $i - \gamma \mathcal{R}$ has no zero eigenvalues and it's not singular. Substituting Equation (11) into (12) from Theorem 2, π^* is optimal if and only if

$$\pi^*(s) \in arg \, max_{a \in \mathcal{A}} \, \gamma \sum \mathcal{R}_a(s') \, V^{\pi}(s')$$
$$\forall \, s \in \mathcal{S}$$
$$\Leftrightarrow \sum \mathcal{R}_{al}(s') \, V^{\pi}(s') \qquad \geq \sum \mathcal{R}_a(s') \, V^{\pi}(s')$$
$$\forall \, s \in \mathcal{S}, a \in \mathcal{A}$$

Where $al \equiv \pi(s)$

$\pi^*(s)$ yields the action α that attains the maximum in the equation (12). Therefore for every state $\mathcal{S}$ and every policy π

$$V^*(s) = V^{\pi^*}(s) \geq V^{\pi}(s)$$

Substituting $P^*(s)$ the provisional function yields

$$P^*(s) = P^{\pi^*}(s) \geq P^{\pi}(s)$$

This means that policy π^* can be used no matter what the initial state of the MDP is. With known rewards and state transition probabilities the algorithm proceeds as follows

Network Status Attribute Computation
I.　　Link Status ($\supset$) is either active (1) or inactive (0)
II.　　Link Cost: The formula to calculate the cost is reference bandwidth divided by interface bandwidth.
　　For example, in the case of Ethernet, it is 100 Mbps / 10 Mbps = 10
　　LC = R/BW

Where,
　　LC = Link Cost
　　R = Reference bandwidth
　　 BW = Interface bandwidth

III.　　Link Capacity of a channel is given by Shannon's law. The equation of Shannon's law is as follows:
C = W Log2 (1 + (S/N))
Where,
C = the capcity of a channel is bits per second
　　　　W = bandwidth of a channel in Hz
　　　　 S/N = signal to noise ratio

IV.　　Response Time (RT):
RT = (M1 + … + Mn)/N
Where:
　　　　RT = average response time
　　　　N = number of iterations
　　　　M = measured time (end time - start time)
　　　　M1 = first iteration
　　　　Mn = nth iteration

V.　　Throughput Rate:
Throughput rate is defined as the amount of data sent and received divided by the measured transaction time.
　　T = (S + R)/m
　　Where:
　　　　T = throughput rate
　　　　S = bytes sent by node 1
　　　　R = bytes received by node 2
　　　　M = measured time (end time - start time)

VI.　　Delivery Performance:
　　Delivery performance is a measure of the amount of data lost when sending data from between two nodes
　　L = ((S - R)/S) * 100
　　Where:
　　　　L = percentage of lost data
　　　　S = total bytes sent by node 1
　　　　R = total bytes received by node 2

Hence using the network attributes to determine the link state

$$S = \supset (l_1 + l_2 + l_3 + l_4 + … + l_n)$$

Where l = network attribute
　　$\supset$ = active (1) or inactive (0)

$$S = \supset(\, 1/l_{LC} + l_C + 1/l_{RT} + l_T + l_L)$$

A.　System Architecture of the *Re-Del Agent*
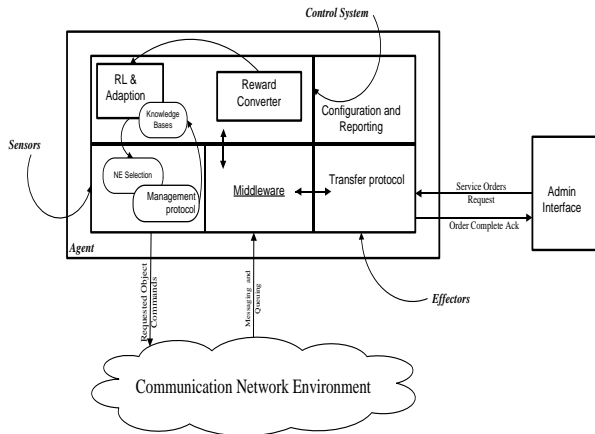
Fig. 2 General block diagram of an Agent in telecommunication network systems

$$P^*(s) = R(S) + \max_{a \in A} \gamma \sum_{S' \in S} P_{sa}(S') \, P^*(S')$$
$$+1/ \left(\sum_{e=0}^{n} \beta_e(s)\right)$$

The algorithm is updating as follows

1. Randomly sample m states $s^{(1)}, s^{(2)}, s^{(3)},\dots s^{(m)} \in \mathcal{QS}$.

2. Initialize $\mho := 0$

3. Initialize $\mathcal{V} := 0$

4. Initialize $\mathcal{R} := 0$

5. Repeat {
   For $i = 1,\dots,m$
   For each action $a \in \mathcal{A}$ {

   $$\text{Sample } s'_{(1)}, s'_{(2)}, s'_{(3)},\dots s'_k \sim \mathcal{R}_{s(i)a}$$

   $$\text{Set } q(a) = 1/k \sum_{j=1}^{k} \mathcal{R}(s^{(i)} + \gamma V(s'_j)$$

/** Hence $q(a)$ is an estimate of $\mathcal{R}(s) + \gamma E_{s' \sim \mathcal{R}_{sa}(.)}$

$[V(s')]$ **/
   }
   Set $y^{(i)} = \max_a q(a)$
   /** Hence $y^{(i)}$ is an estimate of $\mathcal{R}(s) + \gamma$

   $\max_a E_{s' \sim \mathcal{R}_{sa}(.)} [V(s')]$ **/
   }
/** The update of the value function is according

to $V(s^{(i)}) := y^{(i)}$ **/

/** In order to achieve $V(s^{(i)}) \approx y^{(i)}$ using supervised learning **/

   {

$$\text{Set} \quad \mathcal{R} := \arg \min_{R \in R_{all}} \left\{ \sum_{n \in R} \frac{\partial}{\partial} \mathcal{U}rg_{node(n)}(t) \right\}$$

$$\mathcal{V} := \arg \min_{1 \le i \le N_{packet}} \sum_{\substack{n=1 \\ n \ne i}}^{N_{packet}} \left\{ \frac{\Delta Urg_{packet(n)}(t)}{dly_{one-hop}^{\sim packet(n)}} \right\} -$$

$$\frac{Urg_{packet(i)}(t)}{dly_{one-hop}^{\sim packet(i)}}$$

/** Generate a detailed status validity of each address, its deliverable status and the nature of any ambiguities it contains. Errors are corrected and complete partial address records using information from the address **/

$$S^* := \arg \max_k p_d(z_k \mid w_j), \quad j = 1,2,3,\dots,Q$$

If Load Balance $<>$ Required

$$\mho := \arg \min_\mho \tfrac{1}{2} \sum_{j=1}^{k} (\mho^{\tau} \emptyset(s^{(i)}) - y^{(i)})^2 \}$$

   Else
/** In order to achieve load distribution to the node, balancing the load across the node as bonus function $\beta(s)) = 1\backslash(\sum_{e=0}^{n} \beta_e(s))$ to their current agent using supervised learning **/

Initial $\beta(s) := 0$, $\mathcal{P} := 0$

$$\mathcal{P} := \arg \min_\mho \tfrac{1}{2} \sum_{j=1}^{k} (\mho^{\tau} \emptyset(s^{(i)}) - y^{(i)})^2 + \beta(s)$$
   }

IV. EXPERIMENTAL RESULT

Figure 1 plots the normal variations between service provisioning and activation success and failure of the network elements at each demarcation point. Figure 2 plots normal variations of their success rate for total services requests without the proposed methods that are based on the Re-Del agent or any data entropy elimination. When the service requests are sent, the running configuration tries to accomplish the task and if an error is generated, the system logs and sends an alarm depending on the severity of the error. Within a normal operation, field engineers will later diagnose the problem and get it resolved at an unspecified time. This results in a service delay and affects the overall service delivery performance metrics as shown. As can be seen in Figure 2, the variations between the success and failure for node IAD, MIA is very marginal for normal operations. As seen in Figure 14 the

highest success rate is in the upper 80% for normal operation for node PDX and the lowest is the 50% for node DCA. Complete node failure ("00000 - Fail"), Telephone not in service("00013 - TNNotInService"),Bad command("00075 - BadCmd"), Account directory in error ("00027 - AccountIOErr"), service feature request option is not supported by the device("00067 - OptNotSupported"), Invalid line equipment number or line terminal identification ("00269 - InvalidLENorLTID") and object not found ("00291 - ObjectNotFound") are some of the errors that are generated during normal operations.
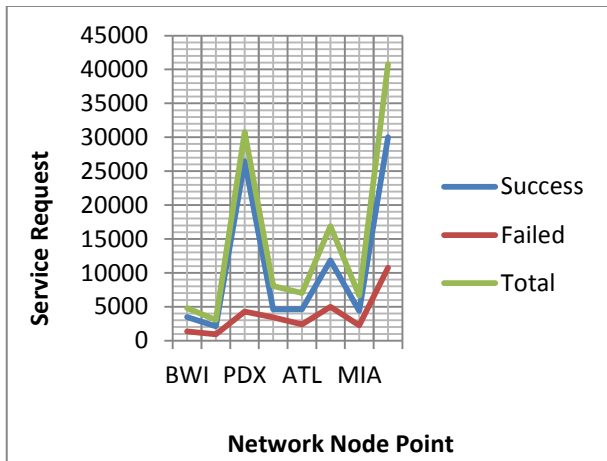
Figure 1 Node specific regular service success and failure distribution

It is very important to note that these kinds of errors can be redirected and provisioned through a different node route.
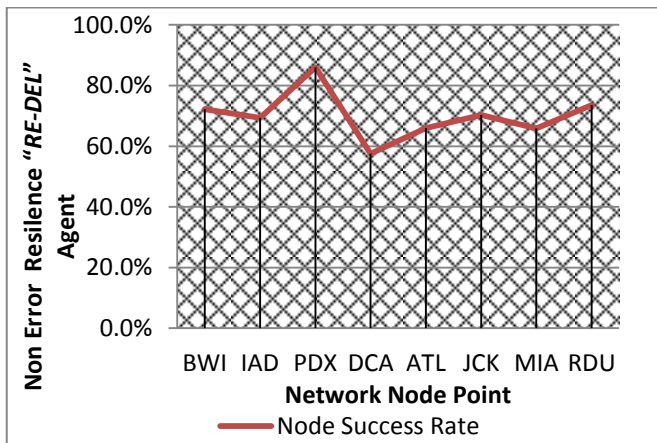
Figure 2 Node specific regular service success and failure distribution Success Rate

The bigger the values of variation between success and failure, the better the service that is delivered. Figure 3 shows the Re-Del agent relegation probability distribution and
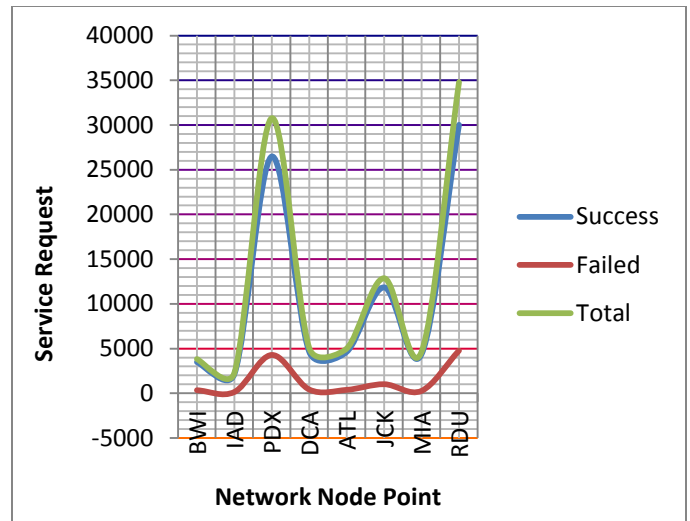
Figure 3 Agent Node specific regular service success and failure distribution

Figure 4 shows the success rate for each network node for the same service errors derived from the environment. In Table VI, sample environment errors redistributed with high success rates under Re-Del Agent are demonstrated, supporting the aforementioned observations.
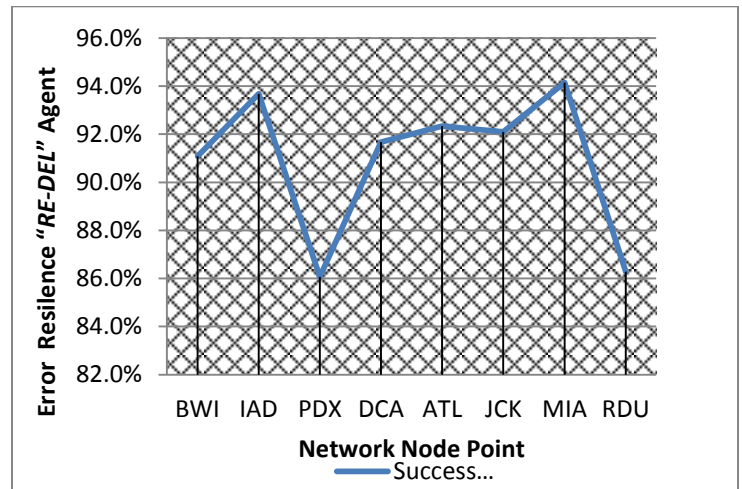
Figure 4 Agent Node specific regular service success and failure distribution success rate

V. Conclusions

The contribution of this paper is the use of reinforcement learning and adaptive control, to model this problem with MDP and to suggest a relatively Re-Del Agent error resilience solution to provide robustness to existing wired networks against a challenging and important defect in existing service order provisioning and activation systems. This approach makes sure of the network link state attributes, thereby taking advantage of most of the communication protocols. The inherent defect in Link State is packet delivery rate or throughput deteriorates as the level of service order request increases, and more seriously, when a network element collapses under severe disruptions. This approach employs the

routing protocol to update its routing table and recalculate the link cost in additional protocols, updating their statistics. It enables the agent to tolerate a level of disruption that Link State cannot. The shortcoming of this approach is that is operated in a fully connected network environment without a single point of failure. It also assumes that the network has a backup and recovery scheme. The agents in this environment do not participate in any form of network element management. For example, the agent will not report any channel bank access line card performance issues. This approach is to mount both communication protocol schemes and a reinforcement learning and adaptive agent on a network node in order to choose a better policy by learning and adapting to its environment. The success rate is higher than depending on human operators to resolve error service orders. Future work will focus on how to increase the quality of data in the agent knowledge base to improve the value of information the agent receives. In conclusion, the analysis of infrastructural device data to summarize the existing network state and uncover any known errors, data mining to identify the device and their error type likelihood probabilities and use of reinforcement learning and adaptive control to bypass high error devices to provision services is successfully achieved.

REFERENCES

[1] *Advance Mobile Broadband Wireless Standard) News release*, IEEE Std.802.16m,2011.[Online].Available:http://standards.ieee.org/news/2011/80216m.html

[2] S. Parkvall,D. Astely "The evolution of LTE toward LTE Advanced" JOURNAL OF COMMUNICATIONS, VOL. 4, NO. 3, APRIL 2009

[3] (2008) ITU-R IMT-Advanced website. [Online]. Available: http://www.itu.int/ITU-R/go/rsg5-imt-advanced

[4] Z. Zhao ,B. Willman ,S .Weber ,J. C. de Oliveira "Performance analysis of a parallel link network with preemption" 2006 Conference on Information Sciences and Systems, Princeton University, March 22-24, 2006

[5] S. Kumar1, L. Xu1, M.K. Mandal and S. Panchanathan, "Error Resiliency Schemes in H.264/AVC Standard" Elsevier J. of Visual Communication & Image Representation (Special issue on Emerging H.264/AVC Video Coding Standard), Vol. 17(2), April 2006

[6] Weihmayer, R.; Brandau, R.; , "A distributed AI architecture for customer network control," *Global Telecommunications Conference, 1990, and Exhibition. 'Communications: Connecting the Future', GLOBECOM '90., IEEE* , vol., no., pp.656-662 vol.1, 2-5 Dec 1990

[7] S. Hussain , Z. Hamid , N. S. Khattak, "Mobility management challenges and issues in 4G heterogeneous networks" Proceedings of the first international conference on Integrated internet ad hoc and sensor networks, May 30-31, 2006, Nice, France

[8] Martin, R.L.; , "Future telecommunications services," *Global Telecommunications Conference, 1988, and Exhibition. 'Communications for the Information Age.' Conference Record, GLOBECOM '88., IEEE* , vol., no., pp.721-725 vol.2, 28 Nov-1 Dec 1988

[9] Burachok, R.; Klymash, M.; Laba, H.; , "Research of Quality of Service (QoS) in Telecommunication Networks with Packet Switching," *CAD Systems in Microelectronics, 2007. CADSM '07. 9th International Conference - The Experience of Designing and Applications of* , vol., no., pp.150-155, 19-24 Feb. 2007

[10] Grafe, R.; Friedrich, O.; , "NGN based IPTV & telecommunication services for the Vista Media Center," *Telecommunications, 2009. ConTEL 2009. 10th International Conference on* , vol., no., pp.429-434, 8-10 June 2009

[11] Jin-Xiang Zhang; Jian-Ping Wu; Ji-Long Wang; , "A novel network management architecture based on CORBA and intelligent agent technology," *Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on* , vol.3, no., pp. 1139- 1143 vol.3, 2002

[12] Fang Fang; Mi Zhengkun; , "Strategy of evolution towards mobile agent-based distributed intelligent network," *Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. 2001 International Conferences on* , vol.2, no., pp.747-752 vol.2, 2001

[13] Brusic, I.; Hassler, V.; Lugmayr, W.; , "Deployment of mobile agents in the mobile telephone network management," *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on* , vol., no., pp. 9 pp., 4-7 Jan. 2000

[14] Weihmayer, R.; Brandau, R.; , "A distributed AI architecture for customer network control," *Global Telecommunications Conference, 1990, and Exhibition. 'Communications: Connecting the Future', GLOBECOM '90., IEEE* , vol., no., pp.656-662 vol.1, 2-5 Dec 1990

[15] Guo Leshen; Chen Junliang; , "Next generation intelligent network based on CORBA and agent technologies," *Communication Technology Proceedings, 2003. ICCT 2003. International Conference on* , vol.2, no., pp. 1562- 1565 vol.2, 9-11 April 2003

[16] Polycarpou, M.; , "Intelligent monitoring and fault tolerance in large-scale distributed systems," *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on* , vol., no., pp.480, 6-8 Oct. 2010

[17] Medved, D.; Brlas, K.; Saric, D.; , "Fault analysis and prediction in telecommunication access network ," *Electrotechnical Conference, 2000. MELECON 2000. 10th Mediterranean* , vol.1, no., pp.136-139 vol.1, 2000

[18] Menachi, E.; Giladi, R.; , "End-to-end flexible transport service provisioning in inter-CSP environments [Next-Generation Telco IT Architectures]," *Communications Magazine, IEEE* , vol.48, no.8, pp.118-125, August 2010

[19] Clark, P.; Hammond, C.; Jaenho Oh; Wen Wang; Hadynski, G.; , "QoS-based provisioning of ATM services over DAMA-controlled SATCOM networks," *Military Communications Conference Proceedings, 1999. MILCOM 1999. IEEE* , vol.2, no., pp.1358-1362 vol.2, 1999

[20] Doyle, L.E.; Kokaram, A.C.; O'Mahony, D.; , "Error-resilience in multimedia applications over ad-hoc networks ," *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on* , vol.3, no., pp.1457-1460 vol.3, 2001

[21] Magedanz, T.; Rothermel, K.; Krause, S.; , "Intelligent agents: an emerging technology for next generation telecommunications?," *INFOCOM '96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation. Proceedings IEEE* , vol.2, no., pp.464-472 vol.2, 24-28 Mar 1996

[22] Morreale, P.; , "Agents on the move [mobile software agents]," Spectrum, IEEE , vol.35, no.4, pp.34-41, Apr 1998

[23] Hooper, E.; , "Intelligent Communication Systems and Network Protocols in Infrastructures," Intelligent Signal Processing and Communications, 2006. ISPACS '06. International Symposium on , vol., no., pp.768-771, 12-15 Dec. 2006

[24] T. Farely; K. Schmidt. (2007) Private line homepage on Telecommunication.[Online].Available: http://www.privateline.com/manual/one.html

[25] Bureau of Labor Statistics, U.S. Department of Labor, *Career Guide to Industries, 2010-11 Edition*, Telecommunications , .[Online].Available: http://www.bls.gov/oco/cg/cgs020.htm (visited *December 12, 2011*).

[26] J. Zico Kolter.; A. Y. Ng "Near-Bayesian exploration in polynomial time". In Proceedings of the 26th International Conference on Machine Learning, pages 513-520, Montreal, 2009. ACM Press.

[27] S. J. Russell.; P. Norvig. Articial Intelligence: A Modern Approach. Prentice Hall; 2nd edition, 2002.

[28] A. L. Strehl.; M. L. Littman "An analysis of model-based Interval Estimation for Markov Decision Processes" Journal of Computer and System Sciences, 74(8):1309-1331, 2008.

[29] E. Hazan.; C. Seshadhri "Efficient learning algorithms for changing environments" In Proceedings of the 26th International Conference on Machine Learning, pages 393-401, Montreal, 2009. ACM Press.

[30] Tae-Hyung Kim; Pyeatt, L.D.; Wunsch, D.C.; , "Reconfigurable disruption tolerant routing via Reinforcement Learning," *Neural Networks, 2009. IJCNN 2009. International Joint Conference on* , vol., no., pp.1611-1616, 14-19 June 2009

[31] Angrisani, L.; Napolitano, A.; Vadursi, M.; , "Modeling and Measuring Link Capacity in Communication Networks," *Instrumentation and Measurement, IEEE Transactions on* , vol.59, no.5, pp.1065-1072, May 2010

Automated Financial Data Extraction – An AI Approach

Hassan Alam, Aman Kumar, Cheryl Lee, Yuliya Tarnikova[1]

Abstract— **Using Natural Language Processing and Pattern Recognition methods, BCL Technologies has developed a financial data extraction system called SmartXBRL©, that simplifies and automates ways to create a compliant XBRL document. In this paper we describe the steps necessary to extract and tag detailed Notes data in a 10-Q financial document. Methods to identify and extract the face financial tables, Document and Entity Information (DEI), and Parenthetical are also reviewed, in addition to providing updated results.**

I. INTRODUCTION

XBRL (eXtensible Business Reporting Language) is a technical standard for describing financial and related data. XBRL is a way to assign standard tags to financial data in reports and systems so the values can be analyzed in context by a computer application. XBRL compliance requires that all disclosure elements in a financial report—every numeric value and supporting reference, including footnotes—must be identified with specific "tags" according to a standardized system. These XBRL tags define the context for the disclosure elements, so that they can be analyzed as structured data.

Using example-based Pattern Recognition methods and combining years of developing both EDGAR filings and natural language processing software, BCL Technologies has developed a simplified and automated way to create a compliant XBRL document. By searching historical tags, we have developed a methodology that displays the closest match to the company's label and/or offers three alternatives, plus a link to extend/customize.

This paper is a follow-up to BCL's previously-published work [14], in order to cover topics which were not included in the previous study: tagging or detailed notes tagging. It also includes updated results for work done in the previous study.

This paper is designed as follows. Section 2 briefly describes a sample 10-Q document. Section 3 gives an overview of previous work done on DEI, Tables, Parentheticals, and Detailed Notes. Section 4 discusses further advances made to Detailed Note tagging. Updated results are given in Section 5. Section 6 concludes and talks about future plans we intend to undertake.

II. QUARTERLY FINANCIAL DOCUMENT (10-Q)

Form 10-Q, used for quarterly reports, is filed after the end of each of the first three fiscal quarters of each fiscal year. It is important to understand the content and structure of an input document (10-Q in this case) before designing extraction methods for an automated system.

In a 10-Q document, companies spell out the finalized numbers for the quarter. A 10-Q contains much of the same information as in the earnings press release, but usually to a much greater level of detail. For instance, many companies leave the statement of cash flows out of their earnings press release, but must include it in their 10-Q. 10-Qs can take different forms depending on the company. However, some fields are common in 10-Q documents for all companies.

A typical 10-Q document has mainly three parts: (1) Document Entity Information (details in section 3); (2) Part I – financial information that includes financial tables and Notes, 'Management's Discussion and Analysis of Financial Condition and Results of Operations', 'Quantitative and Qualitative Disclosures about Market Risk', and 'Controls and Procedures'; (3) Part II, broadly speaking, includes sections on legal proceedings, risk factors, unregistered sales of equity securities and use of proceeds, exhibits, signatures, summary of trademarks, and index to exhibits.

	Three Months Ended July 31,		Nine Months Ended July 31,	
	2011	2010	2011	2010
Net revenue:				
Products	$ 4,704	$ 5,127	$ 3,043	$ 4,153
Services and other	289	277	738	817
Total net revenue	-,---	-,---	-,---	-,---
Costs and expenses:				

Fig. 1. Sample 10-Q document (input document)

The SEC mandates that in Year I a company is required to generate XBRL of their financial information (e.g., 10-Q); that is, their DEI, financial tables, and Notes with titles must be reported in the XBRL format in Year I.

III. PREVIOUS WORK

A. Locate and Extract DEI

Document and Entity Information (DEI) typically describes the company or registrant information – administrative, legal, and financial. The broader categories under DEI that need to be extracted from a 10-Q document are: *Document Type (*whether the document is 10-Q or 10K or some other document)*, Amendment Flag* (whether the document is an amendment or an original)*, Document Period End Date, Entity Registrant Name, Entity Central Index Key, Current Fiscal Year End Date, Document Fiscal Year Focus, Document Fiscal Period Focus, Entity Filer Category, Entity Common Stock, and Shares Outstanding*.

The SmartXBRL software identifies and locates DEI from this document based on keyword matching algorithm. In a preliminary study, we found that SmartXBRL's DEI extraction yielded around 89% accuracy. This is an impressive performance from a theoretical point of view; however, for a commercial software application that is involved in SEC filings, higher accuracy is required. In order to attain that company-specific accuracy, the DEI fields are manually edited. Manual editing is detailed more fully in [14].

With manual modifications, the DEI extraction accuracy is close to 100%, as per SEC's filing requirements. These manual steps complete the DEI page and the DEI page is now ready to be processed toward XBTL filing.

B. Locate and Extract Financial Tables

SmartXBRL locates the financial tables in a document based on keywords and pattern matching methods. The typical steps of any table information extraction system are as follows: (1) Identify and differentiate a table layout from normal text, images, bulleted points, etc., in a document. (2) Identify the header lines and differentiate between a column or row header and the title in a table, and determine the range or scope of each header in a column or a row. (3) Link the cell content to the cell tag (header, title, etc.). In HTML/XML-tagged documents, tables usually link their markup tags or metadata (titles, column headers, etc.) and content of a table cell together. A series of computations are required to retrieve and link metadata and the content of a cell.
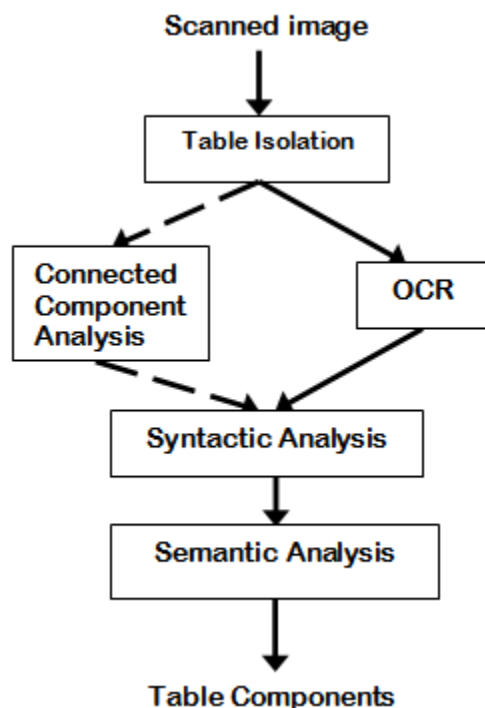
Fig. 2. System flow chart of BCL's earlier Table Detection System [10].

In its earlier related research work [10], BCL used heuristic-based methods for extracting table information by utilizing the absolute position of text elements in a document. This algorithm, visually described in Fig. 2, yielded the results in Table I.

Table I

Results of BCL's previous Table Detection System (Tupaj et. al. 1996). Here P is precision and R is recall.

	Number of Samples	Detection of Tables		Detection of Cells	
		P	R	P	R
Scientific Journals	100	0.930	0.924	0.970	0.916
Financial Tables	20	0.714	0.750	0.920	0.890

In more recent work, BCL was able to (1) isolate embedded tables from documents, and (2) identify table components such as title blocks, table entries, and footer blocks. The previous heuristic algorithm showed greater precision given scientific journals than financial tables. Compare this to Table II. [14]'s algorithm shows not only general improvement of both precision and recall, but also dramatic improvement within the specific domain of financial documents.

Table II

Evaluation of BCL's existing commercial Table Information Extraction system. Here P is precision and R is recall.

	Number of 10-Q Samples	Detection		Extraction of Cells/Content	
		P	R	P	R
Tables	100	.93	.95	.96	.97
DEI	100	.96	.94	.89	.94
Parenthetical	100	.84	.96	.81	.85
Notes	100	.90	.93	.83	.94

Since the publication of [14], BCL has been able to further improve table detection results. See the Updated Results section.

1) Table Detection and Segment Detection
Table detection is the first task in any table information extraction process. The goal is to differentiate and identify tables from non-tables, such as blocks of text, images, graphics, lists, etc., within a document. In our existing application, tables are identified by assuming that text and images are all represented by a bounding rectangle with sides parallel to page borders.

Next, we find segments–page blocks—that are (1) surrounded by the white rectangles described before or page boundaries from all 4 sides, and (2) do not overlap with any of those rectangles.

The algorithms for finding a minimum set of rectangles that covers all white space of a page (none of which overlaps with objects on a page) and detecting segments in a document are given in [14].

Once we have identified a table as a table in a document, the next step is to identify and extract only those tables that represent the face financial tables - Income Statement, Cash Flow, Balance Sheet, and Stockholders' Equity. In the following section we describe the pattern-matching table extraction, architecture focusing on the four face financial tables.

2) Table Extraction Architecture- Face Financials
The first step in table extraction is a *concept search,* applied to the content of a financial document. This step entails searching keywords representing a concept within a sentence. The second step is a *template pattern search*, which searches for a complex pattern consisting of a few words and phrases, rather than searching for one word, as in the concept search. The

template pattern search identifies relationships between matched objects in the defined pattern as well as recognizing the concept itself. The textual information thus extracted is saved in an object-oriented database.

There are three major steps in extracting tables within SmartXBRL: (1) pre-processing; (2) pattern matching; (3) table template generation.

The table extraction architecture is depicted in Fig.
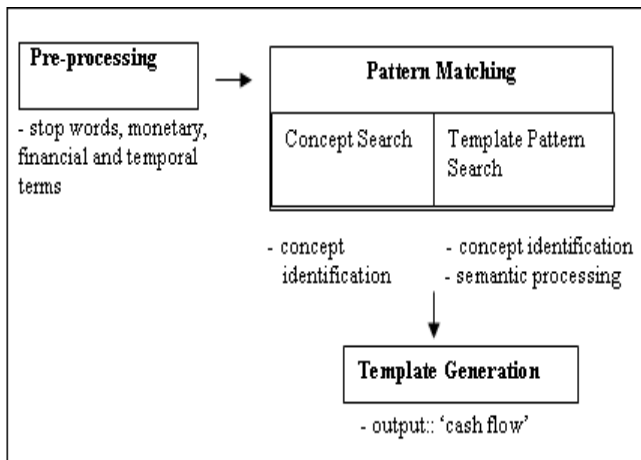
Fig. 3. Table Extraction Architecture

In the pre-processing phase, semantically empty terms (e.g., determiners, etc.), monetary, numeric, and temporal expressions in the financial document are identified. These terms are either put into the stop word list or extracted and stored separately. The concept search module identifies concepts in a sentence. Concepts, such as the names of the four financial tables—<income statement>, <balance sheet>, <cash flow>, and <shareholders' equity>— identified during the pattern matching steps are used to select relevant text strings and fillers to go into a slot.

Keywords representing the same concept are grouped into a list and used to recognize the concept in a sentence. The list is written in a simple format: (*concept-name word1 word2...*).

The concept search module recognizes the concept when a word in the list exists in the title sentence. Processing text in a narrow domain - financial filings, results in concepts often correctly identified from the simple list, since keywords are usually used in a particular meaning of interest in that domain. In addition to the keyword pattern matching the table, extraction algorithm looks at the distance of the table layout from the title of the table.

C. Generate Auto Parenthetical

Parenthetical tables are built when the line item in a face financial table discloses an amount in the input financial document.

BCL's SmartXBRL automatically identifies and extracts parenthetical information from a base financial table and builds a separate table for further processing (tagging and linking). It extracts the parenthetical labels based on a string matching algorithm that first scans the table for numeric data (numbers with units– thousands, millions etc.). Then, label boundaries are identified that basically correspond to the cell boundary in a table. The new parenthetical table is then constructed based on this information–date, currency ($, yen, etc.), units, and line items.

D. Locate and Extract Notes

Notes are generally added to the financial documents to explain items on the financial statements. For example, companies usually group other expenses in a Note instead of showing the expenses individually on, say, an income statement. For Year 1, filers' Notes are expected to be tagged as 'block' text. In SmartXBRL, we locate and extract Notes and block tag the text in the Notes. Following is a snapshot of a Note from a 10-Q document:
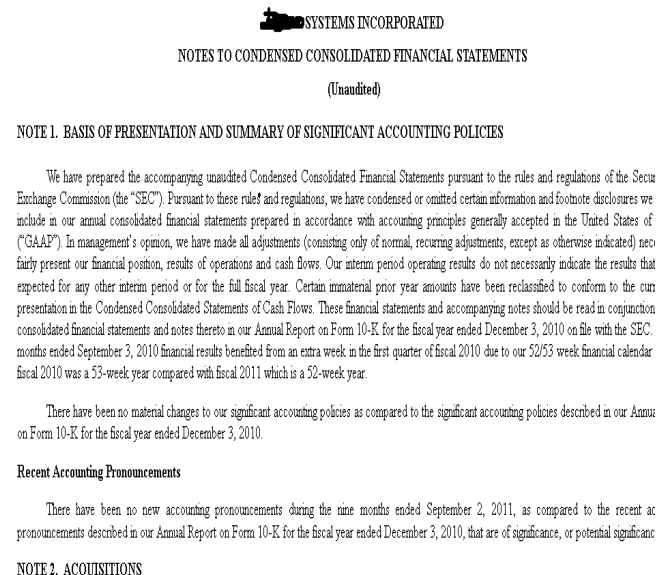
Fig. 4. Notes sample

The identification and extraction of the Notes is done based on pattern-matching algorithm that also looks into the typological position of the notes block in the document. Once the notes are extracted, the title of the notes is tagged and the text content of the note is put in

a block within a tagged note title. Here is a snapshot of the Notes output, as given by SmartXBRL:

Fig. 5. Output Notes (SmartXBRL)

IV. FURTHER WORK ON DETAILED NOTES

Detailed tagging of XBRL documents requires the following items to be located and tagged:

1. Accounting Policies
2. Notes Tables
3. Notes tables line items
4. Numerical items in the notes text

A. Accounting Policies

Accounting Policies are typically one of the Notes in a 10-K form. They may also occur in 10-Q forms. The individual accounting policies are groups of paragraphs (with and without tables) with a title of the significant policy.

The accounting policies need to be detected as a group. Each accounting policy needs to be tagged as such.

SmartXBRL uses a learning system to locate the Accounting Policy Note, extract, and tag each individual Accounting Policy.

Fig. 6. Accounting policies detected and tagged as groups

B. Notes Tables

Each Note may have one or more tables. These tables need to be individually extracted, their subject recognized and the table tagged. To identify and tag the table SmartXBRL uses a natural language-based learning engine which performs the following tasks:

1. Locates the descriptor paragraph above the table
2. Parses the first sentence in the paragraph to identify the label to be applied to the table
3. Uses a learning engine to determine the tag to identify the table

C. Detail Notes Tables Line Items

Each of the line items in a notes table is extracted as a possible xbrl tag. This is similar to tagging the financial statements and a similar algorithm is used for this tagging.

D. Numbers Within Notes Text

The final item that needs to be tagged is each instance of numeric financial items in the notes text. SmartXBRL tags the items with the following steps:

1. Identified each sentence in a note that includes a numeric value of interest. These include monetary items, shares amounts or durations of times. Some numeric values are not of interest to this step, such as dates.
2. The sentences containing the numeric values are parsed, then extracting the label and date associated with the numeric value.
3. The numeric value is placed in a table with the date as the column header, the label as the row header, the numeric value as the entry, and the units and data type as the descriptors.
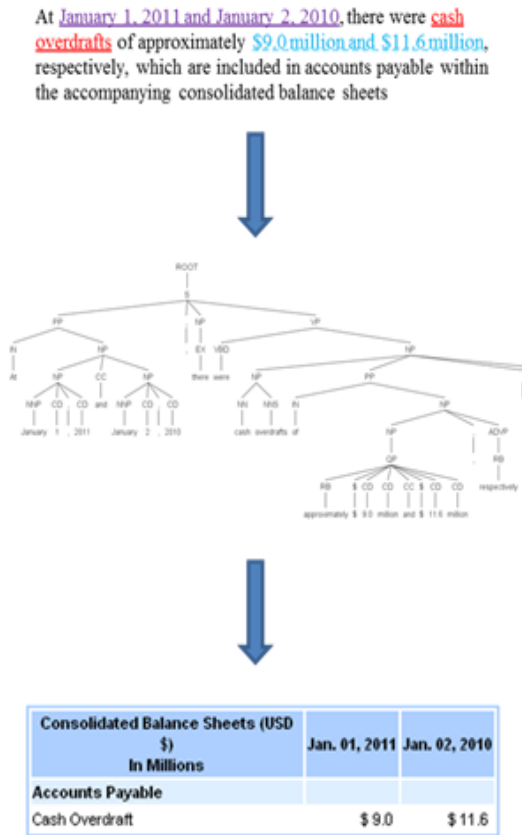
Fig. 7. Note text numbers steps

V. UPDATED RESULTS

SmartXBRL's performance on identifying and extracting (1) face financial tables; (2) DEI; (3) parentheticals; and (4) financial notes from a 10-Q document can be measured in two areas: accurate *detection* of tables, DEI, parentheticals, and notes, and accurate *extraction* of their content. For evaluations, we use standard recall and precision measurements to evaluate our system. Precision (P) measures the degree of correct identification, and recall (R) measures the degree of correct identification with respect to the total number of elements. Table III gives the R&P measurements for SmartXBRL identification and extraction system.

Table III

Evaluation of BCL's existing commercial Table Information Extraction system. Here P is precision and R is recall.

	Number of 10-Q Samples	Detection		Extraction of Cells/Content	
		P	R	P	R
Tables	150	.96	.95	.96	.97
DEI	150	.97	.95	.93	.96
Parenthetical	150	.83	.96	.78	.87
Notes	150	.93	.95	.86	.95

Accuracy for Parenthetical has decreased given input files that have more complicated parenthetical text to extract. We will continue to work on refining our Parenthetical extraction algorithm to increase the accuracy.

BCL has done a number of detail tagged filing during the 2012 fiscal year. Based on its tests with actual data, it achieved the following levels of accuracy.

Table IV

Results of BCL's detail tagging.

Item	Item Type	Action	Accuracy
Accounting Policy Note	Full Accounting Note	Detection and Extraction	90%+
	Individual Policy	Detection and Extraction	90%
	Individual Policy	Tagging	75%
Notes Table	Individual Table	Detection and Extraction	95%
	Table Label	Detection and Extraction	75%
	Individual Table	Tagging	76%
Detail Notes	Labels	Tagging	70-90%
Text Notes	Numeric Values	Detection and Extraction	95%
	Date	Detection and Extraction	95%
	Labels	Detection and Extraction	75%

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have elaborated on the process of extracting and tagging detailed Notes, in addition to providing updated results. We follow keyword based pattern matching methods to extract tables, DEI, parentheticals and notes. Once these items are extracted, the labels and numbers in these items are tagged based on standard XBRL tags.

Since last publication, we have fine-tuned our algorithm to increase the accuracy of Parenthetical tables, both for detection and extraction of the content.

For detailed tagging, we have developed Natural Language Processing methods to do detail tagging of Notes items and tables.

In the future we will try to work on improving the accuracy for both Parenthetical tables and Notes. In addition, we will extend the extraction methods to other financial documents such as 10K, 6K, 8K, etc.

REFERENCES

[1] H. Alam, A. Rahman, T. Tjahjadi, H. Cheng, P. Llido, A. Kumar, R. Hartono, Y. Tarnikova, and C. Wilcox, "Development of Spoken Language User Interfaces: A Tool Kit Approach," *Lecture Notes in Computer Science LNCS 2396,* T. Caelli, A. Amin, R. Duin, M. Kamel and D. Ridder*, Eds*., 2002, pp. 339-347.

[2] M. Hurst, "Layout and language: an efficient algorithm for text blocks detection based on spatial and linguistic evidence," in: *Proceedings of Document Recognition and Retrieval* VIII. New York: Springer-Verlag, 1985, pp. 56–67.

[3] http://www.w3.org/2009/03/xbrl/cfp.html

[4] http://www.w3.org

[5] http://www.xbrl.org

[6] http://www.fdic.gov

[7] http://www.sec.gov

[8] M. Hurst, "The interpretation of tables in texts*,"* Ph.D. Dissertation, School of Cognitive Science, University of Edinburgh 2000.

[9] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th International Conf. on Machine Learning*. 2001.

[10] S. Tupaj, Z. Shi, C. Chang, H. Alam, "Extracting tabular information from text files," Available at http://www.ee.tufts.edu/hchang/paper1.ps, Tufts University, Medford.1996.

[11] Y. Wang, "Document Analysis: Table Structure Understanding and Zone Content Classification," PhD dissertation, Washington University, 2002.

[12] X. Wei, B. Croft, A. McCallum, "Table Extraction for Answer Retrieval, " in: *Information Retrieval Journal*, vol. 9, issue 5, 2006. pp. 589-611.

[13] B. Yildiz, K. Kaiser, and S. Miksch, "pdf2table: A Method to Extract Table Information from PDF Files," in: *Proceedings of the 2nd Indian International Conference on Artificial Intelligence* (IICAI05), Pune, India*,* 2005.

[14] H. Alam, A. Kumar, C. Lee, and Y. Tarnikova, "A Pattern Recognition Approach to Automated XBRL Extraction," in: *IEEE Computational Intelligence for Financial Engineering and Economics 2012*, New York City, 2012.

Visualizing Logical Representation and Reasoning

Sabah Al-Fedaghi

Computer Engineering Department, Kuwait University, Kuwait

Abstract - *Current visualizations of logical representation and reasoning do not provide completeness or clarity of the basic static and dynamic features, creating a conceptual gap that sometimes causes misinterpretation. This paper introduces an alternative methodology that produces a conceptual picture on which constraints, rules, synchronization, and logical control can be superimposed. Several examples are given to demonstrate the capabilities of the method.*

Keywords: artificial intelligence; diagrams; logic representation and reasoning; conceptual model

1   Introduction

Euler circles, Venn diagrams, and Peirce's existential graphs [5] are some of the available diagrammatic tools designed to facilitate logic understanding, argumentation, and reasoning. These diagrams are hardly used anymore in current formal languages and rules of logic representation systems [4][7]; nevertheless, "there is no principal distinction between inference formalisms that use text and those that use diagrams. One can have rigorous, logically sound (and complete) formal systems based on diagrams" [3, p. 214]. Recently, in the area of computer science, especially in artificial intelligence, research into diagrams as representational systems (e.g., UML [6]) has been increasing. Diagrams can also be used to facilitate logical reasoning.

> Logical reasoning with diagrams is often carried out in virtue of their depiction of all possible models of a situation, ... A single diagram is often an abstraction over a class of situations, and once a suitable diagram has been constructed, inferences can simply be read off the representation without any further manipulation. [7]

This paper proposes a new kind of flow-based diagrammatic representation of logic formulas based on a flow-based modeling methodology used in software engineering. Current diagrammatic representations, e.g., Venn diagrams, lack certain dynamic features in a way that does not integrate structure with movement of truth values. Advantages of this flow-based approach include the following:

- An explicit depiction that is represented only implicitly in logical formulas from which it must be inferred, sometimes causing misinterpretation.

- A better understanding of the reasoning process, especially in a human/computer interaction environment
- A mechanism to formalize diagrammatic reasoning [3] through explicit description of operations and relations among objects in the diagram.

The approach utilized in the paper, called the Flowthing Model (FM), will be briefly described in the next section.

2   Flowthing Model

Flow models have been used in several applications (e.g., [8-14]). A flow model is a uniform method for representing things that "flow," i.e., things that are created, processed, released, transferred, and received. "Things that flow" include information, materials (e.g., goods), and money. They flow in *spheres*, i.e., their environments. A sphere is different from a set in the sense that a set is a static structure, whereas a sphere includes flowthings (current members) at different stages in a progression and possible directions (lines) of movement from one stage to another, or movement from/to the spheres of the flowthings. A sphere may have subspheres.

An FM representation is a depiction of the structure of a scheme resembling a road map of components and conceptual flow. A *component* comprises *spheres* (e.g., those of a company, a robot, a human, an assembly line, a station) that enclose or intersect with other spheres (e.g., the sphere of a house contains rooms which in turn include walls, ceilings). Or, a sphere embeds flows (called *flowsystems*; e.g., walls encompass pipes of water flow and wires of electrical flow).

Things that flow in a flowsystem are referred to as *flowthings*. The life cycle of a flowthing is defined in terms of six mutually exclusive *stages*: creation, process, arrival, acceptance, release, and transfer.

Fig. 1 shows a flowsystem with its stages, where it is assumed that no released flowthing flows back to previous stages. The reflexive arrow in the figure indicates flow to the Transfer stage of another flowsystem. For simplicity's sake, the stages Arrive and Accept can be combined and termed *Receive*.
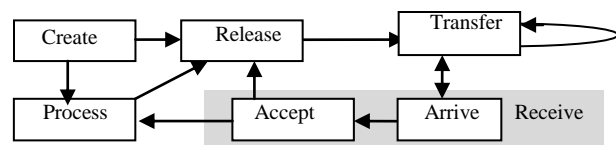
Fig. 1. Flowsystem

The *stages* of the life cycle of a flowthing are mutually exclusive (i.e., a flowthing can be in one and only one stage at a time). All other states or conditions of flowthings are not generic stages. For example, we can have *stored* created flowthings, *stored* processed flowthings, *stored* received flowthings, etc.; thus *stored* is not a generic stage. In contrast, there are no such stages as, e.g., *created-processed, received-transferred*, or *processed-received* stages. Flowthings can be released but not transferred (e.g., the channel is down), or arrived but not accepted (wrong destination), …

In addition to flows, *triggering* is a transformation (denoted by a dashed arrow) from one flow to another, e.g., a flow of electricity triggers a flow of air.

3   Applying FM

A first-order formula such as p(x) indicates a *sphere* with two types of *flowthings*: x, and Truth/False values. Thus, p(x) can be conceptualized as a sphere p formed from two subsystems (Body (symbolic expression), and Truth), as shown in Fig. 2. When the logical expression is clear, only the truth/false value of the expression is diagrammed.
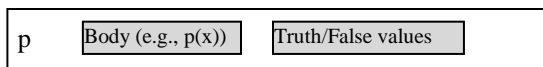
Fig. 2. p as a sphere with two flowsystems

4   Sample Applications

Valid inference is usually specified as p→q (material implication), which includes p, q, and the implication symbol → as a connection between p and q. In this case, true p must imply (logic) true q. False p can imply (logic) true or false q.

Many conceptualizations have been presented that view logical reasoning in ways different from the classical symbolic representation. For example, the so-called proofs-as-programs views a direct relationship between computer programs and proofs; and proof-system models of computation are considered structurally the same kind of objects [2]. In such an approach, p→q is a function in a program that takes in argument p and returns q.

The semantic difficulty with the notion of implication is well known. For example, correlation differs from causation, and confusing them leads to logical fallacies. From the FM point of view, a forward arrow → is misleading, because the flow of truth values can occur in both directions. A true value of p flows, and may be processed, to reach q, while an assignment of truth value to q flows (can be thought as backward flow), to reach p; that is, true in p results in true in q, and false in q results in false in p, or ¬q→¬p. Thus, the problem with the functional structure of the implication in proofs-as-programs, e.g., p(q), is that it is possible that q is given and p is the argument. FM presents an alternative representation.

Consider the implication p→q. Fig. 3 shows its corresponding FM representation. Only the truth value

flowsystem is shown. Thus, p either is assigned truth value (circle 1) or receives it from a previous deduction step (2). In either case the value flows to the implication flowsystem → (3), where it is processed (4) to trigger (5) generating a truth value in → (6) that flows to q (7)

The conceptual flow process in this view is approximately analogous to the physical flow of a liquid, as illustrated in Fig. 4. The truth table stands for the mixer buttons that generate the output according to certain selections. Fig. 5 shows the FM representation of p↔q.
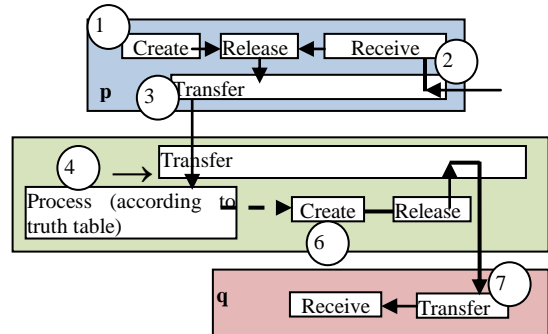
Fig. 3. FM representation of p→q
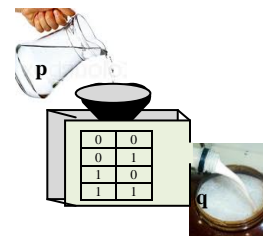
Fig. 4. Approximate illustration of the notion of flow
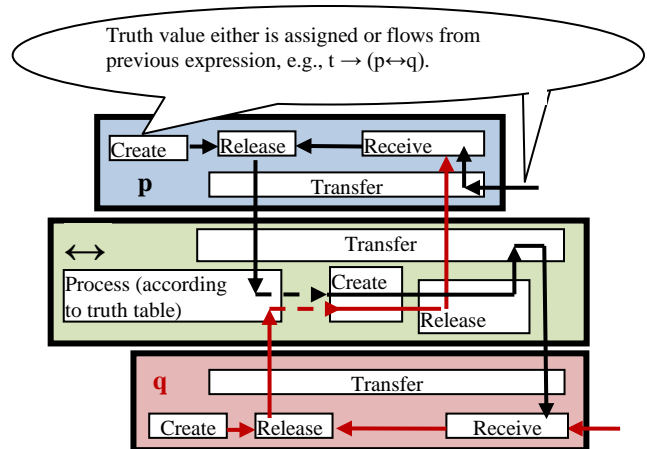
Fig. 5. FM representation of p↔q

Hoffmann [1] gives the following deductively valid argument: *If Paul is a being, then Paul is responsible for what he did*. "It is possible to transform any argument into a deductively valid argument simply by introducing a fitting additional premise like the "if-then" statement" [1]. This can be modeled as shown in Fig. 6.
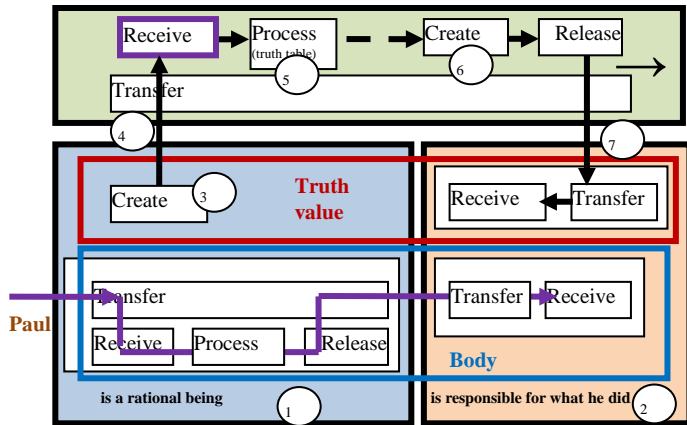
Fig. 6. FM representation of *If Paul is a rational being, then Paul is responsible for what he did*

The figure illustrates what we previously called the body of the expression of p(x) along with a flowthing x. Thus, Paul flows into the spheres of *is a rational being* (circle 1) and *is responsible for what he did* (circle 2). The two predicates are connected with implication:

is a rational being (Paul)→*is responsible for what he did*(Paul)

The implication is conceptualized as: when the first predicate receives (assigned Paul), Paul flows to the second one. Simultaneously, if *is a rational being* (Paul) is given the truth value, say *true* (3), then this value flows to the sphere (upper box) of the implication (4) to be processed using the truth table (5). This processing triggers creation of (6) the corresponding truth value that flows to the second predicate (7), thus assuming the material implications, *true→true*, and *false→true* or *false*, where in the second case two flowthings *true* and *false* are generated to flow to *is responsible for what he did*(Paul). This reminds us of the notion of *firing* in Petri nets, where movement of tokens to a place is generated. Accordingly, in case the first predicate is false, the second has two flowthings, true and false. If we take a snapshot of the second predicate, we find its truth value set is *true* and *false*.

To maintain consistency in the second predicate, it is possible to add a Process stage that outputs one flow value at a time:

is a rational being (Paul) IS FALSE→
is responsible for what he did(Paul) IS TRUE

is a rational being (Paul) IS FALSE→
is responsible for what he did(Paul) IS FALSE

The FM representation presents a complete picture, in an anology of a city, of the buildings, streets, and streams of traffic flow. Constraints, rules, synchronization, and logical control can superimposed on such a map. Current visualization processes do not provide this completeness and clarity of representation of the basic static and dynamic features of the reasoning process.

As a third example, consider a simple exercise of proving that ¬r from:

1. $p \rightarrow \neg q$
2. $r \rightarrow q$
3. $\neg p$

Fig. 7 shows the corresponding FM representation, where the two implications in (1) and (2) intersect in q. Starting with p being true (circle A in the figure), then applying the implication (1) (circle B), the true value flows to negation (circle C) to be received in q as false (D). That is, if p is true, then q is false.

Now, q is at the intersection of the two implications (1) and (2), and since q has received the value *false*, then by backward flow (E to F), r must be false; otherwise a contradiction is created. In this consistent system where a process (G) confirms that q is true by implication (1) and is true by implication (2), accordingly a *false* value of q is created (H) that completes the circle as it flows through implication (2) (circle I) to r (J) to end in the negation of r to be processed as *¬r is true*, i.e., if r is false, then ¬r is true.

On the other hand, if p is false, then q (through the negation in implication 1) can be false or true (material implication). Thus, by backward flow in implication (2), r can be true or false, and ¬r can be true or false according to the value of r.

The same diagram also describes all involved flows if they are explicitly specified. For example, we verify that ¬r cannot be false in the given system, so we assume that and derive a contradiction.

This is accomplished by explicitly adding relevant flows, as shown in Fig. 8, which shows the part of Fig. 7 that includes the newly added flows. In the figure, *¬r is false* is created (circle L), the false value flows to r (M), where it is processed to create a true value of r (N). The value of r then flows by the second implication to q. By the backward flow in $p \rightarrow \neg q$, false ¬q is processed to derive *p is false*, which contradicts (3).

5   Conclusions

The flow-based logical representation and reasoning introduced in this paper is worth further investigation because it is unique for its detailed visual depiction of logical structure. Potential applications include using it for understanding and analysis in studying logic.
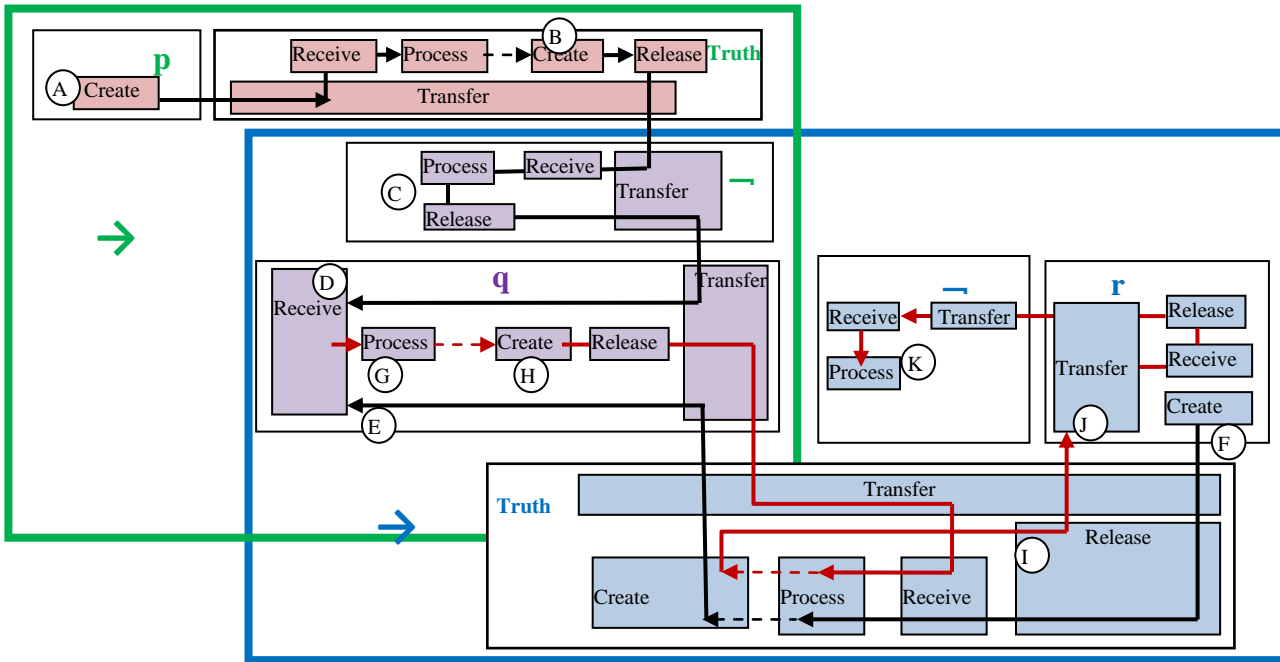
Fig. 7. FM representation of the example
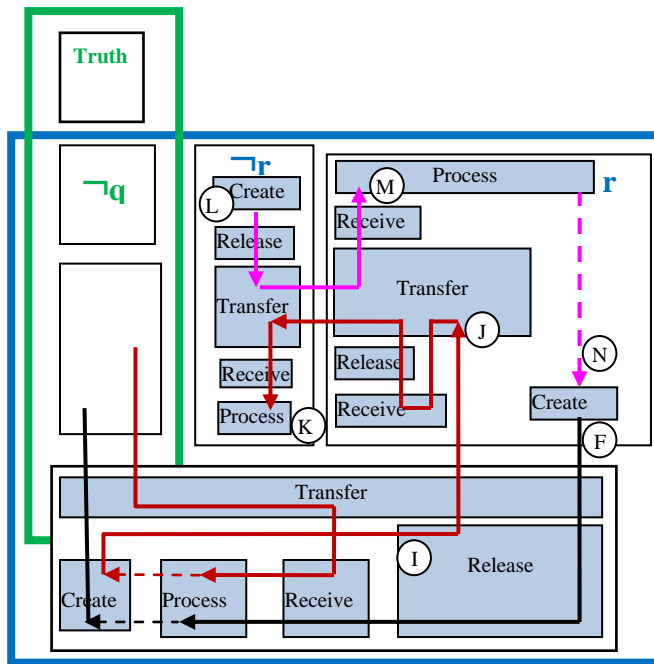
Fig. 8. Partial representation of assigning false to ¬ r

6   References

[1] M.H.G. Hoffmann, Logical Argument Mapping (LAM): A Manual, Mar 20, 2012. http://www.prism.gatech.edu/~mh327/LAM/

[2] W. A. Howard, "The formulae-as-types notion of construction," in Essays on Combinatory Logic, Lambda Calculus and Formalism, J. P. Seldin, J. R. Hindley, and H.B. Curry, Eds. Boston, MA: Academic Press, 1980, pp. 479–490.

[3] J. Barwise and J. Etchemendy. "Computers, visualization, and the nature of reasoning", in The Digital Phoenix: How Computers are Changing Philosophy, T. W. Bynum, D. D. Roberts, and J. H. Moor, Eds. London: Blackwell, 1998, pp. 93–116.

[4] F. Dau and T.U. Dresden, "Mathematical logic with diagrams based on the existential graphs of Peirce," Darmstadt University of Technology. DOI 10.1.1.60.1453

[5] D. Roberts, The Existential Graphs of Charles S. Peirce. The Hague: Mouton, 1973.

[6] D. Berardi, D. Calvanese, and G. De Giacomo, "Reasoning on UML class diagrams," Artificial Intelligence, vol. 168, pp. 70–118, 2005.

[7] S.-J. Shin and O. Lemon, "Diagrams," Stanford Encyclopedia of Philosophy.

[8] S. Al-Fedaghi, "Alternative representation of aspects," 10th IEEE International Conference on Information Technology : New Generations, IEEE ITNG 2013, 15-17 April, 2013, Las Vegas, Nevada, USA.

[9] S. Al-Fedaghi, "Flow-based Enterprise Process", International Journal of Database Theory and Application, Vol. 6, No. 3, pp. 59-70, 2013.

[10] S. Al-Fedaghi and Faleh Faihan, "A Conceptual Visualization of Industrial Control Systems: Electrical Power System", International Review of Automatic Control (IREACO), Vol. 6, No. 2, March 2013.

[11] S. Al-Fedaghi, "A Method for Modeling and Facilitating Understanding of User Requirements in Software Development", Journal of Next Generation Information Technology, Vol. 4, No. 3, pp. 30-38, 2013.

[12] S. Al-Fedaghi, "Schematizing Proofs based on Flow of Truth Values in Logic", IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2013), Manchester, UK.

[13] S. Al-Fedaghi, "How to Diagram your Drama Story", HCI International 2013, 21 - 26 July 2013, Mirage Hotel, Las Vegas, Nevada, USA. Lecture Notes in Computer Science/Artificial Intelligence, Springer. Extended version is published in the Communications in Computer and Information Science (CCIS) series, Vol. 374, 2013, pp 531-535 - Springer.

[14] S. Al-Fedaghi, "Specification of Security Using Stereotypes", 3rd World Conference on Information Technology, November 14 – November 17, 2012, at the University of Barcelona, Barcelona, Spain. This paper is also published in AWERProcedia Information Technology & Computer Science, Vol. 3, pp 1857-1863, 2013. Available from: http://www.world-education-center.org/index.php/P-ITCS

Performance Analysis of Fuzzy Logic And PID Controller for PM DC Motor Drive

Khalid Al-Mutib[1], N. M. Adamali Shah[2], Ebrahim Mattar[3]
[1]King Saud University, Riyadh, Saudi Arabia, muteb@ksu.edu.sa
[2]King Saud University, Riyadh, Saudi Arabia, anoormuhamed@ksu.edu.sa
[3]University of Bahrain, College of Engineering. Kingdom of Bahrain, ebmattar@uob.edu.bh

Abstract - *This paper proposes to control the speed of the permanent magnet (PM) DC motor drive using fuzzy logic control (FLC) based on Matlab simulation environment. The proposed speed controller is developed based on fuzzy logic, to minimize maximum overshoot and steady state error. A comparison study and analysis of proposed fuzzy logic control to classical PID control system applied to PM DC motor is presented. The entire system is modeled and simulated using the matlab simulink tool box. The detailed simulation results confirm the better reduction in maximum overshoot and steady state error by the proposed FLC than the classical PID controller.*

Keywords: DC drive, fuzzy logic controller, speed control, PID.

1. Introduction

DC motor provides easy controllability and high performance, due to its speed can be varied within wide boundaries. DC drive has vast applications such as electric traction, electric cranes and robotic where manipulation of speed controller is required to perform their tasks [4]. In general, the control of systems is difficult and mathematically tedious due to their high nonlinearity properties. This leads to the need of more efforts to obtain more exact mathematical model of the plan to be controller and not simple mathematical operations. Fuzzy logic controller has proven effective for complex, non-linear and imprecisely defined processes for which standard model based control techniques are impractical or impossible [10 - 12]. The FLC is indeed capable of providing the accuracy required by high performance drive system without need of mathematical model. FLC offers simple, quicker and more reliable solution because they are viewed in form of set theory [2]. It also accommodates nonlinearity without utilization of mathematical model. The most important features of FLC is, its simplicity and less intensive mathematical design requirements. The speed of the DC motor is normally controlled by varying armature voltage or flux. In armature controlled dc motor the desired speed is obtained by varying armature voltage. However, the conventional proportional integral derivative(PID) controller has difficulty in dealing with dynamic speed tracking due to parameter variations, and load disturbances. Hence these

controllers show high performance only for one unique act point and also it has high starting overshoot, sensitivity to controller gains and sluggish response due to sudden change in load. The FLC provides a systematic way to incorporate the human intelligence in the controller without knowing the mathematical model of the system. The stability of the system and wide range of operating speed are achieved through fuzzy logic controller. When the optimum membership functions are chosen for input and output of the FLC then it works with self-turing capability and its stability depends upon rule base.
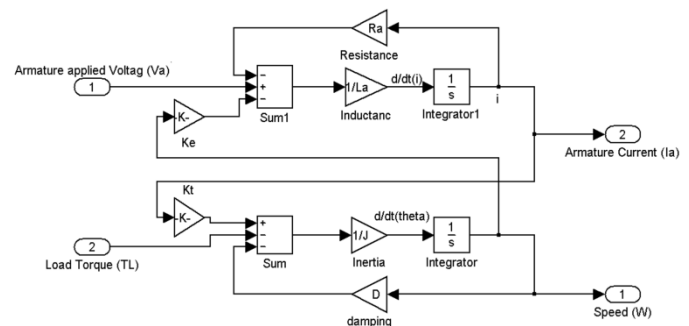
Fig. 1. Matlab/simulink model of DC motor

In this paper, the speed response of DC motor exposed to variable armature voltage is designed based on fuzzy logic controller. The DC motor is operated for a required reference speed under loaded operating condition. Then, to make performance comparison of proposed model, the speed of the system is compared with classical PID system. The proposed system provides low maximum overshot and steady state error. Numerical simulations confirm the accuracy of the propose model.

2. DC motor models

A linear model of permanent magnet (PM) DC motor can be represented by their mechanical equation and electrical equation as follows [1].

$$Va = La\frac{dIa}{dt} + RaIa + eb \qquad (1)$$

$$KmIa = Jm\frac{d\omega m}{dt} + Bm\omega m + TL \qquad (2)$$

where *Va* is the applied armature input voltage, $e_b = K_b\omega_m$ is the back electro-motive-force (EMF) voltage, *Ia* is the armature current TL is the load torque and ω_m is the rotor angular speed. The dynamic model of the system is formed using these differential equations and Matlab simulink blocks as shown in figure 1. The armature reactions effects are neglected since the motor used has either inter-poles or compensating winding.
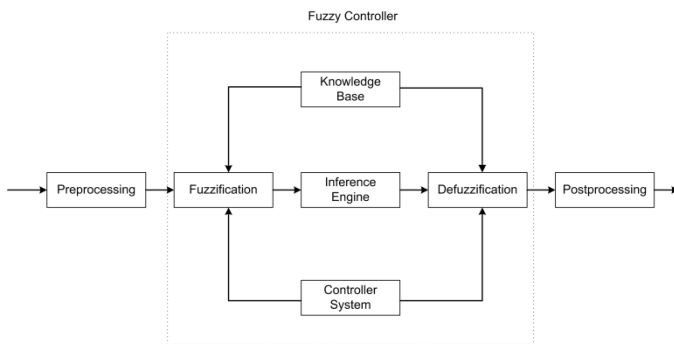
Fig. 2. Fuzzy Logic Controller

3. Fuzzy Logic Controller

3.1. Description

Fuzzy logic controller contains four main parts, out of which two perform transformations. They are fuzzifier (transformation 1), knowledge base, Inference engine and defuzzifier (transformation 2). Fuzzification measure the values of input variable and converts them into suitable linguistic values. Knowledge base consist a database and provides necessary definitions, which are used to define linguistic control rules. This rule base characterized the control goals and control policy of the domain experts by means of a set of linguistic control rules. Decisionmaking logic or inference mechanism is main part of fuzzy controller. It has the capability of simulating human decision making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic. Defuzzification is a scale mapping, which converts the range of values of output variables into corresponding universe of discourse and also yields a non-fuzzy control action from an inferred fuzzy control action. This transformation is performed by membership function (MF). There are number of MF and their shapes are initially defined by the user.

3.2. Input and Output Membership Functions of FLC

The triangular MFs are selected for both input and output variable. There are five MFs for input variable *e* and two MFs for input variable *ce*, whereas there are five MFs for output variable. Depending upon the input variable values, the output

variable value is to be decided from the encoded in the rules as shown in figure 3. Fuzzy inference consist of two processing methods namely, Mamdani's method and Sugeno or Takagi-Sugeno-Kang method to calculate fuzzy output [9]. Out of it Mamdani is more suitable for DC machine and induction machine control. The fuzzy output from the fuzzy inference is process through defuzzification to get the crisp value.

1. If (Error is Adam_PL) then (Control is AA_PL) (1)
2. If (Error is Adam_NL) then (Control is AA_NL) (1)
3. If (Error is Adam_C) and (Adam is AF_N) then (Control is AA_NM) (1)
4. If (Error is Adam_C) and (Adam is AF_P) then (Control is AA_PM) (1)
5. If (Error is Adam_NM) then (Control is AA_NM) (1)
6. If (Error is Adam_PM) then (Control is AA_PM) (1)
7. If (Error is Adam_C) then (Control is AA_C) (1)
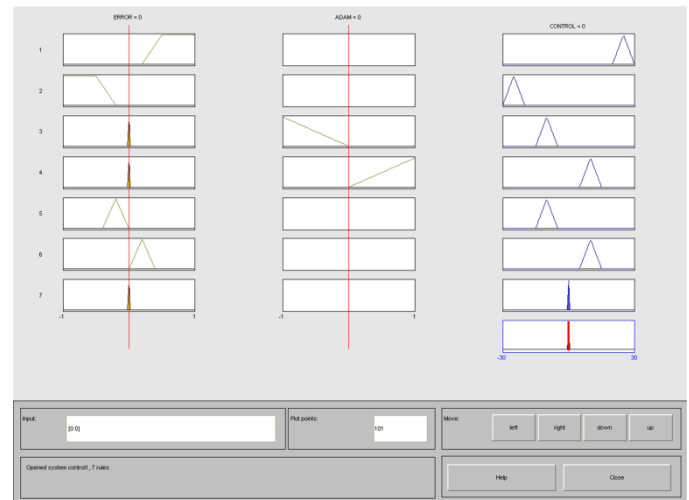
Fig. 3. Fuzzy rules for speed control

The final output torque is then calculated as the average of the individual centroid, weighted by their height (degree of membership). The FLC output torque is applied to the PWM. The PWM controls the DC motor to the desired speed.
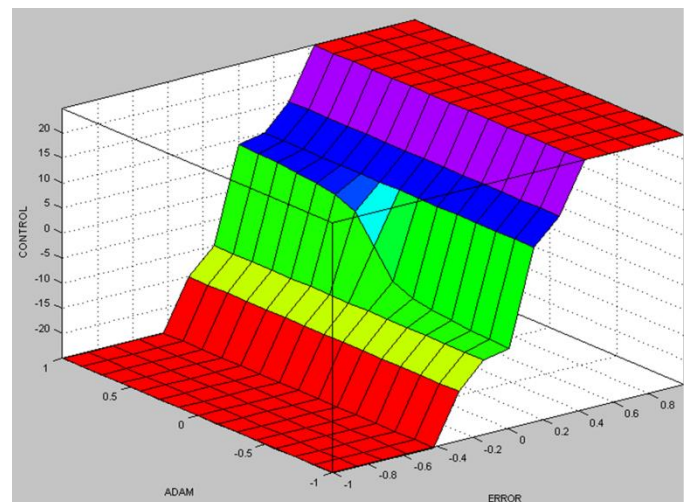
Fig. 4. Surface of the rule

3.3. Design of FLC

The controller observes the speed error signal $e(t)$ and corresponding updates the controller output so that the actual motor speed $y(t)$ matches the reference set speed $r(t)$. There are two input signals to the fuzzy controller, one is the error $e(t)$ that is the difference of set speed $r(t)$ and actual speed $y(t)$, and the other is the derivative of error $ce(t)$.

In a discrete system, $de(t)/dt = \Delta e(t)/\Delta t = ce(t)/Ts$, where $ce(t) = \Delta e(t)$ in the sampling time Ts. With constant Ts, $ce(t)$ is proportional to $de(t)/dt$.
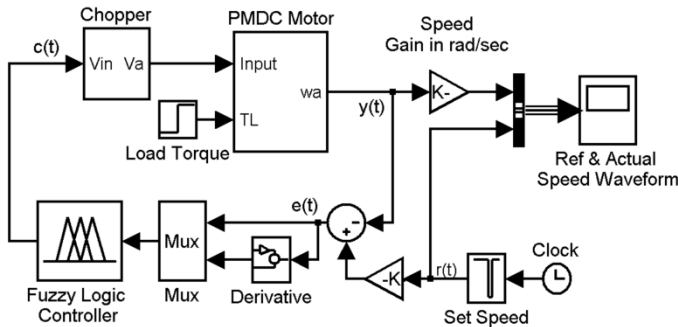
Fig. 5. FLC simulink model

4. PID controller

Convention PID controller system in paper [5] is used to compute the control signal to the PM DC motor for tracking the reference speed as shown in the figure 6. The Proportional-Integral-Derivative (PID) controller used is

$$C(t) = Kpe(t) + KI\int e(t)dt + KD\frac{de(t)}{dt}$$

where K_P is the proportional gain, K_I is the integral gain, and K_D is the derivative gain.
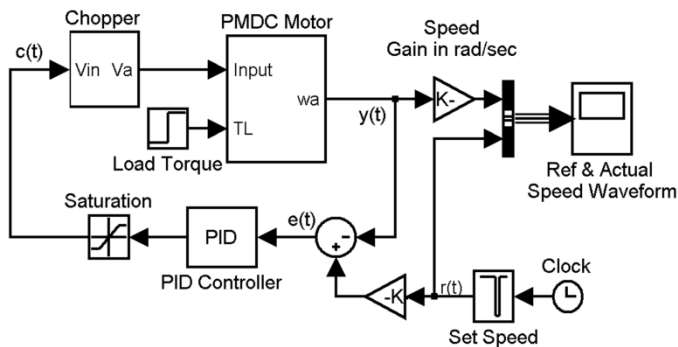
Fig. 6. PID simulink model

5. Simulation Results

To validate the control strategies as explain above, digital simulation were carried out on the PM DC motor model by their governing equations and simulated using Matlab/Simulink. The parameters in the DC motor used are listed in the table 1. A comparative study is carried out to investigate the accuracy of the proposed FLC controller to the classical PID controller by introducing the disturbance in the reference set speed. The disturbance Fig. 7. Comparison of system responses for no load using FLC and PID Controllers is apply of few seconds at each stage and the result were studied.
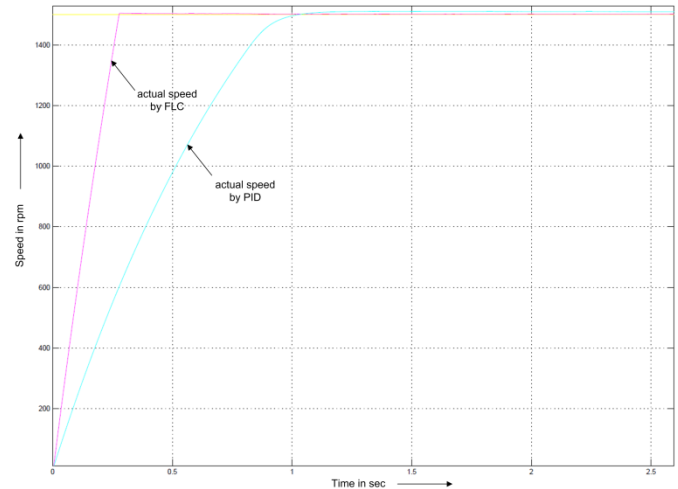
Fig. 7. Comparison of system responses for no load using FLC and PID controllers

From the figure we can observe that the system with FLC controller correctly follows the reference signal.

TABLE I
DC MOTOR PARAMETER

Parameter	Values
Armature resistance (Ra)	2 Ω
Armature inductance (La)	0.05 mH
Moment of inertia (Jm)	1.98 Kg m2
Damping co-efficient (Bm)	0.01 Nm-s/rad
Torque constant (Kt=Km φ)	14.7 Nm/Amp
Back EMF constant (K=Kb φ)	14.7 V-sec /rad
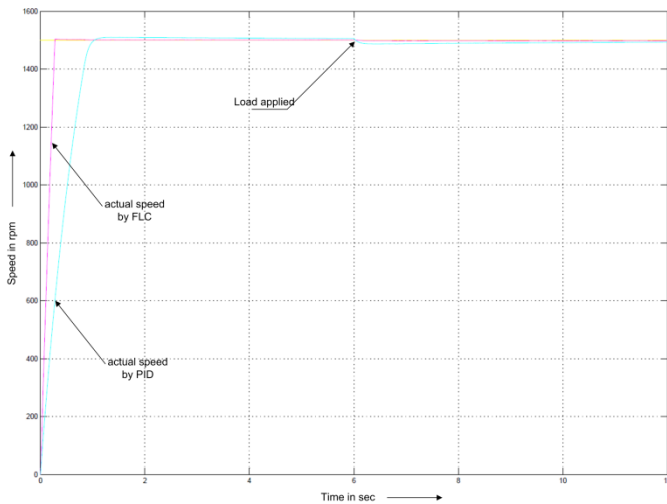
Fig. 8. Comparison of system responses for load impact using FLC and PID controllers

Fig. 11. Comparison of system responses for the continuous disturbance in the control signal using FLC and PID controllers

6. Conclusion

PM DC motor speed control has been performed in Matlab simulink environment. DC motor speed has been controlled with classical PID controller and Fuzzy logic controller (FLC). From the simulation result, it is observed that FLC has clearly better performance for providing Tr (rise time), ess (steady state error) and percentage maximum overshoot (Mp) criteria in comparison with PID controller. FLC also has more sensitive responses against load disturbances in according to classical PID controller.

7. References

[1] K. Ogata, *Modern Control Engineering*, Englewood Cliffs, NJ: Prentice Hall, 1990.

[2] J. Yan, M. Ryan and J. Power, *Using Fuzzy Logic Toward Intelligent System*, NJ: Prentice Hall, 1994.

[3] Y. Tipsuwan, Yodyium, "Fuzzy Logic Microcontroller Implementation for DC motor Speed Control", *IECON'99 Proceedings*, pp. 1271–1276, 1999.

[4] S.Aydemir, S. Sezen, H. M. Ertunc, "Fuzzy Logic Speed Control of DC Motor", *Power Electronics and Motion Control Conference*, pp. 766–771, 2004.

[5] K. Ang, G. Chong, and Y. Li, PID Control System Analysis, Design and Technology, *IEEE Trans.Control System Technology*, vol. 13, pp. 559-576, July 2005.

[6] G. Abbas, N. Abouchi, A. Sani, C. Condemine, Design and Analysis of Fuzzy Logic Based Robust PID controller for PWM-Based Switching Converter, *IEEE International Symposium on Circuits and Systems ISCAS*, pp. 777–780, 2011.

Fig. 9. Comparison of system responses for load impact using FLC and PID controllers zoomed
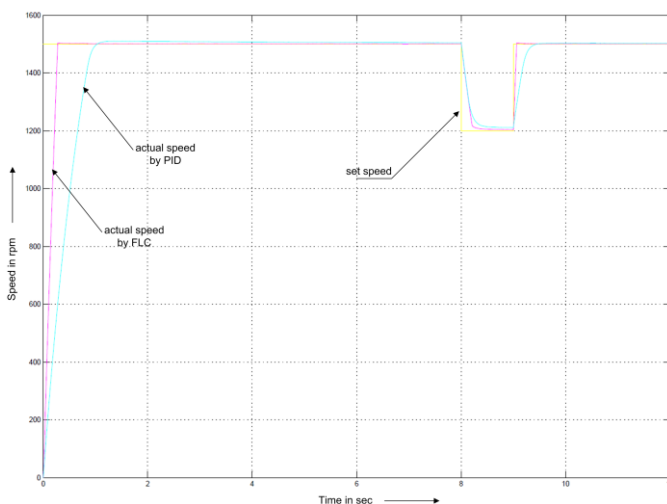
Fig. 10.Comparison of system responses for disturbance in the control signal using FLC and PID controllers

[7] S. Saneifard, N. R. Prasad, H. A. Smolleck, J. J. Wakileh, FuzzyLogic Based Speed Control of a shunt DC Motor, *IEEE Transaction on Education*, vol. 41, no. 2, pp. 159–164, May 1998.

[8] *Fuzzy logic* ToolboxTM *User's Guide*, The MathWorks, Inc.1995–2009.

[9] B. K. Bose, "Expert System, Fuzzy Logic and Neutral Network Application in Power Electronics and Motion", *IEEE Proc.*, vol. 82, no. 8, pp. 1303–1323, Aug 1994.

[10] C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I",*IEEE Trans. Syst. Man, Cybern.*, vol. 20, no. 2, pp.404 - 418, 1990.

[11] C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part II",*IEEE Trans. Syst. Man, Cybern.*, vol. 20, no. 2, pp. 419 - 435, 1990.

A new approach of intelligent control voltage on the low side voltage of distribution transformers based on electrical equivalent model of the system

(Special Session – ICAI'2013: Expert Systems for Intelligent Automation Purposes in Power Systems)

Ernesto A. Mertens Jr.[1], Rogério A. Flauzino[2], Carlos H. da Silveira[1], Ivan N. da Silva[2], Renne T. M. Portal[2], André Bueno[3], Ricardo L. Becker[1], José F. R. da Silva[1]
[1]Elektro Ele. e Serviços S/A, [2]USP/EESC/SEL, [3]RTA

Abstract — **The aim of this paper is to present the problems involved in the control of voltage on the low voltage side of distribution transformers. As proposed for the dealings of this problem is made the presentation of an architecture for the intelligent automatic control of voltage. The goal with this design is to provide grants to set up a system for regulating the voltage on the low side which is technically and economically feasible to be deployed where conventional solutions, with the inclusion of line regulators, are not.**

Key-words — **Voltage regulation, power quality, electricity distribution, intelligent systems, electrical equivalent model.**

I. INTRODUCTION

Voltage regulation in the context of low voltage power distribution is an open problem given that existing solutions to keep the voltage within predefined parameters are often less than optimal as possible. In addition, there is the need for constant refinement on the basis of the regulatory aspects of the Brazilian electricity sector. Thus, it is in this context that we observe a search for alternatives to make the control voltage is able to meet the following requirements:

a)   Maintaining voltage within the range considered adequate;
b)   Be technically and economically feasible;
c)   Do not degrade the quality indices of the supply of electricity in the distribution system in which it operates.

Based on conventional approaches to regulating voltage employed in the distribution system, the aspects quoted above can not be fully met. For conventional approach is cited voltage control for switching transformers tap's and voltage regulators line. Both alternatives perform tension control of the mediun voltage.

Thus, in that paper will be made a statistical analysis of actual data measurement in order to define a complementary approach to conventional techniques in order to overcome Therefore, this paper is organized having in Section II a brief summary of techniques reported in the technical literature and could consubstantiate the formulation of a proposal with the desired characteristics. In Section III regulatory issues regarding voltage power supply in Brazil are highlighted. Section IV presents the results obtained with the statistical processing of actual data and, finally, in Section V conclusions and alternatives for better voltage regulation of distribution systems are the subject of.

II. NON-CONVENTIONAL APPROACHES FOR REGULATING VOLTAGE DISTRIBUTION SYSTEMS

Few tools are proposed technical and scientific literature dealing with the subject of the voltage regulation in the low voltage distribution systems. However, the consensus is the need to seek increasingly innovative techniques to solve the problem. Guided by this aspect of this section is conducted.

Conventional methods of voltage regulation follow routines empirically tested and do not take into account the dynamics of the network load as pointed out in [1]. The increasing integration of distributed generation, just to cite one example, makes the dynamics of the system, and consequently, its tension, is difficult to estimate. Thus, in [1] is used the method of Monte Carlo simulation to estimate the levels of regulation of secondary distribution networks, considering the strong presence of distributed generation. This tool is used for statistical processing of information and is subject to combinatorial explosion when the search space is very large. Thus, the adoption of this type of tool in large system, such as the Brazilian electric system, has an important deterrent.

Many studies consider distributed generation, connected directly to the secondary network as a tool to assist in voltage control. On the other hand, the presence of generators can compromise the voltage control, especially in conditions where such devices do not operate with constant power factor or reactive power flow constant [2]. In [3] considers the impact of distributed generation photovoltaic cells as the quality aspects of the electric power supply.

Thus, it has been distributed generation to an important aspect of the regulatory voltage networks not only primary

but also secondary ones, mainly due to the penetration of such generators. However, it has regulatory terms of the Brazilian electric utilities have to rely on alternative suitable for this task and not rely on disparate sources.

In [4] this issue is reinforced, because it shows that the greatest difficulties involving the regulation voltage in secondary distribution networks with presence of distributed generation are given to the fact of the extreme difficulty in estimating the transgressions in voltage limits observed by customers mainly due to the absence of meters.

Often, the improvement in voltage regulation of the secondary distribution is achieved through actions in primary network, as is the case of [5, 6] using intelligent tools.

Among the research lines of equipment to improve the voltage profile in the secondary networks can highlight the application of inverters connected to the distribution network, as is the case of the proposal presented in [7, 8], where the authors used this device as an active power filter, minimizing the zero sequence currents caused by the imbalance of the network.

They are also used in order to improve the quality of distributed energy consumers the static synchronous compensators or STATCOM. In [9] the authors present a proposal of a STATCOM for distribution that provided good results for the regulation of the secondary network, however this proposal is financially costly due to the power electronics needed to operate the device.

In [10] has a proposal that uses the concept of AC-AC converters that act directly on the secondary distribution network, but only monitor the regulatory limits superior and inferior, not optimally exploiting the range of operation voltage.

III. REGULATORY ISSUES RELATED TO VOLTAGE ELECTRIC POWER SUPPLY

In the ANEEL Distribution Procedures, more specifically in Module 8 are defined the classification criteria the supply voltage. In Fig 1 is present a summary of that classification to the systems with a nominal voltage of 127 V/220 V
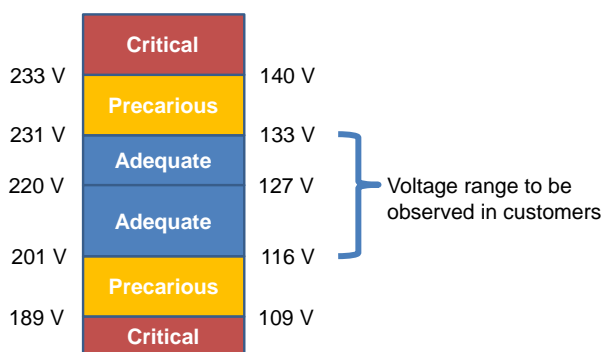
Fig. 1 – Operating ranges for nominal voltage of 127/220 V.

From Figure 1 it can be seen that the range considered suitable for 127 V varies between 116 V and 133 V to 220 V and the corresponding ranges between 201 V and 231 V. Outside the range considered adequate, the concessionaire

shall be monitored in time of transgression, both critical voltage, and for precarious voltage.

Considering the individual indicators for voltage is precarious DRP is the index of relative duration of transgression to precarious voltage, defined as in (1). Already the DRC is the index of relative duration of transgression for critical voltage, defined in (2).

$$DRP = \frac{nlp}{1008} \times 100[\%]$$ (1)

$$DRC = \frac{nlc}{1008} \times 100[\%]$$ (2)

where nlp is the number of readings in the range of precarious voltage and nlc is the number of readings of voltage for the voltage rangecritical. The transgressions are so clearly subject to review in accordance with the temporal ranges transgressed.

The value of the Relative Maximum Duration of Transgression Precarious Voltage - DRPM is set at 3.0%. The value of the Relative Maximum Duration of Transgression Critical Voltage - DRCM is set at 0.5%.

If voltage measurements for complaint and or sample, indicating the amount of DRP DRPM higher than 3%, the distributor should take action to regularize voltage service, within 90 days.

If voltage measurements for complaint and or sample, indicating the amount of DRC higher than 0.5% of the DRCM, the distribution company should take action to regularize voltage service, within 15 days.

Have elapsed the normal deadline for regularization of non-compliance, not been regularized voltage levels in existing deadlines, the distributor must compensate the consumer units that have been subjected to voltages service with transgression of indicators DRP or DRC and those served by the same point connection.

Due to the existing legislation be quite detailed in terms of costs to the dealership if transgressions will be possible to establish criteria that quantify the costs for installation of transformers with intelligent regulation voltage range.

IV. STATISTICAL ANALYSIS OF VOLTAGE DATA

A 8334 records were analyzed for low voltage consumers. The analysis covered the period from 2007 to 2012. These records were analyzed, minimum, and maximum values as a function of the mass of data, a statistical approach was adopted. Figure 2 presents a simplified schematic are represented as a low voltage consumers, as well as the origin of the measurements.

Figure 3 shows that 40% of the measurements were below the minimum voltage of 116V, and 20% of the maximum voltage measurements was above 133V. This result shows how the alternative voltage regulation at low voltage has broadly applicable across the industry distribution. Moreover, not only the problem of undervoltage be attacked but also has the possibility of addressing voltages.

Another review is available in Figure 4, showed that 0.4% of consumers are permanently under voltage, while 0.3% of consumers are in permanent overvoltages.
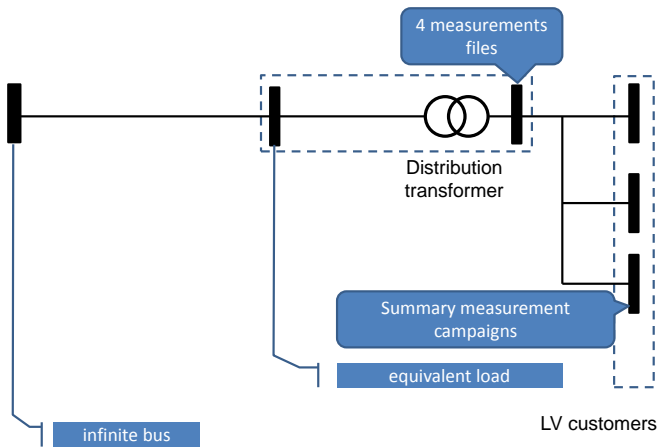

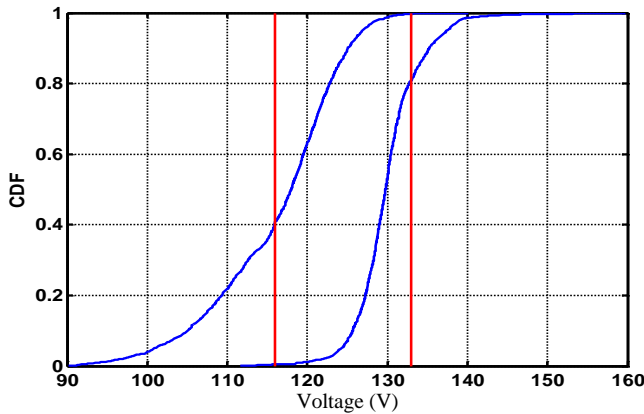**Fig. 2 –** Simplified diagram depicting the origin of measurements.

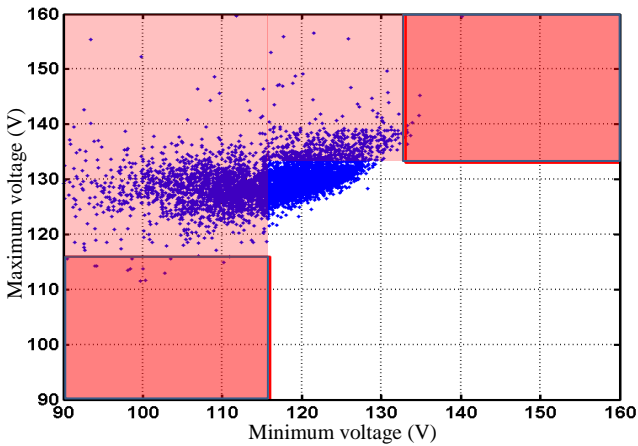**Fig. 3 –** Cumulative probability density function.

Fig. 4 – Evaluation summary of measurement campaigns.

Analyzes were also carried out measurements on distribution transformers with reactive compensation at low voltage. The rated power of the transformer rated power and reactive compensation were as follows:

    a)   75 kVA + 15 kvar

    b)   45 kVA + 7.5 kvar

    c)   45 kVA + 0 kvar

The results may be given in Figures 5 through 9.
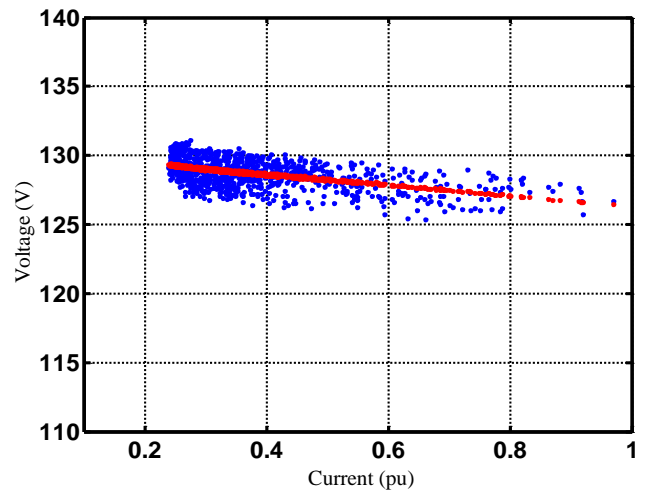
Fig. 5 – Voltage versus current – 75 kVA + 15 kvar.
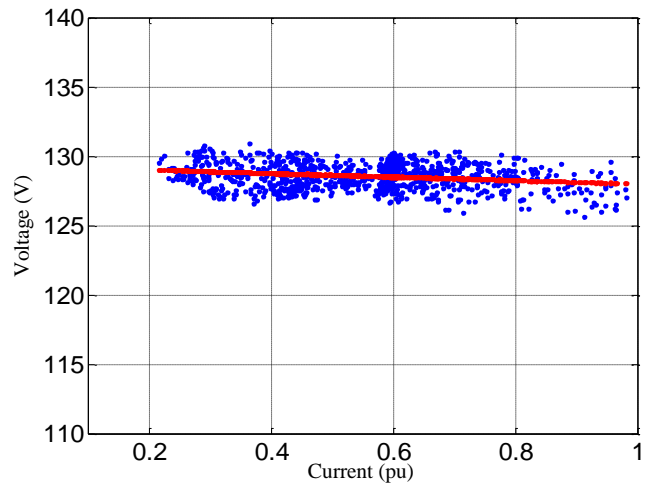
Fig. 6 – Voltage versus current 45 kVA + 7.5 kvar.
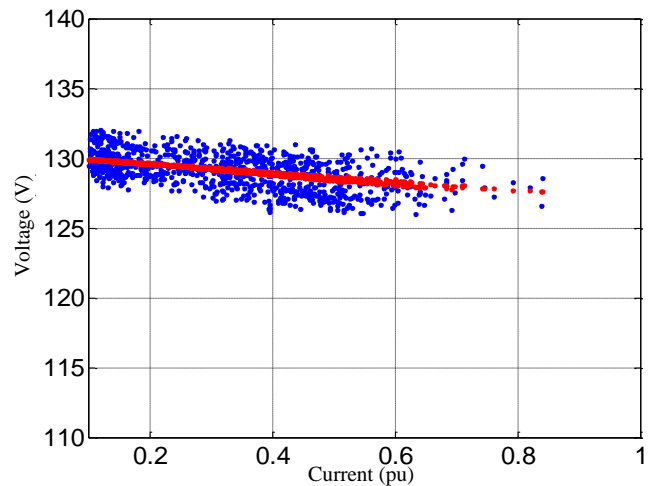
Fig. 7 – Voltage versus current 45 kVA + 0 kvar.

From Figure 9 we can see that for each 1% change in secondary voltage distribution transformer can have 7% of the minimum voltage, 4% of the average voltage and 3% of the maximum voltage.
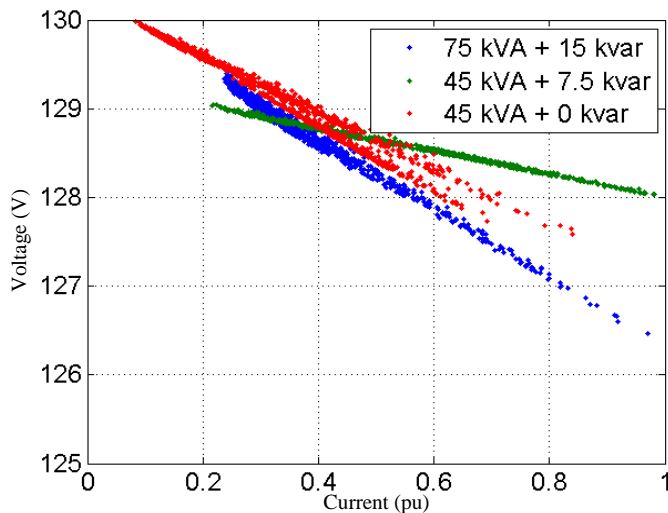
Fig. 8 – Voltage versus current for 3 distributions transformes.
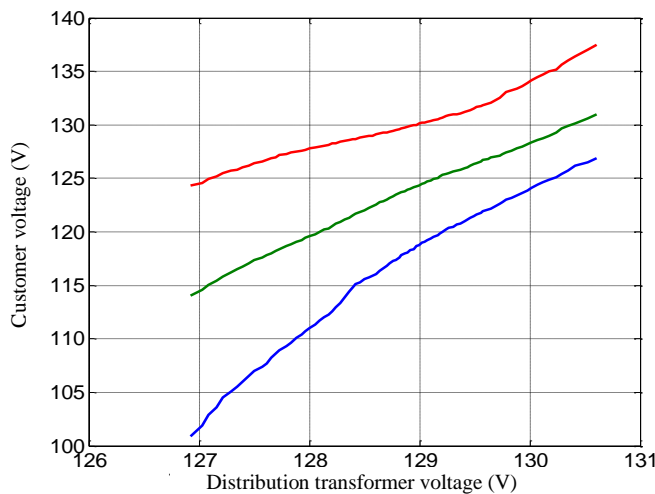
Fig. 9 – Consumers voltage versus distribution transformer voltage.

V. CONCLUSIONS

In the results presented in that paper is possible to contemplate that only a minority of consumers is constantly undervoltage or overvoltage constant. Consumers who own transgression in supply voltage are not in this condition 100% of the time.

This observation allows us to affirm that the alternative solution for these conditions transgression requires monitoring and active control voltage.

Further, it is possible to check by means of Figure 6 to Figure 8 the voltage at the point of common coupling varies, among other quantities, depending on the load current. Thus, an approach to better control voltage range must make use of information not only tension but also current. Thus it will be possible to have an estimate of the parametric distribution system in the direction of the source, as well as making a compensation of voltage drop in the PAC.

Finally, based on the results provided by Figure 9 can make an estimate of the voltage at the load as a function of the voltage on the low voltage distribution transformer.

REFERENCES

[1] McQueen, D.H.O., Hyland, P.R., Watson, S.J., "Application of a Monte Carlo Simulation Method for Predicting Voltage Regulation on Low-Voltage Networks". IEEE Transactions on Power Systems, vol. 20, no. 1, pp. 279-285, 2005.

[2] O'Gorman, R., Redfern, M.A., Al-Nasseri, H., "Voltage control for distribution systems". 2005 International Conference on Future Power Systems, pp. 1-6, 2005.

[3] Conti, S., Greco, A., Messina, N., Raiti, S., "Local Voltage Regulation in LV Distribution Networks with PV Distributed Generation". SPEEDAM 2006 International Symposium on Power Electronics,Electrical Drives, Automation and Motion, pp. 23-28, 2006.

[4] Tsikalakis, A.G., Soultanis, N.L., Hatziargyriou, N.D., "On-line storage management to avoid voltage limit violations". 9th International Conference on Probabilistic Methods Applied to Power Systems, pp. 1-6, 2006.

[5] Provoost, F., Myrzik, J.M.A., Kling, W.L., "Optimizing lv voltage profile by intelligent MV control in Autonomously controlled networks". Proceedings of the 41st International Universities Power Engineering Conference - UPEC '06, pp. 354-358, 2006.

[6] Kulkarni, D. B., Udupi, G.R., "ANN-Based SVC Switching at Distribution Level for Minimal-Injected Harmonics". IEEE Transactions on Power Delivery, vol, 25, no. 3, pp. 1978-1985, 2010.

[7] Belenguer, E., Beltran, H., Aparicio, N., "Distributed generation power inverters as shunt active power filters for losses minimization in the distribution network". 2007 European Conference on Power Electronics and Applications,pp. 1-10, 2007.

[8] Meersman, B., Renders, B., Degroote, L., Vandoorn, T., Vandevelde, L., "Control design of grid-connected three-phase inverters for voltage unbalance correction". Proceedings of the 44th International Universities Power Engineering Conference - UPEC, pp. 1-5, 2009.

[9] Wasiak, I., Mienski, R., Pawelek, R., Gburczyk, P., "Application of DSTATCOM compensators for mitigation of power quality disturbances in low voltage grid with distributed generation". 9th International Conference on Electrical Power Quality and Utilisation - EPQU 2007, pp. 1-6, 2007.

[10] Yamaguchi, K., Kyungsoo, L., Kurokawa, K., " Study on voltage regulation method in the power distribution system". 7th Internatonal Conference on Power Electronics - ICPE '07, pp. 488-492, 2007.

[1] Y. J. Wang, "Analysis of Effects of Three-Phase Voltage Unbalance on Induction Motors with Emphasis on the Angle of the Complex Voltage Unbalanced Factor," *IEEE Trans. on Energy Conversion*, vol. 16, no. 3, pp. 270- 275, 2001.

SESSION

LATE BREAKING PAPERS / POSITION PAPERS: NEURAL NETWORKS, DATABASES, COGNITION AND CLASSIFICATION METHODS, CLUSTERING, INTELLIGENT SYSTEMS, MACHINE LEARNING, AI APPLICATIONS AND ALGORITHMS

Chair(s)

Prof. Hamid Arabnia

A New Method for the Prediction of Carbon Sequestration in Reforested Areas Using a Fuzzy-ART-BP Neural Network

Clodoaldo Nunes[1], Valquíria R. C. Martinho[2], Carlos Roberto Minussi[3]

[1]Infortmatcs Department, Federal Institute of Mato Grosso - IFMT, Cuiabá, MT, Brazil.

[2]Electro-Electronic Department, Federal Institute of Mato Grosso - IFMT, Cuiabá, MT, Brazil.

[3]Electrical Engineering Department, University Estadual Paulista - UNESP, Ilha Solteira, SP, Brazil.

Abstract - *Global emissions of carbon dioxide the main cause of greenhouse effects and, as consequence, global warming, endanger the lives of all living species on the planet. Thus, it becomes essential to adopt measures to reduce carbon emissions, aiming to environmental sustainability and also the development of efficient methods for quantifying the flow of carbon into the atmosphere. Therefore, this paper presents an intelligent system to quantify emissions and carbon sequestration in reforestation areas, medium and long term. The proposed system consists of a combination fuzzy-ART neural network architecture and a multilayer feedforward training based on the backpropagation algorithm. Aiming to test the proposed system, we present an application in an area located in a reforested area in the Amazon region of Mato Grosso-Brazil, on a farm with land area of approximately 8939 hectare.*

Keywords: Estimation of Carbon; Neural Networks; Reforestation Area.

1   Introduction

Over the last years, the increase of the emissions and concentrations of carbon dioxide (CO2) in the atmosphere, due to uncontrolled deforestation, changes in land use and fossil-fuel burning, have contributed to the increase of the greenhouse gases and, consequently, of global warming, fostering climatic changes that threaten life on the planet.

Among the actions to reduce the emission of pollutant gases or Greenhouse Gases (GEE) lies the maintenance of the native forest, the reforestation of the deforested areas or planting of trees in other areas. In order to obtain satisfactory results and economic and environmental benefits associated with the mitigation of these gases and, thus, alleviating the global heating phenomenon, it is necessary to effectively use the potential of the rainforests, as carbon sinks, for the reduction of the GEE [1]] [2].

In this context, it is of great relevance, for the sustainability of the planet and its species, that the flow resulting from the emission and capture of CO2 from the atmosphere, in forest areas, native or reforested, to be methodologically measured and with a significant degree of reliability. This enables to predict even future behavior, inferring on the amount of carbon absorbed by the area analyzed.

Taking into account that the ecological data and the variables involved in the processes of emission and CO2 sequestration (the gases changes) are complex and non-linear, and also, the variability of their most relevant parameters such as trees of different species and ages and, for instance, with distinct growth indexes, type of soil, ambient temperature, luminosity and so many other characteristics and information that influence the variation in levels of exchange of CO2, using artificial intelligence techniques [3], such as the Artificial Neural Networks (RNAs) [4], is a feasible alternative to tackle situations of non-linearity, complexity and variability. Since the RNAs are computational tools inspired in the behavior of the biologial model of the human brain, they process the information and learn through experiences and, thus have considerable capacity of generalization. This characteristic allows for the development of systems capable of dealing with imperfect, absent or noisy data, and still be able to maintain the expected result.

Among the scientific publications on technologies and methods for analysis and quantification of the flow of carbon and of relevance to the purpose of this study, is the modelling method of photosysnthesis at the level of the jatobá (Hymenae Courbaril) leaf [5], using the feedforward neural networks with the retropropagation training technique, or simply, backpropagation (BP) neural network [6], considered to be the precursor work in this area of study. Recently, the relationship between the Carbon Flux (FC) and environmental factors was analyzed [7], using a predictive model based on the Wavelet network [8] and, later, the model was compared to the BP neural network and the support vector machine (SVM) [4], taking into account the network structure, the accuracy and the convergence rate.

This paper presents an intelligent system able to carry out the prediction of the carbon flux in the medium and long-term in reforestation areas reliably inferring what the actual situation is in a certain area analyzed.

This system, of prediction and quantification of carbon was developed using one of the artificial intelligence techniques, the RNAs and, more especifically, a combination formed by a neural network of the ART (Adaptive Resonance Theory) family [9] [10], the Fuzzy-ART [11] and a multilayer feedforward neural network with training performed through the use of retropropagation algorithm [6] [12] (which in this article, for convenience, will be referred to as BP Neural Network. The final intelligent system is designated "Fuzzy-ART-BP" Neural Network. Taking into consideration the proposed experiment, this system shows a satisfactory performance (accuracy and low computational cost) as an interesting alternative method of prediction of the carbon flux in reforested areas.

The vectors, in this paper, are represented by line rather than column notation as commonly adopted in the specialized literature. This is the most apropriate way when neural networks ART-gradient descent are used.

This article is structured in the following way: in Section 2 the definition of the area of study and the construction of the database are described. The development of the system is presented in Section 3. The concepts of the training algorithm BP and of the Fuzzy-ART neural network are briefly and respectively discussed in sections 4 and 5. In Section 6 the implementation of the system is described, the results and the analyses of the tests are presented. Finalizing, the most relevant conclusions are presented in Section 7.

2   Solution Proposed for the Prediction of the Carbon flux

2.1   Definition of the Study Area

The problem to be discussed in this article refers to the prediction of the amount of emissions and carbon sequestration, in reforestation areas medium and long term of a farm located in the Mato-Grossense Amazon, with a territorial extension of nearly 8939 hectare. In that site, for the analysis of carbon, there are areas of native forest, reforestation and an area of the Private Reserve of the Natural Heritage - RPPN (Fig. 1). maximum allowed number of pages is seven for Regular Research Papers (RRP) and Regular Research Reports (RRR); four for Short Research Papers (SRP); and two for Posters (PST).

The interest for this application are the data collected in the reforestation area, through the field forest inventory, which is a document that registers the history of the development of each species, since the date they were planted. For replantion of the deforested area 2 million trees were used, being 49 native species and 2 considered to be exotic, the Teak and the Jambul (Syzygium cumini). The reforested area was divided into small sites of plantation called stands (Fig. 2). These stands are numbered in order to make their

identification, localization and control of the crop development easy. In some stands trees of only one species are planted, in others several species are planted, known as consortium. These data are relevant for the prediction of the amount of carbon in a certain area and/or specific plot or also for the global amount of carbon of the reforested area.

Fig. 1. Heterogeneous areas of the farm.

Fig. 2. Localization and numbering of the stands in the reforested.

2.2   Creating the Database

The amount of carbon in the arboreal area in the reforestation area, described previously, will be predicted by taking into account the time invariant data (parameters of identification of problem: tree species, stand number, etc.) and data whose values will vary as reforestation increases, that is, as the trees grow.

The database consists of information drawn from the field inventory of each stand, from 2003 to 2010. The time invariant data, that make up the database are: number of the stand, identification of the tree, species or common name, characteristics of the soil (phosphorous, pH, texture, amount of aluminium), distance between the stand and the riparian vegetation, spacing among the trees and quantity of species per stand. The time variant characteristics, according to the growth of the arboreal species, are tree height (H) and the Circunference at Breast Height (CBH) [13], that is, trunk circunference measured at a height of 1,3 m from the ground. These measurements were carried out annually.

In each stand there is the possibility of having several species of trees. Aiming to make the provisional system simpler, without significantly compromising the quality of the solutions, we will adopt a procedure taking

into account only one tree representative of the set of trees which, for identification purposes in this paper, will be called "Standard Tree (ST)". The parameter DAPm of the AP is defined as follows:

$$DAPm = \sum_{k=1}^{N} m_k \, DAP_k \qquad (1)$$

DBH: Diameter at breast height
DBHm : DBH of the standard tree
DSTk : DBH of the k-th species

$$DAP_k = \frac{\sum CAP_k}{\pi.n_k} \qquad (2)$$

$$m_k = \frac{n_k}{\sum_{j=1}^{N} n_j} \qquad (3)$$

This strategy, based on the weighted average of the DBH, has provided good results. However, other forms of ST definition can be employed to further improve the results obtained. For example, the use of the Fuzzy logic [14], Dempster-Shafer theory of evidence [15], among others.

3   Fuzzy-ART-BP Neural Network

The neural network to be used (Fig. 3) is composed of 2 modules: (1) Fuzzy-ART; (2) multilayer feedforward neural network with training performed using the BP algorithm (BP Neural Network). This structure is adopted, taking into account that the problem to be solved (estimate of carbon absorption in reforested areas) is designed from two groups of data, that is, binary data (greater amount of information) and data belonging to the body of real numbers. These two groups are defined as follows:

a = [nt ia nc cs dt ea qe]   (4)

b = [d   DAPm(t)]   (5)

being: t   : discrete time;

In Table I, the time-invariant quantities are specified.

The subvectors nt, ia, . . ., and qe of the vector a are line vectors that contain the binary representation of the time- invariant quantities. Each bit corresponds to a component of the associated vector.

Vector d contains information about the evolution of the growth pattern of ST, i.e., they are the values of the DAPm measured in three different time intervals: (t-3), (t-2) and (t-1). Vector b is constituted by vector d added to the value DAPm(t) which corresponds to the output feedback y, considering a delay equal to t (time unit). The output corresponds to the prediction in time interval ahead. For example, the time to be considered

coud be monthly, or another reference chosen by the user.

This proposal is to shift a temporal window, housing, for example, four sequential instances (past data and current time). Other strategies, that aim to improve the performance of the forecast system, can be perfectly implemented.

The vector a represents the input of the Fuzzy-ART unsupervised neural network module. The output of this module are the self-organized classes by the Fuzzy-ART module. The chosen class (Ω) (output of this module) is converted into a binary vector (ψ) through the processing of the module "active-code". The binary vector ψ, along with the input b (5), form the input vector x of the BP neural network. Thus, the input-output model of the BP neural network is described by:

x = [b ψ] (input of the BP neural network)   (6)

y = [DBHm(t+1)] (output of the BP)   (7)

It is observed that in this proposal, the output of the Fuzzy-ART-BP neural network has only 1 component which is the DBHm(t+1). However, there is the possibility of using different information aiming to meet our needs, if there is interest.
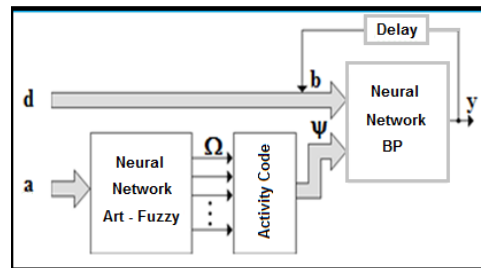

Fig. 3. Fuzzy-ART-BP neural system.

TABLE I.   TIME-INVARIANT QUANTITIES

nt	ia	nc	cs	Dt	ea	qe
Number of the stand	Identification of the tree	Species	Characterístics of the soil	Distance between trees	Spacing between trees	Quantity of Species
8 bits	5 bits	5 bits	6 bits	2 bits	5 bits	4 bits

4   BP Neural Network

The training, by use of backpropagation algorithm, is initiated by presenting a pattern x to the neural network, which will produce an output y. Then, the error of each output is calculated (difference between the desired value and the output). The next step is to determine the error propagation in the reverse direction, through the network associated with parcial derivative of the mean squared error of each element in relation to the weights and, finally, adjusting the weights of each element. A new pattern is presented. So, the process is

repeated, for all the patterns, until total convergence occurs (|error| ≤ pre-established tolerance).

The initial weights are usually adopted as random numbers. The BP algorithm consists of the adaptation of weights for the purpose of minimizing the quadratic error of the neural network. The sum of the instantaneous squared error of each neuron allocated in the last layer (output of the neural network) is given by [16]:

$$\varepsilon^2 = \sum_{i=1}^{ns} \varepsilon_i^2 \qquad (8)$$

being:
$\varepsilon i = di - yi$;
di : output desired of the i-th element of the last layer of the network;
yi : output of the i-th element of the last layer of the network;
ns : number of neurons of the last layer of the network. Taking into account the neuron of index i of the neural network, and using the method of the gradient descent [12], the adjustment of weights can be formulated as [4], [12]:

Vi (h+1) = Vi (h) + θi (h)   (9)
being:
$\boldsymbol{\theta}_i (h) = - \gamma [\boldsymbol{\nabla}_i (h)]$ (10)

γ : stability control parameter or training rate;
h : it represents the iteration index;

∇i (h): the gradient of the squared error in relation to the weights of the neuron i evaluated in em h;

Vi : vector containing the weights of the neurônio i;
where: Vi = [w0i w1i w2i ... wni].

In (9), the direction adopted, to minimize the objective function of the squared error corresponds to the opposite direction to the gradient-vector. The parameter γ determines the length of the vector [θi (h)].

The sigmoid function is defined by [4], [12]:

$$y_i = \frac{1 - \exp(-\lambda s_i)}{1 + \exp(-\lambda s_i)} \qquad (11)$$

or

$$y_i = \frac{1}{1 + \exp(-\lambda s_i)} \qquad (12)$$

where:
λ : constant which determines the slope of the curve yi. It is noteworthy that the range of variation of the sigmoid function yi given by (11) or (12) is [−1, +1] or [0, +1], respectively.

The BP algorithm is addressed in the literature in several ways with the purpose of making it computationally faster. A quite interesting formulation is

the backpropagation algorithm through time [12], which has an stabilizing effect:

Vij(h+1) = Vij(h) + ΔVij(h)   (13)
being:

ΔVij(h) = 2 γ (1 − η) δi(h) Xj + η ΔVij(h − 1)   (14)

Vij : synaptic weight of interconnection among the neurons i e j;

η : time constant, $0 \le \eta \le 1$;

δj : derivative of the square error (gradient);

x : input vector.

The training modes can be of two types, the so-called sequential and batch. The sequential is the one in which the adjustment of the weights is done after the presentation of each pattern and the batch with all the patterns at the same time [12].

So, by performing the calculation of the gradient as shown in (10), considering the sigmoid function defined by (11) or (12) and the term time, the following scheme of weight adaptation is obtained [12]:

Vij (h+1) = Vij (h) + ΔVij (h)   (15)
being:
ΔVij (h) = 2 γ (1 − η) βj xi + η ΔVij (h − 1) (16)
Vij : weight corresponding to the interconnection between the i-th and the j-th neuron.

If the element j is found in the last layer, then:

βj = σj εj   (17)

where:

σj : derivative of the sigmoid function given by (11) or (12), respectively, in relation to sj:

σj = 0,5 λ (1 − yj2)   (18)

σj = λ yj (1 − yj)   (19)

If the element j is found in the other layers, then:

$$\beta_j = \sigma_j \sum_{k \in R(j)} w_{jk} \beta_k \qquad (20)$$

being:

R (j) : the set of the indexes of the elements that are in the row following the row of the element j and that are interconnected to the element j.

The parameter γ serves as an agent to control the stability of the iterative process.

The network weights are initialized randomly considering the interval [0,1] [12].

5   Fuzzy-ART Neural Network

The ART neural network is formed by three layers: F0 (input layer); F1 (comparison layer); F2 (recognition layer) which stores the categories (clusters). The algorithm of this neural network consists basically of the followings steps [11]:

Step1:   Input data

The input data are denoted by vetor $a = [a_1 \quad a_2 \quad a_3 \quad ... \quad a_M]$. This vector must be as follows:

$$\bar{a} = \frac{a}{|a|} \quad , \text{ sendo}: \quad |a| = \sum_i^M a_i \qquad (21)$$

being:

$\bar{a}$: normalized input vector;

$|a|$: norm of the input vector a;

a_i : element of the input vector a with an index i.

Step 2:   Encoding of the input vector

The complement encoding is performed to preserve the scope of the information, that is:

$$\bar{a}_i^c = 1 - \bar{a}_i \qquad (22)$$

$\bar{a}^c$: complementary normalized input vector.

Thus, the input vector can be written:

$$I = \begin{bmatrix} \bar{a} & \bar{a}^c \end{bmatrix} \qquad (23)$$

$$I = \begin{bmatrix} \bar{a}_1 & \bar{a}_2 ... \bar{a}_M & \bar{a}_1^c & \bar{a}_2^c & \bar{a}_M^c \end{bmatrix}$$

being:

$$|I| = \sum_{i=1}^M \bar{a}_i + \sum_{i=1}^M \bar{a}_i^c \qquad (24)$$

=   M (all vectors with normalization and encoding complemented will have same length M).

Step 3:   Vector Activity

The vector of F$_2$ activity is simbolized by $\Omega = [\Omega_1 \, \Omega_2 \, ... \, \Omega_N \,]$, being N the number of categories created in F$_2$. Thus, we have:

$$\Omega_j = \begin{cases} 1, & \text{If node } j \text{ de F}_2 \text{ is active} \\ 0, & \text{if not} \end{cases}$$

Step 4:   Parameters of the neural network

The parameters used in the processing of the *Fuzzy-ART Network* are:

Parameter of choice : $\alpha > 0$;

Parameter of training   : $\beta \in [\, 0, 1\,]$;
Parameter of monitoring : $\rho \in [\, 0, 1\,]$.

Step 5: Inicialization of weights

Initially all weights have values equal to 1, that is:

$$w_{j1}(0) = w_{j2M}(0) = ... = w_{NM}(0) = 1 \qquad (25)$$

Indicating that there is not any active category.

Step 6:   Choice of the category

Given the input vector I in F$_1$, for each knot j in F$_2$, the choice function of T_j is determined by:

$$T_j(I) = \frac{|I \wedge w_j|}{\alpha + |w_j|} \qquad (26)$$

being:

$\wedge$: Fuzzy AND operator, defined by:

$$(I \wedge w)_i = \min(I_i , w_i)$$

The chosen category associated with the knot J active, which is:

$$J = \arg \max_{j=1,...,N} T_j \qquad (27)$$

Using (27), if there is more than one active category, the chosen category will be that of a lower index.

Step 7:   Resonance or Reset

The ressonance occurs if the vigilance criterion (28) is achieved:

$$\frac{|I \wedge w_j|}{|I|} \geq \rho \qquad (28)$$

If the criterion defined in (28) is not achieved, the device called reset occurs. At reset, the knot J of F2 is excluded from the search process, that is, adopting TJ = 0. Then, a new category is chosen through (27) for the resonance process. This procedure will be performed until the network finds a category that satisfies eq. (28).

Step 8: Updating of the weights (Training)

After the input vector I has completed the state of resonance, the training process follows, in which the modification of the vector weight occurs given by:

$$w_J^{new} = \beta (I \wedge w_J^{old}) + (1-\beta) w_J^{old} \qquad (29)$$

being:

J   : active category;

w_J^{new} : weight vector updated,

w_J^{old} : weight vector referred to the previous updating.
If β = 1, thus, we have the fast training.

6 Application and Analysis of Results

The application of the intelligent system, "Fuzzy-ART - BP" Neural Network, proposed in order to perform the prediction of the carbon flux, used a database composed by 8.213 lines and 38 columns. Each row corresponds to a tree analyzed and represents the pattern input vectors and their desired output. The data in columns from 1 to 35 represent the attributes corresponding to the vector a (input of Fuzzy-ART neural network), binary representation of the time invariant quantities. In columns 36, 37 and 38 the atributes are time variants comprising the pairs of input pattern vectors d (and, hence, the constitution of vector b) and that of the output y, as indicated in Fig. 3. The rows of the database were subdivided for the specific analysis of each stand represented by them.

After the subdivision of the database, using the Fuzzy-ART Neural Network, module 1 of the system, and considering the time-invariant quantities of each tree of the stand, it was found in the output of this module, the class representative of each stand (Ω). The classes are self-organized by the Fuzzy-ART module, as it is a non-supervised neural network. Then, through the processing of the "active-code" module, the class (Ω) which represents the stand was converted into a binary vector (ψ). A demonstration of this phase of the process can be seen in Fig. 4. The parameters used in the training phase of the ART-Fuzzy network are specified in Table II.

The technique of windowing to find the DAPm of the standard tree of each stand in analysis was implemented using module 2 of the system – BP Neural Network. In this experiment, for the implementation of the windowing, we adopted yearly scheduling, from January to December, from 2003 to 2014 and from 2003 to 2010 the value of DAPm was known.

In this technique, we considered the time-variant quantities representing the growth of the standard tree of the stand under analysis, and for two successive years prior to the year that the prediction was desired.

The result of the windowing process along with the binary vector (ψ) representative class of the stand, constituted the input vector of the BP network. At the output of this module, it was obtained in the form of DAPm, a prediction of carbon absorption in the stands analyzed for a time interval forward. This phase of the process is shown in Fig, 5.
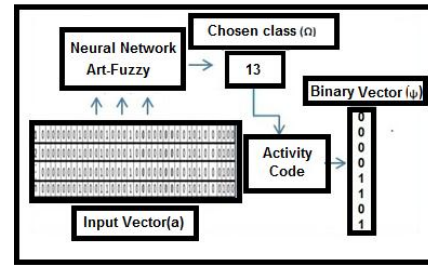

Fig. 4. Processing of the Fuzzy-ART neural network.

TABLE II.   PARAMETERS OF THE FUZZY-ART.NETWORK

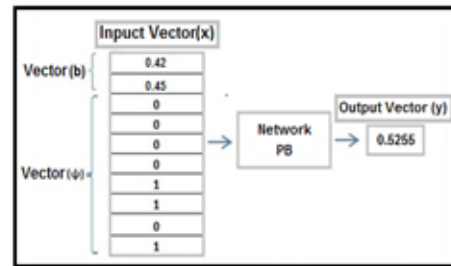| Parameters	Values
Vigilance parameter (ρ)	0.99
Training rate(β)	1.0
Choice parameter (α)	0.1

Fig. 5. Processing of the BP Neural Network.

After the process of windowing and training of the BP neural network, the DAPm forecast was carried out and the results of the Fuzzy-ART-BP Neural Network, a proposed system in this experiment for predicting carbon absorption, are shown in Table III and in the graph of Fig. 6.

Considering the experiment and analyzing the results obtained, it was observed that the absorption of carbon found in the simulations was checked with those already known and has shown a significant degree of reliability, and accurately enabling the knowledge of the actual situation in the area being analyzed.

In Fig. 6 it is clearly observed the aproximations of the results between the values of DAPm, that is, the target values and the DAPm values of the output of the proposed network, having a mean error between them of approximately 0.02.
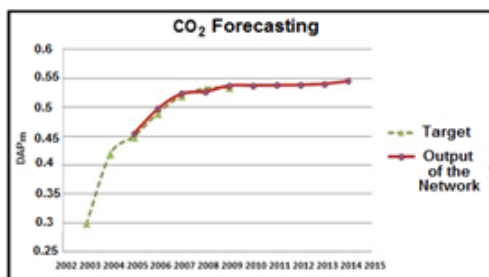
Acknowledgments

Fig. 5. Predictin of carbon.

TABLE III. PREDICTION OF CARBON ABSORPTION.

Year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
Target	8.10	11.34	12.15	13.23	14.04	14.39	14.43	-	-	-	-	-
Output Network	-	-	-	-	14.15	14.25	14.53	14.54	14.55	14.56	14.61	14.74

7   Conclusions

This work presents a new method for the prediction of carbon sequestration in the medium and long-term in reforested areas. This method uses Fuzzy-ART-BP Neural Network composed by two modules: (1) Fuzzy-ART; (2) Multilayer feedforward neural network with training performed using the BP algorithm.

The analysis of the results obtained demonstrated that the proposed system is satisfactory (error inferior to 0,2%), that is, the absorption of carbon can be methodologically measured and with a significant degree of reliability, making the inference of estimates of carbon absorption possible.

The results show that the Fuzzy-ART-BP Neural Network is very efficient for the prediction of carbon absorption, which suggests a strong potential for its implementation for the analysis of the flux resulting from the capture and sequestration of CO2, in forests reforested, thus contributing to the adoption of measures that contribute to the sustainability of the planet.

8   References

[1]   A. LeBlanc, "Issues related to including forestry-based offsets in a GHG emissions trading system," Environmental Science & Policy, vol. 2, pp. 199-206, 1999.

[2]   C. H. R. T. Silva, "COP16 - Voluntary Goals and Reform of the Forestry Code : deforestation in Brazil and mitigation of global climate change," Study Centre Consulting's Federal Senate," ed, 2010.(in Portuguese).

[3]   E. A. Bender, Mathematical methods in artificial intelligence. Los Alamitos, Calif. :: IEEE Computer Society Press, 1996.

[4]   S. S. Haykin, Neural Networks and Learning Machines: Prentice Hall, 2009.

[5]   M. L. B. Barriga Puente de La Vega, "Phytomonitoring and modeling of photosynthesis jatoba (Hymenaea. L.) with artificial neural networks" (Doctoral Thesis), Graduate Program in Electrical Engineering, Polytechnic School of the University of São Paulo, São Paulo, 2003.(in Portuguese).

[6]   P. Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences," Harvard University, 1974.

[7]   K. Wang, X. Yue-ju, J. King, H.-m. Chen, and C. Qiang, "Prediction of carbon flux based on wavelet networks," pp. 1553-1556, 2011.

[8]   I. Daubechies, Ten Lectures on Wavelets: Society for Industrial and Applied Mathematics, 1992.

[9]   G. A. Carpenter and S. Grossberg, "A self-organizing neural network for supervised learning, recognition, and prediction," Comm. Mag., vol. 30, pp. 38-49, 1992.

[10]   S. C. Marchiori, M. d. C. G. Silveira, A. D. P. Lotufo, C. R. Minussi, and M. L. M. Lopes, "Neural network based on adaptive resonance theory with continuous training for multi-configuration transient stability analysis of electric power systems," Appl. Soft Comput., vol. 11, pp. 706-715, 2011.

[11]   G. A. Carpenter, S. Grossberg, and D. B. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system," Neural Netw., vol. 4, pp. 759-771, 1991.

[12]   B. Widrow and M. A. Lehr, "30 years of adaptive neural networks: Perceptron, Madaline, and backpropagation," Proceedings of the IEEE, vol. 78, pp. 1415-1442, 1990.

[13]   C. R. Sanquetta, L. F. Watzlawick, R. Balbinot, M. A. B. Ziliotto, and F. S. Gomes, As florestas e o carbono. Curitiba, 2002.

[14]   L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338-353, 1965.

[15]   L. A. Klein, Sensor and Data Fusion: A Tool for Information Assessment and Decision Making: SPIE Press, 2004.

[16]   M. L. M. Lopes, C. R. Minussi, and A. D. P. Lotufo, "Electric load forecasting using a Fuzzy ART & ARTMAP neural network," Applied Soft Computing, vol. 5, pp. 235-244, 2005.

Simulation Results for a Cache Management System used in a Deductive Database

Larry Williams[1], Martin Maskarinec[1], and Kathleen Neumann[1]
[1] School of Computer Sciences, Western Illinois University, Macomb, IL

Abstract - *This paper will present the effectiveness of a cache management system which was previously developed for a Deductive Database. In order to do this, a simulation of user inputted queries was developed and then executed on two different input sets. This paper presents the results of these simulation runs and a comparison between the results of our caching algorithm as compared to two other, standard approaches.*

Keywords: Intelligent Database, Deductive Database, Cache Algorithms

1. Introduction

A deductive rule uses predicates to represent knowledge that may be derived from known facts. For example, we may write "P(X,Z) :- F1(X,Y), F2(Y,Z)". This rule indicates that the predicate "P" is dependent on the facts "F1" and "F2". These predicates and facts may have arguments; the variables X, Y, and Z indicate how the arguments of the resulting predicate P are derived from the predicates of F1 and F2 and any other constraints (in this example, the second argument of F1 must match the first argument of F2).

A deductive database is a collection of these rules. These rules may contain many facts and/or other predicates. As the set of rules expands, it is convenient to view it as a "Predicate Connection Graph", such as the one depicted in Figure 1. This particular Connection Graph shows a rather large number of predicates and facts; a user may query any node in the Graph at any time. See [2] for a complete discussion of how queries are processed in such a system.

As more and more queries are presented to the deductive database on larger and larger predicate results, it becomes important to cache previously realized results so, if they are queried again, the results are readily available and do not need to be recalculated. As with all caching algorithms, it is unreasonable to assume all previously realized results may be saved in memory – there is just too much data for too many nodes. Thus, the important question to answer here is: when the maximum amount of available memory is exceeded, what should be saved to the cache and what should be removed and left to be recalculated when queried?

There are several caching algorithms available for relational databases (see [3] for an example). However, these do not take into account the dependencies of predicate caches. As can be seen in Figure 1, removing the cache for node P2 has two major ramifications: 1) most obviously, nodes down the tree would need to be referenced to recreate the result for P2 if it is queried again, and 2) nodes up the tree would also become more expensive to recalculate, since they reference P2 and the cost of acquiring the result for P2 will go up if its cache is removed.

Previously, we created a caching algorithm to take into account these special requirements of a deductive database [1]. In this paper, we present our results of testing this algorithm to determine just how effective it is.

2. Review of Caching Algorithm

When a query request comes in to the Deductive Database, the system will first check to see if it is cached. If so, a hit counter will be incremented and the results returned. If not, the results will be calculated by querying the children nodes for their results and then putting these results together as required by the predicate rule. Once this result is compiled, it needs to be determined if it should be cached or not. This is done by determining the node's "Recalculation Cost".

The recalculation cost of a node is a function of the cost of producing the result of the node, the number of parents, and the miss percentage. Briefly, the cost of producing the result of the node looks "down" the predicate graph, while the number of parents looks "up" the predicate graph. This miss percentage is also factored in since a large number of misses would indicate a frequently queried node that is not currently being cached. See [1] for a complete description of the recalculation cost methodology.
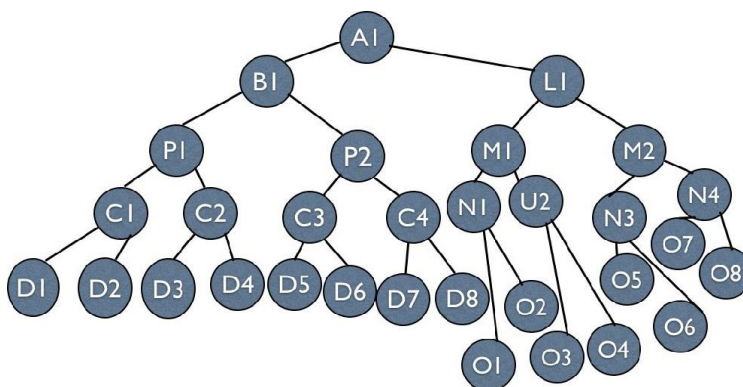
Figure 1. A Sample Predicate Graph

3 Simulation

The simulation compares three different types of cache algorithms: First In, First Out; Last In First Out; and the Recalculation Cost algorithm described in [1]. For each of the three algorithms, the same predicate graph will be read from a file. This file will contain the list of predicates and their connections; also, simulated size information will be provided as well. Then, a set of 100 simulated queries will be put to each algorithm to simulate user input. These queries will also be read from a file to ensure all algorithms are given exactly the same input.

We have created two sets of 100 simulated queries. The first set is one that contains relatively few repeated queries (we refer to this one as the "low performance" set because the cache will not be hit very often in any case). The second set contains relatively many repeated queries (we refer to this one as the "high performance" set because the cache algorithm should enable many hits). We will then measure system performance for both of these sets in terms of hit percentage as well as number of bytes retrieved from disk (for cache misses).

Next, we describe the three algorithms in more detail.

3.1 First In, First Out

In the First In, First Out algorithm, the system will cache all the nodes until the cache is full; once the cache is full the cache management system will remove the node that was placed in the cache first. This algorithm acts basically like a queue.

3.2 Last In, First Out

In the Last In, First Out algorithm, the system will cache all the nodes until the cache is full; once the cache is full, the system will remove the last node that was entered into the cache. This algorithm acts basically like a stack.

3.3 Recalculation Cost Algorithm

In the Recalculation Cost Algorithm, the system will cache the nodes based on the algorithm described in [1].

4. Results

Figure 2 shows the hit rate for all cases. As can be easily seen, the FIFO algorithm running on the high performance set trends much worse than the other two as more and more hits occur. The LIFO algorithm and the recalculation algorithm trend very similarly with the recalculation algorithm slightly out-performing LIFO by the end of the simulation. On the low performance set, all three trend similarly.
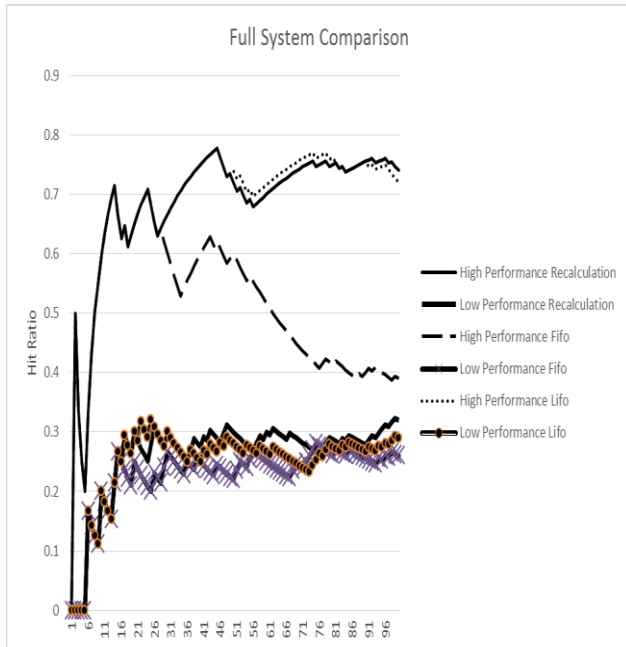
Figure 2. Hit rate all cases.

Figure 3. Bytes processed all cases.

At first glance, this seems to indicate the recalculation algorithm is not much better than the LIFO algorithm. However, it must be remembered that Figure 2 shows hit rate and this is not what the recalculation algorithm is designed to optimize. A large number of hits on small and/or easily recalculated predicates may increase the hit percentage, but caching larger and/or more expensive predicates to recalculate would save time.

To demonstrate this, Figure 3 presents the same simulation runs, but looking at total bytes read from disk, rather than a simple hit rate. Thus, each time a miss occurs, the number of bytes necessary to be retrieved to construct the result is calculated and added to a running sum. The final values for this running sum are shown in Figure 3 for all three algorithms run on both input data sets.

As can be seen in Figure 3, the recalculation method clearly outperforms the other two for both input data sets. In fact, it requires only about 60% of the disk accesses of the other two cases in both input data sets.
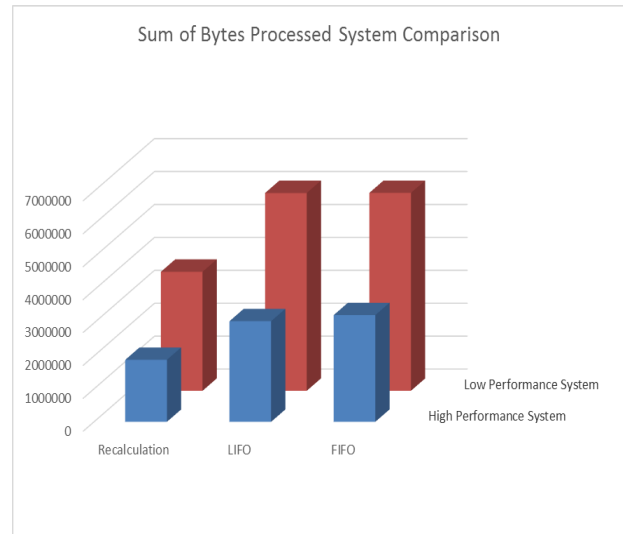
5. Conclusion

A simulation of a cache management system has presented which demonstrates this system outperforms basic FIFO and LIFO techniques.   Using our recalculation cost method, 50% less disk accesses are required to meet the simulated user's requests.

6. Bibliography

[1] Williams, Maskarinec, Neumann, "A Cache Management System for a Distributed Deductive Database", the International Conference on Artificial Intelligence (IC-AI 2012), Las Vegas, NV, July 2012.

[2] Maskarinec, Neumann, "Heterogeneous Cost Functions and a Partitioning Algorithm for a Distributed Deductive Database", the International Conference on Artificial Intelligence (IC-AI 2003), Las Vegas, NV, June, 2003.

[3] Altinel, Mehmet, Christof Bornhovd, Sailesh Krishnamurthy, C. Mohan, Hamid Pirahesh, and Berthold Reinwald. "Cache Tables: Paving The Way For an Adaptive Database Cache." VLDB Conference. Berlin, Germany, 2003. Print.

The Cognitive Architecture of the Human Mind

Christian C. Wagner[1], and James G. Schiiler[1]

[1]Industrial and Systems Engineering, Oakland University, Rochester, MI, U.S.A.

Abstract - *The Cognitive Architecture is the initial stage in building a new type of cognitive system based on human memory called The Cognitive Database (CDB). In human cognition, human memory is the central component of human intelligence. In the computer model of cognition, all of human memory is labeled as Total Memory (TM). The basic structural element of TM is a memory node (MemNo). MemNos are defined along with a set of operations intrinsic to the creation, retrieval, update, and deletion of MemNos. In addition to covering memory, The Cognitive Architecture covers these areas: major components of a cognitive system, recursion, automaticity of association, control routine, abstraction rules, and distance measure to stop the recursion.*

Keywords: artificial intelligence; psychology; databases

1 Introduction

The Cognitive Architecture is a blueprint for implementing The Cognitive Database (CDB). A CDB is a new type of cognitive system based on human cognition. The word *cognitive* comes from psychology, referencing the mental processes of the mind. The word *database* comes from computer science and is the general term for a collection of data. The creation of any intelligence, whether human, humanlike, or artificial, is highly desirable. Our interests are *not* focused on building a human, but on modeling human cognition at a higher level. Understand how difficult and complex of a task this is. We are trying to architect a system that can invent set theory, a system that can move from set theory to fuzzy set theory. We must accommodate the theories of medicine, economics, artificial intelligence, manufacturing and every other field of human endeavor. The first and foremost part of the system that needs to be identified is the basic element of cognition. From there, the basic elements can be combined into ever higher level structures.

2 The basic element of cognition

The basic element of cognition is a memory node (MemNo). The parts of a MemNo are themselves other MemNos. With this recursion, the information represented by a MemNo may be anything from the most basic physiological state (e.g., "I am thirsty") to a complex and deeply recursive concept (e.g., the very question of asking the brain how the brain functions).

2.1 Memory Nodes

MemNos are of many types, as all conscious aspects of human existence are represented therein. There are five types of MemNos: Sensory, Motor, Cognitive, Object and Relationship. (PST).

- Sensory MemNos are the summation of a collection of sensory experiences into a concept (sight, hearing, touch, taste, smell, balance, thirst, hunger, and so on; for example, input from the eye recognized as the concept "blue").

- Motor MemNos are the summation of a collection of commands that drive the physical effectors of the human experience into a concept (muscle movement in all its forms; for example, output of moving the hand back and forth recognized as "wave").

- Cognitive MemNos are the summation of a collection of cognitive tasks into a concept (decide, choose, think, remember, and so on; for example, the cognitive task of looking through a set of alternatives and selecting one recognized as "choose").

- Object MemNos are the summation of an unordered collection of MemNos into a concept (generalization and abstraction of objects: for example, the unordered concepts of "furry-1," "tail-3," and so on are recognized as defining the concept "dog-2").

- Relationship MemNos are the summation of an ordered collection of MemNos into a concept (relationships that are verbal, prepositional, and so forth; for example, the concepts of MemNos "drink-4," "man-2," "shake-2" are recognized as the concept "man-drinks-shake").

2.2 Memory Node operations

A set of operations are intrinsic to the operation of MemNos. As one might expect from ordinary database operations, the four CRUD operations are critical to MemNos: create, retrieve, update, and delete operations. The precise mechanism for the operations of MemNos will be described in another paper.

As seen in the functional notation on the following page, when any of the operations are applied to a set of MemNos (M_i), the result will be a new MemNo.

$$M_{new} = \text{operation}(M_a, M_b, \ldots) \qquad (1)$$

The arguments are MemNos and the operations result in new MemNos. This property of MemNo operations is important because the operations that compare words can compare higher-level elements, such as concepts, through recursion. For example, the type of process that recognizes a wheel on a car can, through recursion, recognize the hood of the car, and recognizes the car itself. Only the content of the process changes, not the process itself.

2.3   Total Memory

In the computer model of human cognition, human memory (labeled as Total Memory, TM) is the central component of human intelligence. TM is composed of all memory subsystems including: short term memory (STM, our working memory), episodic memory (EM, our life history), and abstracted memory (AM, our mental models).

$$TM = STM \times EM \times AM \qquad (2)$$

STM is the portion of human memory that models the current moment of experience. And, as that current moment passes, some partial copy is transferred into EM. As processing time permits, the experiences of EM are generalized into AM. Note that all parts of TM hold nothing more than MemNos − the recursive structural element of human cognition.

3   The Cognitive Architecture

Observe the diagram of The Cognitive Architecture in Fig. 1. The diagram shows the major components of a CDB. The architectural diagram and description helps implement the CDB.
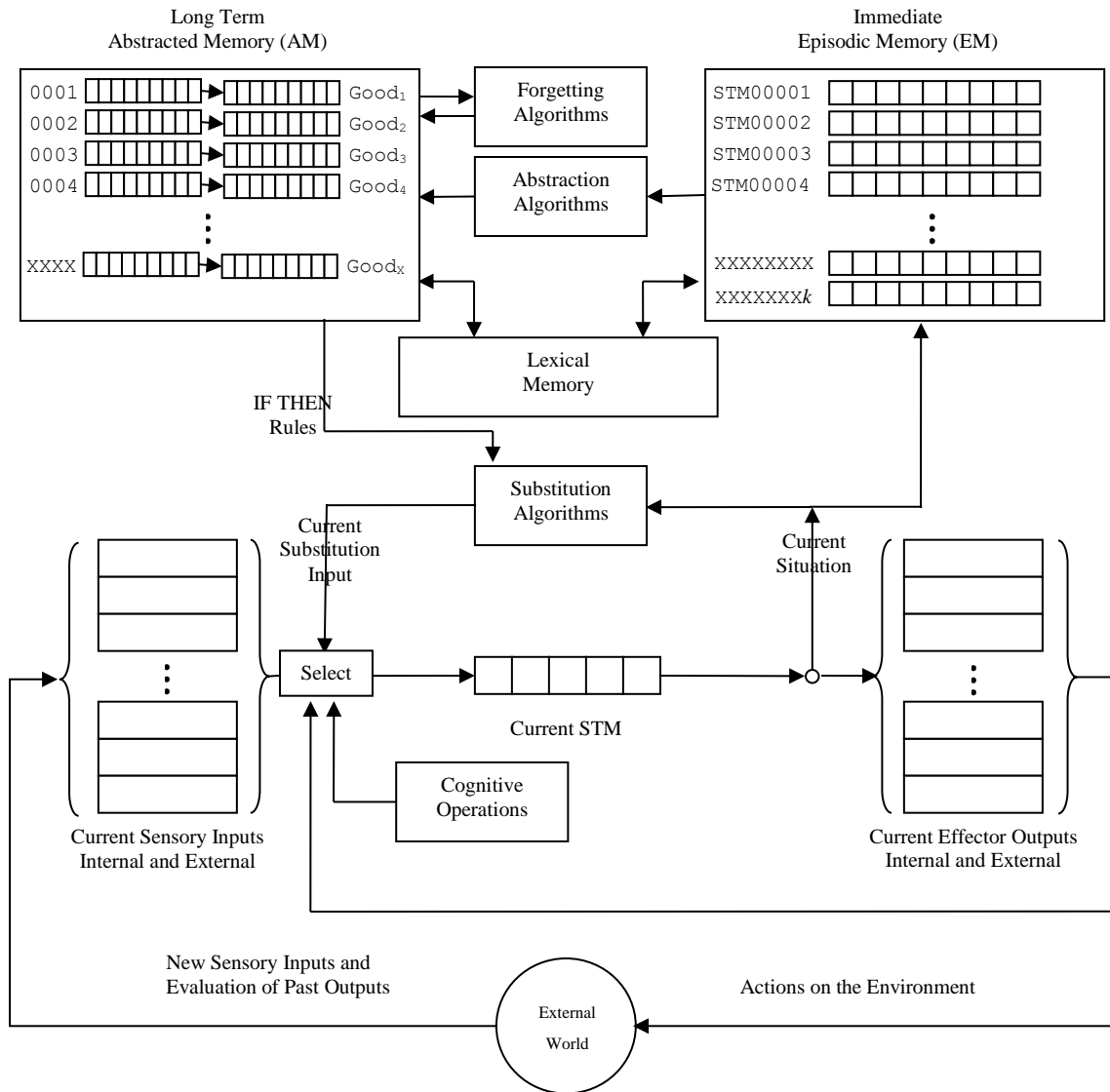

Fig. 1. The Cognitive Architecture

3.1 Components

1) Sensors: Sensory inputs are the ingredients out of which intelligence is created. Books, websites, cameras, and microphones are sources of output in the external world. Text, images, and sounds are the outputs from these sources. As shown in Fig. 1, a cognitive system has external sensors (e.g., eyes, ears, sense of touch, web input sensor, etc.) sensing external outputs and bringing them into the system's internal world as external inputs. Similarly, a cognitive system has internal sensors (e.g., memory system, hunger sensor, sense of balance, sex-drive sensor, etc.) sensing internal inputs.

2) Short Term Memory: Current STM is located in the center of Fig. 1. It holds what is currently happening to the cognitive system. In the nearby box labeled "select" are selectors that select what sensors have their inputs posted to the current STM. The current inputs selected pass through STM into EM.

3) Episodic Memory: The large box located at the top right of Fig. 1 is immediate EM, which is everything that ever happened to the cognitive system. EM records everything that ever happened in STM. EM has the same basic structure as STM, except there are k rows instead of one.

4) Abstracted Memory: The large box in the upper left corner of Fig. 1. is long term AM. It is the abstraction or generalization space. The nearby abstraction algorithms transform EM into AM. AM is constantly rewriting or changing current STM. The neighboring forgetting algorithms are where forgetting routines take place. The structure of AM is similar to EM except its rows are more tightly compressed. A goodness measure is attached to each abstraction row indicating how useful the abstraction is to the system.

5) Lexical Memory: Shown in the high center of Fig. 1 is a lexical memory or vocabulary (V = 1997, 1998, 350 v8, a, aardvark, aardvarks, abandon, ..., zoos), with usage to understand sensory inputs, which are the elemental values of information in the CDB. V is a language relational database for attaching syntax to a word sensed by a web input sensor.

6) Automaticity: In the center of Fig. 1 are the automaticity rules or substitution algorithms, which automatically rewrite current STM. For example, when a CDB senses a "dog," it automatically thinks "dog," and then thinks "cat," then "leash," and then thinks "bark," and so on. In this way, it seems more human. The more humanlike we create the CDB, the more intelligent the system will appear.

7) Control Routine: The box near the bottom center of Fig. 1, within the cognitive operations box, is the main executive (ME). This is the top-level manager. For instance, ME is on a similar level as the Microsoft Excel application; anything dealing with the whole CDB goes here. Algorithms of the brain are not easily organized, because the order is not fixed. The order is more dynamic. ME is the control routine which decides when to execute what algorithms. For example, when to run the performance algorithm (PA), abstraction algorithms (AA), forgetting algorithms, or when to chunk MemNos together, etc. Sensors are a good place for ME to start because they bring input into the senses.

8) Effectors: CDB changes the external world by issuing commands to its external effectors (move robotic arm; flip on sensors/effectors, etc.). CDB also changes its internal world by issuing commands to its internal effectors (abstract, introspect, reflect, search, etc.).

3.2 Memory Algorithms

1) Performance Algorithm: The performance algorithm (PA) is where the CDB spends most of its time. It performs, senses, introspects, and it self-reflects. While the CDB is performing, AM is always trying to rewrite EM with rewrite rules, as can be seen in steps of the PA pseudo code:

```
Loop over lifetime.
    Get all inputs from sensors.
    Select new STM based on selection operation.
    Move selected new STM into current STM.
    Do any rewrite rules suggest a way to rewrite the
    current moment of STM?
        Yes, create new STM with rewritten content.
        No, continue.
    Loop back to get all inputs from sensors.
```

2) The Abstraction Algorithm: Sensory input comes into the brain faster than can be processed, so at night when we sleep (offline), we dream and try to organize inputs by integrating or connecting things. Down time is when abstracted seTs (T) and abstracted seQuences (Q) are most easily created and changed. The four transformational rules from transformational grammar (insertion, deletion, substitution, and movement) [1] are applied in a different way that we call grammatical abstraction. There are a set of three rules: {INS/DEL, SUB, MOV}. The INS and DEL rules can be seen as the same rule. The initial abstraction algorithm (AA_1), given in pseudo code, processes through the episodes of life in EM and creates useful abstractions in AM:

```
foreach Episode P in EM do
    for i = 1 to (# Experiences in P) - 1 do
        for j = (i + 1) to (# Experiences in P) do
            Experience X₁ = P[i];
            Experience X₂ = P[j];
            ABSTRACT₁ (X₁, X₂);

ABSTRACT₁ (Experience X₁, Experience X₂)
    if IS_SUB (X₁, X₂)
        SUB (X₁, X₂)
    if IS_INS (X₁, X₂)
        INS (X₁, X₂)
    if IS_MOV (X₁, X₂)
        MOV (X₁, X₂)
```

Of course, pseudo code cannot capture the true complexity of the process. For example, the IS_SUB predicate checks to see if two experiences are identical in form except for one position where they each have a different MemNo. If X_1 and X_2 do differ only in this one way, a potential object MemNo (T) is created along with a tentative relation MemNo (Q). For a more concrete example, if X_1 = "*the dog*" and X_2 = "*the cat*," then IS_SUB is true, and the sub action proceeds. The word *the* is the same on the left sides of both sentences, so a new sequence can be created, $Q_1 = \langle$the$\rangle$. Then, the words at the substitution position, *dog* and *cat*, are added to a new set, T_1 = {dog, cat}, and then added to $Q_1 = \langle$the {dog, cat}$\rangle$, resulting in $Q_1 = \langle$the $T_1\rangle$. This means *the* can be followed by either *dog* or *cat*. The other rules have similar logic to the sub rule.

4   Learning meaning

Meaning is sensory elements being integrated and correlated among the senses. For instance, you have pictures of water, sounds of water, the taste of water, the feeling of water, the cold temperature of water, a thirst for water, and the word *water*. The association of a word to its senses becomes its meaning. The "water" MemNo points to everything about water. "Water" points to "drink," "splash," "float," "swim," "wash," and so on. Each MemNo in turn points back to "water," such as "drink" points back to "water", and it also points to "cold water". The mind is optimized for efficiency, so we only stores about 10 maximum items about "water" in current STM.

From all these inputs coming into the system at the same time, the brain has to figure what is meaningful to connect together. The whole point is to live more intelligently by finding patterns in the operation of the universe and predicting what will come next, and then rewarding correct predictions. For example, if a lion roars and eats your friend, you had better start realizing when you hear a roar, you should start running.

4.1   Thought Experiment

An attempt to learn the meaning of a sentence can now be illustrated with an example. Imagine the first experience that a CDB has is X_1 = "*the man drank the shake*." The CDB processes X_1 and creates four new sensory MemNos: S_1 = "the," S_2 = "man," S_3 = "drank" and S_4 = "shake," as well as one new relational MemNo: $R_1 = \langle S_1, S_2, S_3, S_1, S_4 \rangle$. The utility of the new MemNos are all initialized to 1.

When the CDB next processes experience X_2 as "*the man drank the water*," S_5 = "water" is created as well as $R_2 = \langle S_1, S_2, S_3, S_1, S_5 \rangle$. Applying the AA_1 to these two experiences in the episode creates one new object MemNo, T_1 = {shake, water}, and one new relation MemNo, $Q_1 = \langle$the man drank the $T_1\rangle$.

As one might expect, the complex building of concepts is quite epiphenomenal, depending upon the exact sequence of experiences of the system.

4.2   Distance measure

Eventually, after experiencing many such sentences, the CDB will have formed a network of MemNos in STM. In order to stop the recursion from possibly activating more information than is useful, such as having the "dog" MemNo, followed by the "cat" MemNo, the "leash" MemNo, the "wolf" MemNo, and so on all the way to the "computer" MemNo. Therefore, the distance has to be considered and limited. An analogy of the process is the cells on the human retina. Two cells on the retina have similar experiences if they are next to each other in space. Cells further away from the points of contact have weaker associations. This is a universal principal of our universe, *not* only of humanity. The bound of the distance measure is a maximum of 10 MemNos from the source, a minium of zero MemNos, and normally activates within a distance of five MemNos. If something is more than five MemNos away, it is ignored, unless the goals or drives are great enough to throw an exception to explore up to 10 MemNos away from the MemNo being considered. Words use context to disambiguate meanings, exploiting that nearby words have closely related meanings.

5   Conclusions

The Cognitive Architecture and the CDB are evolving works. In the tradition of AI, the model is defined, and then its implications are observed in implementation. As Hawkins has observed [2], intelligence is largely held in the human memory system. This paper lays out the beginnings of The Cognitive Architecture to define human memory and the design of the Cognitive Data Base (CDB) to model human intelligence.

6   References

[1]   Noam Chomsky. "Aspects of the Theory of Syntax." MIT Press, 1986.

[2]   Jeff Hawkings and Subutai Ahmad. "Hierarchical Temporal Memory Including HTM Cortical Learning Algorithms." Groksolutions.com.

Measuring Divergences among Mass Functions

Jianbing Ma

School of Design, Engineering and Computing, Bournemouth University,
Bournemouth BH12 5BB, UK
jma@bournemouth.ac.uk

Abstract—**Evidence theory is widely used in data mining, machine learning, clustering and database systems. In these applications, often combination of mass functions is performed without checking the degree of relevance between the mass functions, which may lead to counterintuitive results. In this paper, we aim to measure the divergences among mass functions and hence can prevent highly inconsistent mass functions from being combined. Divergence measures are then proposed to this end. In addition, incompleteness measures and similarity measures are also provided based on divergence measures which are useful in practice.**

Keywords-**Divergence Measure; Similarity Measure; Evidence Theory; Clustering**

I. Introduction

The Dempster-Shafer theory of evidence [1], [2] is a well known framework to model and reason with incomplete information in intelligent systems. In knowledge discovery community, this theory is widely used in data mining [3], [4], knowledge representation and reasoning [5], [6], machine learning [7], [8], gender profiling [9], [10], [11], clustering [12], [13], bus surveillance [14], [15], [16], [17], and database systems [18], [19], etc. In these applications, generally Dempster's combination rule is applied to mass functions from multiple sources to obtain an aggregated result. However, this rule for combining two sources of conflicting beliefs is criticized by many authors [20], [21] as it may lead to some counterintuitive results such as an almost impossible choice (with a very small degree of belief) by both sources becomes to the unique and certain choice after combination.

To remedy this weakness, a set of alternative combination rules are proposed, e.g., [22], [23], [24], [25], [26], [27], [28], etc. These studies are mainly focused on investigating the conditions by which these alternatives can be used to resolve the conflict. However, a fundamental question that what does *conflict* mean between two sources of beliefs is ignored in a long time. These papers by default follow the perspective of Dempster's rule that the conflict can be measured by the so-called *conflicting* mass, i.e., the mass of the combined belief assigned to the empty set before normalization. In [21], this issue is observed and examined in details. Liu argued that the conflicting mass itself cannot be an adequate measurement for conflict between two sources of beliefs, but it needs to be grouped with another dimension of conflict measure, i.e., the distance between betting commitments of beliefs, to form a safe measure of conflict. More precisely, only when the two measures both give high values, it is safe to declare that the two sources of beliefs are in conflict.

However, in these two measures, the conflict brought by the incompleteness nature of the beliefs is ignored. For instance, these two measures might both give 0 when the two beliefs are not the same, e.g., when one belief is represented as a vacuous belief and the other is represented by an evenly distributed mass function. In that situation, the inconsistency of the two beliefs due to the incompleteness of the former would disappear in the two measures. Hence, it is interesting to investigate the inconsistency of beliefs on the base of incompleteness. In this paper, we propose a *divergence* measure to compare two sources of beliefs considering both uncertainty and incompleteness. Based on the divergence measure, the incompleteness measure is also provided.

Similarity measures are widely used in clustering [12], [13]. However, typically such measures are not well justified. In this paper, we also propose a similarity measure based on divergence measure and hence with good properties.

The rest of the paper is organized as follows. In Section 2, we recall some basic concepts of evidence theory. In Section 3, 4, and 5, we propose the divergence measure, the incompleteness measure and the similarity measure, respectively. Finally, in Section 6, we conclude the paper.

II. Preliminaries

By abuse of notation, when A is a set, $|A|$ denotes its cardinality, when a is a real value, $|a|$ denotes the absolute value of a. The semantics of $||$ will be made clear by the context.

For reader's convenience, we recall some basic concepts of Dempster-Shafer's theory of evidence.

Let Ω be a finite set called the frame of discernment. In this paper, we denote $\Omega = \{w_1, \ldots, w_n\}$.

Definition 1: A basic belief assignment (bba for short) is a mapping $m : 2^\Omega \to [0, 1]$ such that $\sum_{A \subseteq \Omega} m(A) = 1$.

A bba m is also called a mass function when $m(\emptyset) = 0$ is required.

If $m(A) > 0$, then A is called a focal element of m. Let $\mathscr{F}_m$ be the set of focal elements in m.

Two bbas m_1 and m_2 are called *Orthogonal* iff there exists $A_1 \in \mathscr{F}_{m_1}, A_2 \in \mathscr{F}_{m_2}$ such that $m_1(A_1) = m_2(A_2) = 1$ and $A_1 \cap A_2 = \emptyset$.

A bba m is called *Bayesian* iff all of its focal elements are singletons. In this situation, m is in fact a probability function. A bba m is called *Partitioned* iff its focal elements $A_1, \cdots, A_k$ satisfy

$$A_1 \cup \cdots \cup A_k = W, \quad and \quad A_i \cap A_j = \emptyset \ \ for \ \ i \neq j.$$

The famous Dempster's rule of combination on two bbas m_1 and m_2 from two distinct sources is defined as

$$m_{12}(C) = \frac{\sum_{A \cap B = C} m_1(A)m_2(B)}{1 - \sum_{A \cap B = \emptyset} m_1(A)m_2(B)}, \forall\, C \neq \emptyset. \quad (1)$$

Here m_{12} denotes the bba after applying Dempster's combination rule. We also use $m_{12}(\emptyset)$ to denote the conflicting mass made by Dempster's rule such that $m_{12}(\emptyset) = \sum_{A \cap B = \emptyset} m_1(A)m_2(B)$. Note that Dempster's rule only applies when $m_{12}(\emptyset) < 1$.

Definition 2: Let m be a bba over Ω. Its associated pignistic probability function $BetP_m : \Omega \to [0,1]$ is defined as:

$$BetP_m(w) = \sum_{A \subseteq \Omega, w \in A} \frac{1}{|A|} \frac{m(A)}{1 - m(\emptyset)}, m(\emptyset) < 1.$$

where $|A|$ is the cardinality of A.

$BetP_m$ is the *probability expectation function* of m [27]. Usually, $BetP_m(A)$ is called the *betting commitment* to A. Note that the transformation from m to $BetP_m$ eliminates the influence of incompleteness.

In [21], the *distance between betting commitments of beliefs* is defined as follows.

Definition 3: Let m_1, m_2 be two bbas over Ω, and their corresponding pignistic probability function be $BetP_{m_1}$ and $BetP_{m_2}$ respectively, then the distance between betting commitments of m_1 and m_2 is defined as: $difBetP_{m_1}^{m_2} = max_{A \subseteq \Omega}(|BetP_{m_1}(A) - BetP_{m_2}(A)|)$.

According to [21], $difBetP_{m_1}^{m_2}$ gives the maximum extent of the differences between betting commitments to all the subsets. Hereafter we will simply write $difBetP$ instead of $difBetP_{m_1}^{m_2}$ if there is no confusion as which two bbas are being compared.

$difBetP$ does not consider any incompleteness information, as illustrated by the following result.

Proposition 1: $difBetP = \frac{\sum_{w \in \Omega} |BetP^{m_1}(w) - BetP^{m_2}(w)|}{2}$.

This result shows that $difBetP$ totally depends on values of singletons, which does not involve incompleteness.

III. DIVERGENCE BETWEEN TWO BBAS

Although the conflict mass $m_{12}(\emptyset)$ and the difference between betting commitment $difBetP$ present a good measure as to whether two bbas are in conflict, they cannot distinguish two bbas which are different but in *harmony*. More precisely, there could be two different bbas such that the corresponding $m_{12}(\emptyset)$ and $difBetP$ both are zero.

Example 1: Let m_1, m_2 be two bbas from two distinct sources on frame $\Omega = \{w_1, w_2, w_3, w_4, w_5\}$ be:

$$m_1(\{w_1\}) = m_1(\{w_2\}) = m_1(\{w_3\})$$
$$= m_1(\{w_4\}) = m_1(\{w_5\}) = 0.2,$$
$$m_2(\{w_1, w_2, w_3, w_4, w_5\}) = 1.$$

Obviously, m_1 and m_2 are different, but we have $m_{12}(\emptyset) = 0$ and $difBetP = 0$.

From the above example, it is obvious that the conflict mass and the difference between betting commitment could not fully reflect the difference between two bbas. It might be argued that in some sense, m_1 and m_2 in the above example is *coherent*. However, in many applications, e.g., in classification problems [29], m_1 has a clear meaning that one item is classified as w_1 (resp. w_2, w_3, w_4, w_5) with a degree of certainty 0.2, whilst m_2 just tells that the item cannot be classified. That is, m_2 contains much more incompleteness than m_1. Therefore, we need a kind of measure that can distinguish these two bbas. Intuitively, the measure is required to have the following constraints:

Let div denote the measure and m_1, m_2 be two bbas.

Non-Negativeness

$$div(m_1, m_2) \geq 0, \text{ and } div(m_1, m_2) = 0 \text{ iff } m_1 = m_2.$$

Explanation: Any two different bbas could be distinguished by this measure.

Orthogonality

$$div(m_1, m_2) \leq 1, \text{ and } div(m_1, m_2) = 1$$

iff m_1 and m_2 are orthogonal.

Explanation: Intuitively and semantically, orthogonal bbas are in total conflict, and vice versa. Therefore, this measure gives a greatest value 1 to indicate this *total conflict* whilst for other occasions, this measure should give a value less than 1.

Obviously, $m_{12}(\emptyset)$ and $difBetP$, does not satisfy the above two conditions.

Now we start to define a *divergence* measure to describe the difference between two bbas satisfying the above conditions. This definition takes several steps. First, we define the divergence between two focal elements of two bbas, respectively.

Definition 4: Let m_1, m_2 be two bbas over Ω, and A_1, A_2 be two arbitrary focal elements of m_1 and m_2, respectively, we define the divergence of A_1 and A_2 w.r.t m_1 and m_2 as:

$$div_{m_2}^{m_1}(A_1, A_2) = \frac{1}{2}\Big[m_1(A_1)\frac{|A_1 - A_2|}{|A_1|}$$

$$+ m_2(A_2)\frac{|A_2 - A_1|}{|A_2|} + |m_1(A_1)\frac{|A_1 \cap A_2|}{|A_1|} - m_2(A_2)\frac{|A_1 \cap A_2|}{|A_2|}|\Big].$$

Here $m_1(A_1)\frac{|A_1 - A_2|}{|A_1|}$ indicates the part that A_1 differs from A_2, $m_2(A_2)\frac{|A_2 - A_1|}{|A_2|}$ indicates the part that A_2 differs from A_1, and $|m_1(A_1)\frac{|A_1 \cap A_2|}{|A_1|} - m_2(A_2)\frac{|A_1 \cap A_2|}{|A_2|}|$ indicates the difference of the intersected part.

In particular, if $A_1 \cap A_2 = \emptyset$, then $div_{m_2}^{m_1}(A_1, A_2) = \frac{1}{2}(m_1(A_1) + m_2(A_2))$. If $A_1 = A_2 = A$, then $div_{m_2}^{m_1}(A_1, A_2) = \frac{1}{2}(|m_1(A) - m_2(A)|)$. $div_{m_2}^{m_1}(A_1, A_2)$ can be seen as the degree that A_1 can be adapted to A_2.

We find that $div_{m_2}^{m_1}(A_1, A_2)$ has some good properties, i.e., it is a super-distance.

Proposition 2: Let m_1, m_2 be two bbas over Ω, and A_1, A_2 be two arbitrary focal elements of m_1 and m_2, respectively, then we have

F-Super-NonNegativeness

$$div_{m_2}^{m_1}(A_1, A_2) \geq 0, \text{ and } div_{m_2}^{m_1}(A_1, A_2) = 0$$

iff $A_1 = A_2$ and $m_1(A_1) = m_2(A_2)$.

F-Symmetry
$$div_{m_2}^{m_1}(A_1, A_2) = div_{m_1}^{m_2}(A_2, A_1).$$

F-Triangle Inequity Let A_3 be a focal element of a bba m_3 over Ω, then we have $div_{m_2}^{m_1}(A_1, A_2) + div_{m_3}^{m_2}(A_2, A_3) \geq div_{m_1}^{m_3}(A_3, A_1)$.

F-Orthogonality $div_{m_2}^{m_1}(A_1, A_2) = 1$ iff $A_1 \cap A_2 = \emptyset$ and $m_1(A_1) = m_2(A_2) = 1$.

Here $F-$ indicates that these properties are for divergence between focal elements.

We then define the divergence between a focal element A_1 of m_1 to m_2, i.e., $div^{m_1}(A_1, m_2)$ as follows.

Definition 5: Let m_1, m_2 be two bbas over Ω, and A_1 be an arbitrary focal element of m_1, then we define the divergence of A_1 to m_2 as

$$div^{m_1}(A_1, m_2) = min_{A_2 \in \mathscr{F}_{m_2}} div_{m_2}^{m_1}(A_1, A_2).$$

By convention the divergence of A_1 to m_2 is the minimal divergence of A_1 to any focal element of m_2. $div^{m_1}(A_1, m_2)$ can be viewed as the minimal degree that A_1 can be adapted to m_2.

Based on $div^{m_1}(A_1, m_2)$, we can define the divergence of one bba m_1 to another bba m_2, i.e., $div^{m_1}(m_2)$, as follows.

Definition 6: Let m_1, m_2 be two bbas over Ω, then we define the divergence of m_1 to m_2 as $div^{m_1}(m_2) = max_{A_1 \in \mathscr{F}_{m_1}} div^{m_1}(A_1, m_2)$.

Here the reason why we use max instead of using min is that if m_1 and m_2 are partly consistent, using min will make the divergence equivalent to zero.

Example 2: Let m_1, m_2 be two bbas from two distinct sources on frame $\Omega = \{w_1, w_2, w_3, w_4, w_5\}$ be:

$$m_1(\{w_1\}) = 0.1, m_1(\{w_2\}) = m_1(\{w_3\}) = 0.45,$$
$$m_2(\{w_1\}) = 0.1, m_2(\{w_4\}) = m_2(\{w_5\}) = 0.45.$$

If we use min in Def. 6, then as m_1 and m_2 are partly consistent on $\{w_1\}$, we get $div^{m_1}(m_2) = 0$ which is evidently not reasonable.

The reason why we use max instead of using $+$ is to avoid multiple uses of one focal element, as illustrated by the following example.

Example 3: Let m_1, m_2 be two bbas from two distinct sources on frame $\Omega = \{w_1, w_2, w_3, w_4, w_5\}$ be:

$$m_1(\{w_1\}) = 0.1, m_1(\{w_2\}) = m_1(\{w_3\}) = 0.45,$$
$$m_2(\{w_1\}) = 0.1, m_2(\{w_4\}) = m_2(\{w_5\}) = 0.45.$$

Note that $div^{m_1}(\{w_1\}, m_2) = div^{m_1}(\{w_1\}, \{w_1\})$, $div^{m_1}(\{w_2\}, m_2) = div^{m_1}(\{w_2\}, \{w_1\})$, and $div^{m_1}(\{w_3\}, m_2) = div^{m_1}(\{w_3\}, \{w_1\})$. Hence if we use $+$ instead of max in Def. 6, then $\{w_1\}$ in m_2 will be used three times in summation (since we assume to use $+$) which is not reasonable.

If we consider $div^{m_1}(A, m_2)$ as the degree of minimal change that A can be adapted to a focal element in m_2, then $div^{m_1}(m_2)$ can be regarded as the upper bound of all minimal changes that focal elements of m_1 can be adapted to m_2.

For divergence of one bba to another, we can prove that it satisfies the proposed two conditions. Namely, we have

Proposition 3: Let m_1, m_2 be two bbas over Ω, then $div^{m_1}(m_2)$ satisfies the NonNegativeness and Orthogonality properties as:

NonNegativeness $div^{m_1}(m_2) \geq 0$, and $div^{m_1}(m_2) = 0$ iff $m_1 = m_2$.

Orthogonality $div^{m_1}(m_2) \leq 1$ and $div^{m_1}(m_2) = 1$ iff m_1, m_2 are orthogonal.

It is not very surprising that $div^{m_1}(m_2) \neq div^{m_2}(m_1)$. Since although $div^{m_1}(m_2)$ is distance alike, it is not a distance measure. $div^{m_1}(m_2)$ gives the maximal degree of all the focal elements of m_1 in *adaption* whilst $div^{m_2}(m_1)$ is for focal elements of m_2 in *adaption*. But the adaption itself is not symmetrical. That is, if $div^{m_1}(A_1, m_2) = div^{m_1}(A_1, A_2)$, then it is not necessary that $div^{m_2}(A_2, m_1) = div^{m_2}(A_2, A_1)$.

Of course, we can define symmetric divergence measures between two bbas based on $div^{m_1}(m_2)$.

Definition 7: Let m_1, m_2 be two bbas over Ω, then we define the divergence between m_1 and m_2 as $div(m_1, m_2) = max(div^{m_1}(m_2), div^{m_2}(m_1))$.

$div(m_1, m_2)$ gives the maximal degree of minimal changes that m_1 and m_2 could be adapted interchangeably.

It is obvious that $div(m_1, m_2)$ also satisfies the proposed two conditions.

Proposition 4: Let m_1, m_2 be two bbas over Ω, then $div(m_1, m_2)$ satisfies the NonNegativeness and Orthogonality properties.

NonNegativeness $div(m_1, m_2) \geq 0$, and $div(m_1, m_2) = 0$ iff $m_1 = m_2$.

Orthogonality $div(m_1, m_2) \leq 1$ and $div(m_1, m_2) = 1$ iff m_1, m_2 are orthogonal.

IV. AN INCOMPLETENESS MEASURE OF A BBA

It is well known that probability measures are used to represent uncertainty and bbas are used to represent both uncertainty and incompleteness. However, there are no measures on bbas to detect in what degree it is incomplete. We think this fundamental problem should be considered. Therefore, in this section, we propose a measure on the incompleteness of a bba. This incompleteness measure is based on the divergence measure $div(m_1, m_2)$. To illustrate the problem clearer, we first look at the following examples.

Example 4: Let m_1, m_2, m_3 be three bbas from three distinct sources on the same frame be:

$$m_1(w_1) = m_1(w_2) = m_1(w_3) = m_1(w_4) = m_1(w_5) = 0.2,$$
$$m_2(\{w_1, w_2\}) = 0.4, m_2(\{w_3, w_4, w_5\}) = 0.6.$$
$$m_3(\{w_1, w_2, w_3, w_4, w_5\}) = 1.$$

For m_1 and m_2, we get $m_{12}(\emptyset) = 0.48$, $difBetP_{m_2}^{m_1} = 0$, and $div(m_1, m_2) = 0.2$.

For m_1 and m_3, we get $m_{13}(\emptyset) = 0$, $difBetP_{m_3}^{m_1} = 0$, but $div(m_1, m_3) = 0.4$.

It is not surprising to see that $div(m_1, m_3)$ is greater than $div(m_1, m_2)$. Compared with m_2, although the corresponding pignistic functions of m_2 and m_3 are the same, m_3 increases the incompleteness of belief. For the conflict mass $m_{12}(\emptyset)$ and

$m_{13}(\emptyset)$, it is somehow counterintuitive that increasing incompleteness actually reduces conflict. For $difBetP$, it simply cannot measure the change of incompleteness in this occasion. In fact, as $difBetP$ is based on pignistic transformation which completely removes the incompleteness (Prop. 1), this is not astonishing. In contrast, $div(m_1, m_2)$ can nicely reflect the change of incompleteness. In this sense, $div(m_1, m_2)$ can be seen as depicting the conflict brought by incompleteness.

Now we define the incompleteness of a bba.

Definition 8: Let m be a bba over Ω, then we define the incompleteness of m as

$$icmp(m) = div(m, BetP_m)$$

where $BetP_m$ be the pignistic probability function of m. Intuitively, as $BetP_m$ is the corresponding pignistic probability function of m, in some sense, it can be seen as measuring the uncertainty part of m, hence a divergence measure between m and $BetP_m$ can intuitively *remove* the influence of uncertainty in m, and reveal the incompleteness of m only.

Example 5: Let m be a bba over $\Omega = \{w_1, w_2, w_3\}$ such that $m_1(\{w_1, w_2\}) = 0.7, m_1(\{w_3\}) = 0.3$, then we have $icmp(m) = 0.35$.

Naturally, if m is already a Bayesian bba, then it is not incomplete. In fact, we actually have the following result.

Proposition 5: Let m be a Bayesian bba over Ω, then we have $icmp(m) = 0$.

A direct and clear illustration of incompleteness is the partitioned bbas.

Example 6: Let m_1, m_2 be two bbas over $\Omega = \{w_1, w_2, w_3, w_4, w_5\}$ such that $m_1(\{w_1, w_2, w_3\}) = 0.6, m_1(\{w_4, w_5\}) = 0.4$, and $m_2(\{w_1\}) = 0.2, m_2(\{w_2, w_3\}) = 0.4, m_2(\{w_4\}) = 0.2, m_2(\{w_5\}) = 0.2$.

Obviously, m_1 is more incomplete than m_2. In fact, we do have $icmp(m_1) = 0.4 > 0.2 = icmp(m_2)$.

It could be generalized to the following result. But first we need to introduce a concept. For two partitioned bbas m_1 and m_2, m_2 is called *finer* than m_1 iff any focal element of m_2 is a subset of a focal element of m_1, i.e., $\forall A_2 \in \mathscr{F}_{m_2}$, $\exists A_1 \in \mathscr{F}_{m_1}$, s.t., $A_2 \subseteq A_1$.

Proposition 6: Let m_1, m_2 be two partitioned bbas and m_2 be finer than m_1. If $BetP_{m_1} = BetP_{m_2}$, then we have $icmp(m_1) \geq icmp(m_2)$.

V. Similarity Measures

In [13], similarity measures based on bbas are used for clustering as follows.

Let $\mathscr{M}_i = \bigcup_{A \in \mathscr{F}_{m_i}} A$ be the union of all focal elements in m_i, the similarity measure from [13] is $sim(m_1, m_2) = min(\sum_{A \subseteq \mathscr{M}_2} m_1(A), \sum_{B \subseteq \mathscr{M}_1} m_2(B))$.

In [12], entropy based similarity measures are used for clustering. In both papers, the proposed similarity measures are not well justified.

It is obvious that divergence measures and similarity measures are two sides of one coin. Namely, we can define our similarity measure based on divergence measures.

Definition 9: Let m_1, m_2 be two bbas over Ω, then a similarity measure is defined as

$$sim'(m_1, m_2) = 1 - div(m_1, m_2).$$

Properties of the similarity measure can be easily induced from those of the divergence measure.

VI. Conclusion

In this paper, we investigated how to measure the difference between two bbas and the incompleteness information contained by a bba. We hence proposed divergence measures for bbas indicating the degree to which one bba can adapt to another. Based on the divergence measure, incompleteness and similarity measures were also provided. In addition, properties of the proposed measures were studied.

Measuring the divergence between bbas could also be helpful for belief merging and revision which combine information from multiple sources [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40]. Divergence information can be seen as a clue for whether sources are independent or not.

For future work, we want to study the sufficient and necessary conditions when $div_{m_2}^{m_1} = div_{m_1}^{m_2}$. Furthermore, we aim to use the measures in real applications like clustering and data mining.

References

[1] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *The Annals of Statistics*, vol. 28, pp. 325–339, 1967.

[2] G. Shafer, *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[3] S. Anand, D. Bell, and J. Hughes, "Edm: A general framework for data mining based on evidence theory," *Data & Knowledge Engineering*, vol. 18(3), pp. 189–223, 1996.

[4] Y. Aslandogan, G. Mahajani, and S. Taylor, *Inter. Conf. on Information Technology: Coding and Computing*, 2004, vol. 2, ch. Evidence Combination in Medical Data Mining.

[5] J. Ma, W. Liu, and P. Miller, "A characteristic function approach to inconsistency measures for knowledge bases," in *Procs. of SUM*, 2012, pp. 473–485.

[6] J. Ma, W. Liu, and P. Miller, "Belief change with noisy sensing in the situation calculus," in *Procs. of UAI*, 2011.

[7] S. Hegarat-Mascle, I. Bloch, and D. Vidal-Madjar, "Application of dempster-shafer evidence theory to unsupervised classification in multisource remote sensing," *IEEE transactions on geoscience and remote sensing*, vol. 35(4), pp. 795–979, 1997.

[8] D. Zeng, J. Xu, and G. Xu, "Data fusion for traffic incident detection using d-s evidence theory with probabilistic svms," *Journal of Computers*, vol. 3(10), 2008.

[9] J. Ma, W. Liu, and P. Miller, "Evidential fusion for gender profiling," in *Procs. of SUM*, 2012, pp. 514–524.

[10] J. Ma, W. Liu, and P. Miller, "An improvement of subject reacquisition by reasoning and revision," in *Procs. of SUM*, 2013.

[11] J. Ma, W. Liu, and P. Miller, "An evidential improvement for gender profiling," in *Procs. of Belief Functions*, 2012, pp. 29–36.

[12] W. Perry and H. Stephanou, *Proc. of IEEE Inter. Symp. on Intelligent Control*, 1991, ch. Belief function divergence as a classifier, pp. 280–285.

[13] Y. Xie and V. Phoha, "K-cap'01: Procs. of 1st inter. conf. on knowledge capture." ACM, 2001, pp. 202–208.

[14] J. Ma, W. Liu, P. Miller, and W. Yan, "Event composition with imperfect information for bus surveillance," in *Procs. of AVSS*. IEEE Press, 2009, pp. 382–387.

[15] J. Ma, W. Liu, and P. Miller, "Handling sequential observations in intelligent surveillance," in *Proceedings of SUM*, 2011, pp. 547–560.

[16] P. Miller, W. Liu, F. Fowler, H. Zhou, J. Shen, J. Ma, J. Zhang, W. Yan, K. McLaughlin, and S. Sezer, "Intelligent sensor information system for public transport: To safely go ...." in *Procs. of AVSS*, 2010.

[17] J. Ma, W. Liu, and P. Miller, "Event modelling and reasoning with uncertain information for distributed sensor networks," in *Procs. of SUM.* Springer, 2010, pp. 236–249.

[18] S. McClean and B. Scotney, "Using evidence theory for the integration of distributed databases," *International Journal of Intelligent Systems*, vol. 12(10), pp. 763–776, 1998.

[19] D. He, A. Göker, and D. Harper, "Combining evidence for automatic web session identification," *Information Processing & Management*, vol. 38(5), pp. 727–742, 2002.

[20] L. Zadeh, "A simple view of the dempster-shafer theory of evidence and its implication for the rule of combination," *AI Magazine*, vol. 7, pp. 85–90, 1986.

[21] W. Liu, "Analyzing the degree of conflict among belief functions," *Artificial Intelligence*, vol. 170, pp. 909–924, 2006.

[22] P. Smets and R. Kennes, "The transferable belief model," *Artificial Intelligence*, vol. 66(2), pp. 191–234, 1994.

[23] P. Smets, "Proc. of fusion 2000 intern. conf. on information fusion," Paris, July 2000.

[24] D. Dubois and H. Prade, "Representation and combination of uncertainty with belief functions and possibility measures," *Comput. Intel.*, vol. 4, pp. 244–264, 1988.

[25] R. Yager, "On the relationships of methods of aggregation of evidence in expert systems," *Cybernetics and Systems*, vol. 16, pp. 1–21, 1985.

[26] R. Yager, "On the dempster-shafer framework and new combination rules," *Inform. Sci.*, vol. 41, pp. 93–138, 1987.

[27] A. Josang, "The consensus operator for combining beliefs," *Artificial Intelligence*, vol. 141(1-2), pp. 157–170, 2002.

[28] F. Smarandache and J. Dezert, "An introduction to the dsm theory for the combination of paradoxical, uncertain, and imprecise sources of information," http://arxiv.org/abs/cs/0608002.

[29] Y. Bi, J. Guan, and D. Bell, "The combination of multiple classifiers using an evidential reasoning approach," *Artif. Intell.*, vol. 172(15), pp. 1731–1751, 2008.

[30] J. Ma and W. Liu, "A general model for epistemic state revision using plausibility measures," in *Procs. of ECAI*, 2008, pp. 356–360.

[31] J. Ma and W. Liu, "Modeling belief change on epistemic states," in *Procs. of FLAIRS*, 2009.

[32] J. Ma, W. Liu, D. Dubois, and H. Prade, "Revision rules in the theory of evidence," in *Procs. of ICTAI*, 2010, pp. 295–302.

[33] J. Ma, W. Liu, D. Dubois, and H. Prade, "Bridging jeffrey's rule, agm revision and dempster conditioning in the theory of evidence," *International Journal on Artificial Intelligence Tools*, vol. 20(4), pp. 691–720, 2011.

[34] J. Ma and W. Liu, "A framework for managing uncertain inputs: An axiomization of rewarding," *Int. J. Approx. Reasoning*, vol. 52(7), pp. 917–934, 2011.

[35] J. Ma, W. Liu, and S. Benferhat, "A belief revision framework for revising epistemic states with partial epistemic states," in *Procs. of AAAI*, 2010, pp. 333–338.

[36] J. Ma, W. Liu, and A. Hunter, "Inducing probability distributions from knowledge bases with (in)dependence relations," in *Procs. of AAAI*, 2010, pp. 339–344.

[37] J. Ma, W. Liu, and A. Hunter, "Modeling and reasoning with qualitative comparative clinical knowledge," *Int. J. Intell. Syst.*, vol. 26(1), pp. 25–46, 2011.

[38] J. Ma, "Qualitative approach to bayesian networks with multiple causes," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 42(2), pp. 382–391, 2012.

[39] J. Ma, S. Benferhat, and W. Liu, "Revising partial pre-orders with partial pre-orders: A unit-based revision framework," 2012.

[40] J. Ma, S. Benferhat, and W. Liu, "Revision over partial pre-orders: A postulational study," in *Procs. of SUM*, 2012, pp. 219–232.

Appendix

We only give the proof of Proposition 2. Other proofs are omitted due to their straightforwardness.

Proof of Proposition 2: The F-symmetry property is straightforward.

For F-Super-NonNegativeness, we get $div_{m_2}^{m_1}(A_1, A_2)$ iff $m_1(A_1)\frac{|A_1 - A_2|}{|A_1|} = m_2(A_2)\frac{|A_2 - A_1|}{|A_2|} = |m_1(A_1)\frac{|A_1 \cap A_2|}{|A_1|} - m_2(A_2)\frac{|A_1 \cap A_2|}{|A_2|}| = 0$, as A_1, A_2 are focal elements, from $m_1(A_1)\frac{|A_1 - A_2|}{|A_1|} = m_2(A_2)\frac{|A_2 - A_1|}{|A_2|} = 0$, we get $|A_1 - A_2| = |A_2 - A_1| = 0$, namely $A_1 = A_2$, hence from $|m_1(A_1)\frac{|A_1 \cap A_2|}{|A_1|} - m_2(A_2)\frac{|A_1 \cap A_2|}{|A_2|}| = 0$, we get $m_1(A_1) = m_2(A_2)$.

For F-Triangle Inequity, let $x_1 = \frac{m_1(A_1)}{2*|A_1|}$, $x_2 = \frac{m_2(A_2)}{2*|A_2|}$ and $x_3 = \frac{m_3(A_3)}{2*|A_3|}$, notice that $|A - A'| = |A| - |A \cap A'|$, we only need to show for any $0 < x_1, x_2, x_3$, $x_1(|A_1| - |A_1 \cap A_3|) + x_2(|A_2| - |A_2 \cap A_3|) + x_3(|A_3| - |A_1 \cap A_3|) + x_3(|A_3| - |A_2 \cap A_3|) + |(x_1 - x_3)|A_1 \cap A_3| + |(x_2 - x_3)|A_2 \cap A_3| \geq x_1(|A_1| - |A_1 \cap A_2|) + x_2(|A_2| - |A_2 \cap A_1|) + |(x_1 - x_2)|A_1 \cap A_2|$.

For simplicity, denote $L(X, Y, Z) = X(|A_1| - |A_1 \cap A_3|) + Y(|A_2| - |A_2 \cap A_3|) + Z(|A_3| - |A_1 \cap A_3|) + Z(|A_3| - |A_2 \cap A_3|) + |(X - Z)|A_1 \cap A_3| + |(Y - Z)|A_2 \cap A_3|$ and $R(X, Y) = X(|A_1| - |A_1 \cap A_2|) + Y(|A_2| - |A_2 \cap A_1|) + |(X - Y)|A_1 \cap A_2|$, we need to show $L(x_1, x_2, x_3) \geq R(x_1, x_2)$. We consider the following three cases.

Case 1: If $x_1 \leq x_2 \leq x_3$, let $\delta_x = x_3 - x_2 \geq 0$, we get $L(x_1, x_3, x_3) = L(x_1, x_2, x_3) + \delta_x(|A_2| - |A_2 \cap A_3|) - \delta_x|A_2 \cap A_3|$ and $R(x_1, x_3) = R(x_1, x_2) + \delta_x(|A_2| - |A_1 \cap A_2|) + \delta_x|A_1 \cap A_2| = R(x_1, x_2) + \delta_x|A_2|$. Hence we only need to $L(x_1, x_3, x_3) - \delta_x(|A_2| - |A_2 \cap A_3|) + \delta_x|A_2 \cap A_3| \geq R(x_1, x_3) - \delta_x|A_2|$ or $L(x_1, x_3, x_3) + 2 * \delta_x|A_2 \cap A_3| \geq R(x_1, x_3)$. As $2 * \delta_x|A_2 \cap A_3| \geq 0$, it is sufficient to show $L(x_1, x_3, x_3) \geq R(x_1, x_3)$. Let $\lambda_x = x_3 - x_1 \geq 0$, we get $L(x_1, x_3, x_3) = L(x_1, x_1, x_1) + \lambda_x(|A_2| - |A_2 \cap A_3|) + \lambda_x(|A_3| - |A_1 \cap A_3|) + \lambda_x(|A_3| - |A_2 \cap A_3|) + \lambda_x|A_1 \cap A_3|$ and $R(x_1, x_3) = R(x_1, x_1) + \lambda_x(|A_2| - |A_2 \cap A_1|) + \lambda_x|A_1 \cap A_2|$. So we remain to show $L(x_1, x_1, x_1) + \lambda_x(|A_2| - |A_2 \cap A_3|) + \lambda_x(|A_3| - |A_1 \cap A_3|) + \lambda_x(|A_3| - |A_2 \cap A_3|) + \lambda_x|A_1 \cap A_3| \geq R(x_1, x_1) + \lambda_x(|A_2| - |A_2 \cap A_1|) + \lambda_x|A_1 \cap A_2|$ which can be simplified to $L(x_1, x_1, x_1) + 2 * \lambda_x(|A_3| - |A_2 \cap A_3|) \geq R(x_1, x_1)$. As $2 * \lambda_x(|A_3| - |A_2 \cap A_3|) \geq 0$, it is sufficient to show $L(x_1, x_1, x_1) \geq R(x_1, x_1)$ which can be written as $x_1(|A_1| - |A_1 \cap A_3|) + x_1(|A_2| - |A_2 \cap A_3|) + x_1(|A_3| - |A_1 \cap A_3|) + x_1(|A_3| - |A_2 \cap A_3|) \geq x_1(|A_1| - |A_1 \cap A_2|) + x_1(|A_2| - |A_2 \cap A_1|)$ and simplified to $|A_3| + |A_1 \cap A_2| \geq |A_1 \cap A_3| + |A_2 \cap A_3|$, let $|A_1 \cap A_3| = a$, $|A_2 \cap A_3| = b$, and $|A_1 \cap A_2 \cap A_3| = c$, we have $|A_3| \geq a + b - c$ and $|A_1 \cap A_2| \geq c$, therefore, $|A_3| + |A_1 \cap A_2| \geq a + b = |A_1 \cap A_3| + |A_2 \cap A_3|$. So finally we proved $L(x_1, x_2, x_3) \geq R(x_1, x_2)$ when $x_1 \leq x_2 \leq x_3$. Similar proofs can be made for the case $x_2 \leq x_1 \leq x_3$.

Case 2: If $x_3 \leq x_1 \leq x_2$, then if $|A_3| - |A_1 \cap A_3| - |A_2 \cap A_3| \geq 0$, we get $L(x_1, x_2, x_3) = L(x_1, x_2, 0) + x_3(|A_3| - |A_1 \cap A_3|) + x_3(|A_3| - |A_2 \cap A_3|) - x_3|A_1 \cap A_3| - x_3|A_2 \cap A_3| = L(x_1, x_2, 0) + 2 * x_3(|A_3| - |A_1 \cap A_3| - |A_2 \cap A_3|) \geq L(x_1, x_2, 0) = x_1(|A_1| - |A_1 \cap A_3|) + x_2(|A_2| - |A_2 \cap A_3|) + x_1|A_1 \cap A_3| + x_2|A_2 \cap A_3| = x_1|A_1| + x_2|A_2| \geq x_1|A_1| + x_2|A_2| - (x_1|A_1 \cap A_2| + x_2|A_2 \cap A_1| - x_1|A_1 \cap A_2| - x_2|A_1 \cap A_2|) = R(x_1, x_2)$. If $|A_3| - |A_1 \cap A_3| - |A_2 \cap A_3| < 0$, let $\delta_x = x_1 - x_3 \geq 0$, we have $L(x_1, x_2, x_1) = L(x_1, x_2, x_3) + \delta_x(|A_3| - |A_1 \cap A_3|) + \delta_x(|A_3| - |A_2 \cap A_3|) - \delta_x|A_1 \cap A_3| - \delta_x|A_2 \cap A_3| = L(x_1, x_2, x_3) + 2 * \delta_x(|A_3| - |A_1 \cap A_3| - |A_2 \cap A_3|) \leq L(x_1, x_2, x_3)$, so we only need to show $L(x_1, x_2, x_1) \geq R(x_1, x_2)$ which can be written as $x_1(|A_1| - |A_1 \cap A_3|) + x_2(|A_2| - |A_2 \cap A_3|) + x_1(|A_3| - |A_1 \cap A_3|) + x_1(|A_3| - |A_2 \cap A_3|) + (x_2 - x_1)|A_2 \cap A_3| \geq x_1(|A_1| - |A_1 \cap A_2|) + x_2(|A_2| - |A_2 \cap A_1|) + (x_2 - x_1)|A_1 \cap A_2|$ and simplified to $|A_3| + |A_1 \cap A_2| \geq |A_1 \cap A_3| + |A_2 \cap A_3|$ which

is already proved in Case 1. So we now have $L(x_1, x_2, x_3) \geq R(x_1, x_2)$ when $x_3 \leq x_1 \leq x_2$. Similar proofs can be made for the case $x_3 \leq x_2 \leq x_1$.

Case 3: If $x_1 \leq x_3 \leq x_2$, let $\delta_x = x_3 - x_1$, then we have $L(x_1, x_2, x_3) = L(x_1, x_2, x_1) + \delta_x(2 * |A_3| - |A_1 \cap A_3| - |A_2 \cap A_3|) + \delta_x|A_1 \cap A_3| + \delta_x|A_2 \cap A_3| = L(x_1, x_2, x_1) + 2 * \delta_x|A_3| \geq L(x_1, x_2, x_1)$, and in Case 2, we already showed that $L(x_1, x_2, x_1) \geq R(x_1, x_2)$. So we now have $L(x_1, x_2, x_3) \geq R(x_1, x_2)$ when $x_1 \leq x_3 \leq x_2$. Similar proofs can be made for the case $x_2 \leq x_3 \leq x_1$. Q. E. D.

Vehicle Image Classification Method Using Edge Dimensions, SVM and Prototype

Fabrízia Medeiros de S. Matos [1,2], **and Renata Maria Cardoso R. de Souza** [2]

[1]Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, João Pessoa, Brazil
[2]Centro de Informática, Universidade Federal de Pernambuco, Recife, Brazil

Abstract - *Automatic vehicle classification is an important task in Intelligent Transport System because it allows the attainment of the traffic parameter called vehicles count by category. Proposed methods of vehicle image classification, obtained from videos of roads traffic, have known limitations, such as strong dependence of detection methods, hard image normalization and low accuracy. This paper presents a vehicle classification method based on edge dimensions of the vehicle image. The method consists of three stages: image pre-processing, training and classification. In the pre-processing stage, the removal of horizontal bands over vehicle edge image is applied. The used features are edge dimensions, such as width and height, and as to the used classification rules there are prototype and SVM. The objectives of the method are acceptable accuracy and less dependence of detection methods. The classification results of distance to the prototype and SVM are compared with NN.*

Keywords: vehicle image classification, prototype, SVM

1 Introduction

Given the increasing social demands for mobility and safety in road transportation, many governments are making the development of automatic, economic, and real-time solutions for reliable traffic flow analysis a priority [1].

An Intelligent Transportation System (ITS) is defined as an application that incorporates electronic, computer, and communication technologies into vehicles and roadways for monitoring traffic conditions, reducing congestion, enhancing mobility, and so on. To achieve these goals, there have been many approaches for tackling related problems in ITS [2].

A vehicle classification system is essential for effective transportation system (to traffic management and toll system), parking optimization and law enforcement and autonomous navigation [3]. A vehicle classification system includes knowledge of areas as electronic technology and pattern recognition [4].

Automatic vehicle classification is an import task in ITS because it allows obtaining the traffic parameter called vehicles count by category, which is used to control and manage the roadways traffic [5].

Methods based on different sensors to get information about vehicles count by category, including video, magnetic induction coil, sound sensors, temperature sensors and microwave have been proposed. Magnetic induction coil has been used in ITS, but these devices have a short life. Its installation and maintenance inevitably cause damage to the road, once it obtains only little traffic information and it is also sensitive to traffic speed. The use of video has increased as support for traffic management due to the advantages such as acquisition price of sensors, installation without damaging the roads, easiness of transferring information to other devices geographically apart, and finally, the wide range of information available [4].

A key goal of automatic video-based traffic analysis is to detect and track vehicles driving through a controlled area and thus identify abnormal events, such as traffic congestion, speeding violations and other illegal driving behaviors, and even accidents. By using video, it is possible to compute traffic measures of involved vehicles, including their speeds, types, or overall numbers in the analyzed road region [1].

Different methods have been presented for vehicles classification based on images processing. Some methods use the edge of the vehicle image and define simple features over it, like width, height, number of point or area [6][7][8]. Others introduce descriptors, such as eigenfaces [4], Principal Components Analysis (PCA) [9][10], fractal dimension [10][11], Histogram of Oriented Gradients (HOG) and Measurement Based Feature (MBF) [13] . In the classification phase, methods make use of techniques, such as Nearest Neighbor (NN), Adaptive Nearest Neighbor (A-NN), neural networks, Support Vector Machine (SVM), Linear Discriminate Analysis (LDA) and Hidden Markov Models (HMM). With reference to accuracy rates, their results are high, even reaching 100% accuracy.

In the classification of vehicles image, the used database has variations like images obtained from video in different positions: frontal [4][10], lateral[5] and back [8].

Proposed methods do not have uniformity in relation to the number of classes. For example, some methods classify the image into two categories, but they are defined in different ways, car and not a car [7], taxi and bus [9]. Among the methods that use a greater number of classes, there is also no consensus with, for example, the use of three classes [6][12], four classes [10][11], five classes [13], seven classes [8] , eight classes [5] and nine classes [4].

The great dynamic of traffic real scenes consists in overcoming challenges to classifications. Lighting and climatic variations, shadows away from the camera, image quality, dynamic positioning of vehicles on roads and heavy reliance on detection techniques are considered aspects that

hinder the automatic methods from collecting traffic parameters based on image processing.

Vehicle detection methods make segmentation of the principal vehicle of the scene, however, image pre-processing methods have been proposed for increasing the segmentation quality and noise reduction.

Proposed method is a classification process including pre-processing, training and classification. The classification is based on minimum distance to the prototype and SVM. The test image is classified into motorbike, car, bus or truck. These four classes are defined based on reports of the Brazilian Ministry of Transport [14]. The separation of the bus and truck classes, instead of considering only large vehicles, is important for this ministry because there are planning, investments and political decisions, related to public transport, focused on vehicles of the bus class.

The classification is a hierarchical division with two levels. Firstly, a separation into two groups is fulfilled, a small and a large one; secondly, each group is subdivided into two final classes. In classification, the achieved success rate was 88% in the first level and 68% considering the final classification. Two analyses were made: (i) features based on edge dimensions and based on PCA; (ii) results classification of prototype and SVM versus NN. Finally, results are compared with the proposed ones in [10][11].

This paper is organized as follows: related works are presented in Section 2; details of the proposed methods, test methodology and database are presented in Section 3; and Section 4 reports experimental results. Finally, a conclusion is presented in Section 5.

2   Related Works

For increasing the segmentation quality and noise reduction, image pre-processing methods are presented in [4][5][9][10][11][13]. Dilation erosion and Gaussian median filter were used to minimize the impact of the noise in [4]. To normalized light intensity and suppressed noise, in [5], Gaussian parameters adapter and morphological operation were applied. To noise reduction, position and grayscale adjustments in [9], the techniques such as median filtrate, morphological operations and histogram equalization were used. Noise reduction was accomplished by applying the spatial wiener filter in [10]. Spatial wiener filter and horizontal bands removal were used in [11]to noise reduction and segmentation. Noise and shadow removal based on morphologic operation were proposed in [13].

A vehicle classification method based on eigenfaces is proposed in [4]. A vehicle face was defined as a rectangle from a front image of the vehicle. After normalization and some pre-processing steps, the eigenvectors of a vehicle face image are extracted and these eigenvectors are used for constructing a library of features of vehicle faces. In the classification, firstly, a vehicle face image and its eigenvector are extracted and then the difference between the vehicle face eigenvector and the eigenvectors in feature library are compared, using the minimum distance method. The achieved recognition rate is 100%, considering classes associated with different car factories (Volkswagen, BMW, Chevrolet, and others).

A statistical algorithm based on expectation-maximization is proposed for separation of two vehicle groups: car and not car [7]. The expectation-maximization algorithm was used to estimate parameters for a mixed Gaussian model based on the assumption that car and non-car dimensions, in different rows in the image, have distinct dimension distributions. The images are pre-processed by median and high-boost filters. The used features are height, width and area. While applying expectation-maximization algorithm including two steps (e-step and m-step), the probability of correct image vehicle classification was 0.9.

By using an analysis of Time-Spatial Image (TSI), seven features (width, area, compactness, length-width ratio, major axis, minor rectangularity and solidity) are obtained from the image for vehicle classification [8]. The proposed classifier requires a training set of data for which a linear discriminant array was created using the feature vectors. This training set of data is clustered in a desired number of vehicle classes using Fuzzy C-Means (FCM) algorithm. Finally, a decision is made that a vehicle belongs to a certain class by a process of majority voting among the FCM-based training data records in the neighborhood using a weighted distance measurement. Seven classes are considered and error rate ranges from 6,9% up to 14,29%, depending on controlled time conditions (normal, summer and clouds).

Features based on edge (number of edge points), block of PCA and two classification levels were proposed in [10]. The classification level 1, based on the number of edge points and on the distance to prototype, divides the vehicles into two groups, small and large. Afterwards, using PCA blocks of the original image as features and applying NN and A-NN, the classification level 2 divides each group into two classes: the small group is divided into motorcycle and car; the large group is divided into bus and truck. Twenty images were used in the tests, five in each class. The accuracy was 100% and 95%, respectively in classification level 1 and classification level 2.

Training SVM with MBF and IPHOG (intensity pyramid-based HOG) was proposed in [13]. Five classes were considering: car, van, bus, motorcycle and unknown. A 202-dimensional feature vector was constructed, comprising the MBF and IPHOG to train the SVM. A Kalmar filter tracks a vehicle to enable final classification by majority voting over consecutive frames. Classification accuracy of 94% was achieved.

A vehicle classification method using a set of images collected from six views of each vehicle was proposed in [12]. These six views include one frontal, one of back, two of lateral and two transverse. A set of salient discriminative features is collected. These are derived from a color image. Initially, image pyramids are created from the input image and the feature-dependent saliencies are computed in parallel. The salient map was then used to compute widh and height of the vehicle. Three vehicles categories were considered: sedan, SUV and sport (convertible). SVM classifiers are trained to recognize a specific type of vehicle and the final decision

depends on the majority voting of a classifying committee. The average recognition reached 99,13%.

2.1 Image Pre-processing Method Based on Edge

A pre-processing phase applied to image edge is presented in [11]. The objective of this method is to identify start and stop horizontal lines that seem to be the limits of the most relevant area of the edge image, that is, it removes horizontal lines that seem to be not vehicle parts.

The proposed method, presented in [11], removes horizontal lines based on contiguous horizontal line numbers with and without information (presence of edges points). This process is accomplished by means of the three following phases: (i) summary table is prepared including contiguous vertical line numbers with and without information; (ii) identification in summary table of the stop line; and (iii) search process in summary table up to start line identification.

Summary Table: Let us assume that matrix M (dimension r x c) is the image edge (binary matrix). Starting in the first line (line 1) up to the last line (line r), the number of lines with and without information is counted and stored, in sequence. New summary lines must be added whenever changes in the type of lines occur (with or without). Summary table must have six columns: the s_type column specifies if it is a line block with information (type =1) or not (type=0); the s_number column sets the contiguous line number of the respective type; s_col_1 and s_col_2 columns, significant only for line type equal to 1, represent the first and last columns considering all lines summarized in a specific summary line; and s_row_1 and s_row_2 columns, significant only for line type equal to 1, represent the first and last lines considering all lines summarized in a specific summary line.

Stop line: Considering that the vehicle is detected nearby the road white line, it was defined that the stop line must also be nearby the road white line. The stop line is then defined like the last summary line with s_type value 1.

Start line: A search process sets the start line with the summary line where the principal vehicle image starts. Conditions for three types of empty horizontal blocks are defined: large, medium and small.

3 Proposed Method

Proposed method for vehicles image classification is a general classification process including pre-processing, training and classification steps. It classifies a test image into motorbike, car, bus or truck.

In Figure 1, the block diagram of proposed vehicles image classification is presented, with three classical steps in dashed blocks. The pre-processing step makes a background subtraction, noise reduction, edge detection and band removal. These pre-processing steps were proposed in [12]. Training step makes selection of the six features based on edge dimensions of the vehicle image. Based on these features, the prototypes and SVM structure are defined. The used classification is a hierarchical process based on [10]:

firstly (level 1), is made a division into two groups (small and large), and secondly (level 2) is made a division of each group into two classes. Feature selection of test image must be done as the normalization applies to original image.
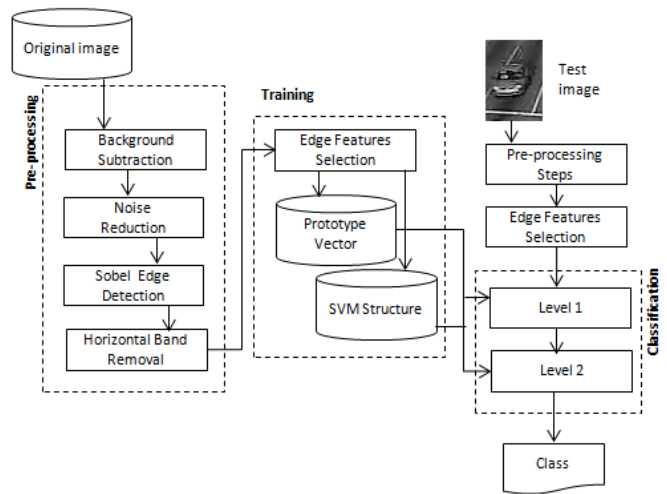

Fig. 1. Proposed classification diagram

3.1 Pre-Processing

The pre-processing applied over edge is based on [11]. However, the horizontal band removal is simplified for only two types, large and median. Small horizontal bands proposed in [11] are not considered here because, as the edge is not continuous, this removal deletes valid information of the vehicle image.

Background Subtraction: To produce the background subtraction, firstly a background image of the road, without vehicles, is carried out manually. Secondly, an algebraic subtraction between an original image and its background is performed.

Noise Reduction: Wiener filter is used with neighborhood size 5 x 5 [15].

Sobel Edge Detection: Sobel approximation edge detection is used [15].

Horizontal Band Removal: Removal of empty horizontal bands, large and small, are applied in sequence. Large and small empty bands are defined based on width of the image edge.

A summary table is elaborated including the column type, number of lines, first column and last column. The start line is located and the search process over the summary lines finds the stop line.

In Figure 2, pre-processing results are shown. An image sequence of a car (a) and of a motorcycle (b) is shown. The four first images of each class are, respectively, original image, background subtraction result, image noise reduction result and edge detection image. In a), the fifth image is the result of large horizontal band removal. In b), the fifth image is the result of small horizontal band removal.
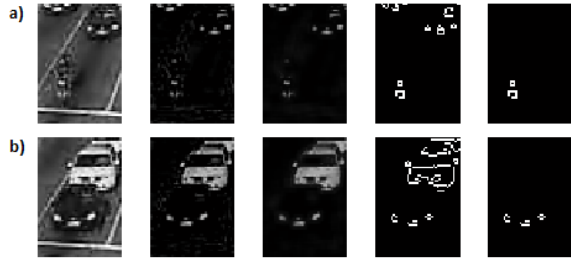
Fig. 2. Pre-processing: (a) motorcycle image, (b) car image

3.2 Training

The following features were selected: number of edge points (NP), width (W), height (H), fractal (F), block width (BW) and block height (BH). The first four features (NP, W, H, F) are described in [11].

Firstly, each feature value is calculated, after these values are normalized, dividing by maximum value of each feature.

Number of edge points (NP): Final pre-processed image is a binary matrix. The matrix elements with value equal to one (elements of the edge) are counted.

Width (W): It is the difference between the largest and the smallest columns, of horizontal summary table, limited by start and stop lines.

Height (H): It is the difference between the largest and the smallest rows, of horizontal summary table, limited by start and stop lines.

Fractal dimension (F): The fractal dimension is computed according to [16].

Block width (BW): Each line of the horizontal summary table, with s_type column equal to 1, is defined as a horizontal block, and its summary columns, $s\_col_1(i)$ and $s\_col_2(i)$, define the width of the block. Being n the number of blocks, BW is defined according to:

$$BW = (\textstyle\sum_1^n |s\_col_1(i) - s\_col_2(i)|)/n \qquad (1)$$

Block height (BH): Each line of the horizontal summary table, with s_type equal to 1, is defined as a vertical block, and its summary columns, $s\_row_1(i)$ and $s\_row_2(i)$, define the height of the block. Being n the number of blocks, BH is defined according to:

$$BW = (\textstyle\sum_1^n |s\_row_1(i) - s\_row_2(i)|)/n \qquad (2)$$

Additionally, in training the prototypes and SVM machines are defined.

Prototypes: The training image set is divided into two groups: small and large. Motorcycles and car images are placed into small groups and bus and trucks images are placed into large groups. In each group, a 6-dimensional prototype vector with the median value of each feature is stored.

Afterwards, in each group two prototypes, one for each class of the group, are defined. In each class, a 6-dimensional prototype vector with the median value of each feature is stored.

SVM: Feature vectors are used to elaborate the SVM structure. In a similar way to prototypes, first, a SVM for each group is defined, and second, one SVM for each class of that group are defined.

3.3 Classification

The classification process classifies a test image into motorbike, car, bus or truck. In the first step (level 1), a division into two groups (small and large) is made and in the second one (level 2), a division of each group into two classes is fulfilled: the small group is divided into motorcycle and car; the large group is divided into bus and truck. This hierarchical process is based on [10] .

Level 1 and level 2 of the classification, based on distance to prototype, are presented below.

Level 1: Let us assume that $\vec{X}$ is the feature vector of the test image and that $\vec{P}_1$ and $\vec{P}_2$ are feature vectors of the groups 1 and 2, respectively. Each vector is 6-dimensional. The level 1 classification affects the group i if $distance_i$ is the shortest distance, according to Equation 3:

$$distance_i = \vec{X} - \vec{P}_i \qquad (3)$$

Level 2: This classification is similar to level 1, having four prototypes, two in each group. If the test image is associated, in level 1, with the group i, the two prototypes of this group are considered in level 2 classification.

Classification SVM is similar to classification based on distance to prototype, dividing the images into two groups, and after, dividing each group into two classes.

The classification rule applied in level 1 and in level 2 are always the same. By examples, if in level 1 was applied SVM quadratic, than this same rule is applied in level 2 too.

3.4 Test Methodology

The used data set image is the same dataset used in [11]. It was manually obtained from a real video of an urban road, with three lanes. The dimension selected to vehicle images was 61 x 43, because it allows long vehicles (bus and truck) to be seen. For each of the four classes, 25 examples were acquired totalizing 100 vehicle images. These images were obtained from any of the road lanes and it occurred that in a vehicle image, part of another vehicle was shown. In Figure 3, two random images of each of the four classes were presented in sequence (motorcycle, car, bus and truck). There is neither centralization, nor illumination normalization.
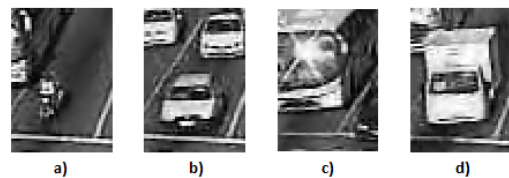

Fig. 3. Vehicle image examples (motor, car, bus and truck)

By leaving one image out, it was meant to divide the dataset into training and test. Let us assume that $x_{c,p}$ is a image of the class c in the pose p. Being all data set $D = \{x_{c,1}, x_{c,2}, ... x_{c,n}, \}$, with $c=\{1, ... , 4\}$ and $n=25$. When the test image is $x_{c,j}$, four images are removed of the dataset (one of each class) and the training set becomes $D' = D - \{x_{1,j}, x_{2,j}, x_{3,j}, x_{4,j}\}$. Four images are removed in order to maintain the number of representatives from each class equivalently.

All combination of six features are made, from 1 to 6, totalizing 60 sets of features: 6 sets with one feature, 15 sets with 2 features, 20 sets with 3 features, 12 sets with 4 features, 6 sets with 5 features and one set with 6 features. Each one of these 60 sets is used as input to six classifiers: prototype, NN, SVM with linear kernel, SVM with quadratic kernel, SVM with RBF kernel and SVM with polynomial kernel. Additionally, PCA is applied on the combination features, when more than one attribute is used (54 sets), and the first main component is used as input to the six classifiers.

The number of classified images in level 1 is always 100. In level 2, however, the number of classified image corresponds to the number of correct classification of the respective level 1. By example, if in level 1 80 images were classified correctly, then, in level 2, only these 80 images will be classified.

4   Results

In Figure 4, distribution graphs of the three selected features, considering all images, are presented. In a) and b), respectively, the data organized into two groups and into four classes are shown.
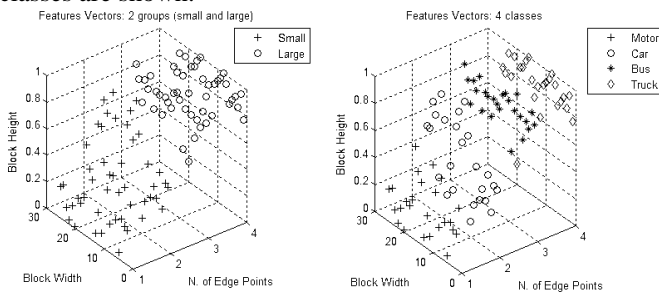

Fig. 4. Distribution graphs of selected features

In Figure 5, the twelve best classification results, related to level 1, are presented. These results, between 83% and 88%, are based on 100 classification tests. These best results include the accuracy of all 114 feature sets (60 sets without PCA and 54 sets with PCA) and six classifiers. The maximum accuracy (88%), achieved in 5 cases, are explained in Tables I and II.
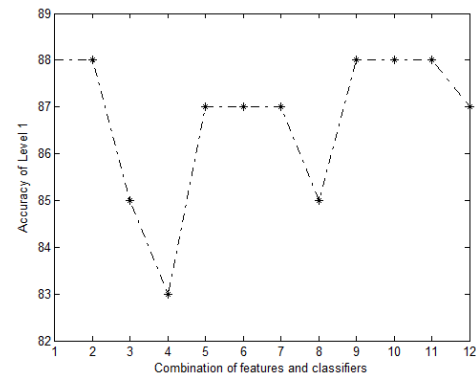

Fig. 5. Best classification accuracy of level 1

In Table I, the best results of level 1, of each classifier, are presented. Two results of each classifier are presented: the first one is associated with the set of features and the second one considers the set of PCA of the features. The L1 result column represents the number of right classification in 100 tests.

TABLE I.   BEST RESULTS OF LEVEL 1 CLASSIFICATION

Classifier	Features	L1 Results
Prototype	NP+W+BW	**88**
Prototype	PCA of (NP+W+F+BH)	**88**
SVM Linear	NP+H+F	87
SVM Linear	PCA of (H+F+BW)	87
SVM Quadratic	W+F+BW	**88**
SVM Quadratic	PCA of (W+H+F) or PCA of (NP+W+H+BW+BH)	87
SVM RBF	NP+W+F+BH	**88**
SVM RBF	PCA of (W+F+BW+BH)	**88**
SVM Polynomial	NP+BW+BH	87
SVM Polynomial	PCA (W+H+F)	85
NN	NP + BW	85
NN	PCA of (NP + BW)	86

Table II presents the results of level 2 classification associated with the best results of level 1, that in Table I are in bold. The L2 result column represents the number of right classification in 88 tests. As the objective of the method is the classification of vehicle image into four classes, the L2 result column is the final result because it also represents the final accuracy of 100 tests.

TABLE II.   BEST RESULTS OF LEVEL 2 CLASSIFICATION

Classifier	Features	L2 Results
Prototype	NP+W+BW	56
Prototype	PCA of (NP+W+F+BH)	**57**
SVM Quadratic	W+F+BW	56
SVM RBF	NP+W+F+BH	**64**
SVM RBF	PCA of (W+F+BW+BH)	**67**

Considering the best three results of Level 2, that in Table II are in bold, Table III presents the errors detailed by class, both level 1 and level 2. The columns M, C, T and B

represent the number of errors in the classification, respectively, of motorcycle, car, truck and bus.

TABLE III.   NUMBER OF ERRORS: CLASSIFICATION LEVEL 1 AND LEVEL 2

	Errors (Level 1)				Errors (Level 2)			
Classifier and Features	M	C	T	B	M	C	T	B
Prototype PCA (NP+W+F+BW)	1	9	1	1	7	7	11	7
SVM RBF NP+W+F+BH	1	7	2	2	3	3	9	9
SVM RBF PCA (W+F+BW+BH)	2	7	2	1	3	0	8	10

In Figure 6, the twelve images that were classified incorrectly in level 1, are presented, in the combination of the features NP+W+F+BW and a SVM classifier with RBF. It presents the original image (a) and its respective edge (b), after the pre-processing. In sequence, the errors presents refer to one motorcycle, seven cars, two buses and two trucks.

In the vehicle edge of the last image of Figure 6, there are considerable loss of information. This was caused by bands removal applied in the pre-processing stage.
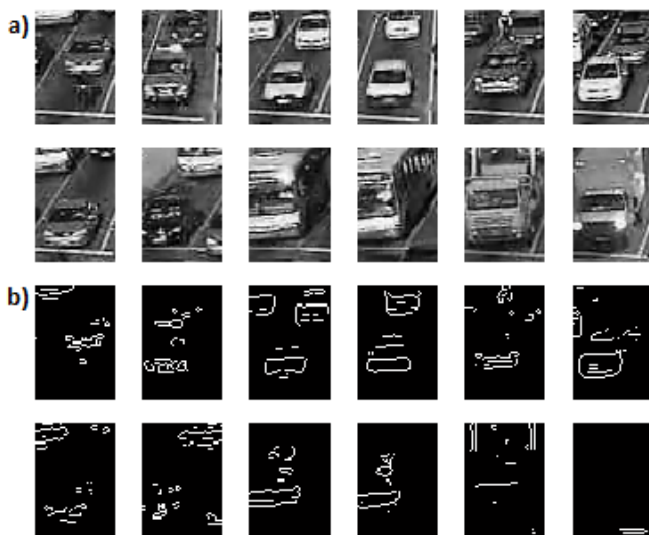

Fig. 6. Images classified incorrectly in level 1, features NP+W+F+BW, classifier SVM RBF :
a) Vehicle images   b) Vehicle edge

In Figure 7, the eighteen images that were classified incorrectly in level 2, bus and truck classes, are presented, in the combination of the features NP+W+F+BW and a SVM classifier with RBF. It presents the original image (a) and its respective edge (b), after the pre-processing. In sequence, the errors presents refer to nine buses and nine trucks.

Fig. 7. Images classified incorrectly in level 2, features NP+W+F+BW, classifier SVM RBF :
a) Vehicle images   b) Vehicle edge

5   Conclusion

Vehicle classification based on image obtained by traffic video and with separation into four classes does not have a linear separation frontier. Different combinations of features and classifiers achieve a numerical approximate accuracy and the errors occurred with similarity. This suggests that with the used feature selection, the maximum accuracy will not achieve nearby 100%. Furthermore, in this experiment good results are not associated only to PCA application.

Removal of bands is suitable because it allows removing objects that are not part of the main vehicle of the scene, for example, part of another vehicle. This completes segmentation which is usually fulfilled by detection methods. However, it may suggest that this process needs improvement based on the errors analyzed in the classification level 1 of the car class, because there is more than one vehicle in the image and this reflects in the image edge. Furthermore, there is another type of limitation of the technique associated with significant loss of information by eliminating bands.

The hierarchical classification is suitable firstly because it contributes to high accuracy in isolated classification phases. Secondly, because it reduces the processing time in level 2, since it removes some examples of the comparison process. In level 1 the classifier may be less specialized than SVM, as distance to prototype, but in level 2 a classifier with better separation rules, such SVM, reaches better results.

The final maximum results (67%), in this database, are relevant, even having achieved 100% in level 1, in [10], with only twenty images. Additionally, in this database, without hierarchical classification, 55% are achieved in [11].

The errors in level 1 classification are heavily concentrated in the small group related to the car type. These errors occur because many car images have more than one car; consequently the width and height of these images are not suitable features. The errors in level 2 were more representative in the large group. Such errors occur because in the bus and truck classes, the height and width of the edge have similarity.

The accuracy in level 2 is low and the differences between classifiers are few. So, the method, including pre-processing, features and classifiers need to be improved because, at present, the error tolerated in this application area is approximately 5%.

Future works include other features, such as texture and HOG, and different prototypes.

6   References

[1]   Angel Sanchez, P.D. Suarez., A. Conci, E.O. Nunes. "Video-Based Distance Traffic Analysis: Application to Vehicle Tracking and Counting". Computing in Science & Engineering, vol. 13, issue 3, 38-45, May-June 2011.

[2]   Jun-Wei Hsieh, Shih-Hao Yu, Yung-Sheng Chen, Wen-Fong Hu. "Automatic Traffic Surveillance System for Vehicle Tracking and Classification". IEEE Transactions on Intelligent Transportation Systems, vol. 7, issue 2, 175-187, June 2006.

[3]   Mehran Kafai, B. Bhanu. "Dynamic Bayesian Networks for Vehicle Classification in Video". IEEE Transactions on Industrial Informatics, vol. 8 , issue 1 , 100-109, Feb. 2012.

[4]   Wei Wang, Y. Shang, J. Guo, Z. Qian. "Real-Time Vehicle Classification Based on Eigenface". International Conference on Consumer Electronics, Communications and Networks (CECNet), 4292-4295, April 2011.

[5]   Brendan T. Morris, M.M. Trivedi. "Learning, Modeling and Classification of Vehicle Track Patterns from Live Video". IEEE Transaction on Intelligent Transportation Systems, vol. 9, issue 3, 425-437, Sept. 2008.

[6]   Andrew H.S. Lai, George S.K. Fung, Nelson H.C. Yung. "Vehicle Type Classification from Visual-based Dimension Estimation". IEEE Intelligent Transportation System, 201-206, Aug. 2001.

[7]   S. Pumrin, D.J. Dailey. "Vehicle Image classification via Expectation-Maximization Algorithm". International Symposium on Circuits and Systems (ISCAS), (II)468-471, May 2003.

[8]   Nilluthpol C. Mithun, N.U. Rashi, S.M.M. Rahman "Detection and Classification Using Multiple Time-Spatial Images". IEEE Transaction on Intelligent Transportation System, vol. 13, issue 3, 1215-1225, Sep. 2012.

[9]   Zhou Zhu-yu, D. Tian-min, Lv Xian-yang. "Study for Vehicle Recognition and Classification Based on Gabor Wavelets Transform & HMM". International Conference on Consumer Electronics, Communications and Networks (CECNet), 5272-5275, April 2011.

[10] Fabrízia .M.S. Matos, R.M.C.R. de Souza. "An Image Vehicle Classification Method Based on Edge and PCA Applied to Blocks". IEEE International Conference on Systems, Man, and Cybernetics (SMC), 1688-1693, Oct. 2012.

[11] Fabrízia.M.S. Matos, R.M.C.R. de Souza "Vehicle Image Classification Based on Edge: Features and Distances Comparison". International Conference on Neural Information Processing (ICONIP), LNCS, vol, 7666, 691-698, Nov. 2012.

[12] Ana M. Cretu, P. Payeur, R. Laganiere. "Salient Features Based on Visual Attention for Multi-View Vehicle Classification". IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA), 1-6, Sept. 2011.

[13] Zezhi Chen, T. Ellis, S.A. Velastin. "Vehicle Detection, Tracking and Classification in Urban Traffic". International IEEE Conference on Intelligent Transportation Systems, 951-956, Sept. 2012.

[14] <http://www.transportes.gov.br/conteudo/53887>. Accessed in 14/03/2013.

[15] Rafael C. Gonzalez, R. E. Woods, S. L. Eddins. "Digital Image Processing Using Matlab". Pearson Prentice Hall, 2004.

[16] Hossein Ebrahimpour-Komleh, V. Chandran, S. Sridharan. "Face Recognition Using Fractal Codes". International Conference on Image Processing, 58-61 vol. 3, Sept. 2001.

Specializing the Logic of Multiple-Valued Argumentation to the Jaina Seven-Valued Logic

Shogo Ohta
Graduate School of Science and Technology, Niigata University
Niigata, Japan
s-ohta@cs.ie.niigata-u.ac.jp

Hajime Sawamura
Institute of Science and Technology, Niigata University
Niigata, Japan
sawamura@ie.niigata-u.ac.jp

Takeshi Hagiwara
Institute of Science and Technology, Niigata University
Niigata, Japan
hagiwara@ie.niigata-u.ac.jp

Jacques Riche
Department of Computer Science, Katholieke Universiteit Leuven
Leuven, Belgium
riche@cs.kuleuven.ac.be

Abstract—**Argumentation is a dialectical process of knowing things (inquiry) and justifying them (advocacy) in general. Computational argumentation has been recognized as a social computing mechanism or paradigm in the multi-agent systems community. We have developed a computational argumentation framework that basically consists of EALP (Extended Annotated Logic Programming) and LMA (Logic of Multiple-valued Argumentation) constructed on top of EALP. EALP is a very generic knowledge representation language for uncertain arguments, and LMA built on top of it also yields a generic argumentation framework so that it allows agents to construct uncertain arguments under truth values specified depending on application domains.**

In this paper, we specialize such a generic argumentation system to an argumentation system that can deal with Eastern arguments based on the Jaina seven-valued logic. We illustrate this specialization using the implemented argumentation system: PIRIKA (Pilot of the Right Knowledge and Argument) based on EALP and LMA, which is now opened to the public as an open source software.

Keywords-**multiple-valued argumentation, neural net argumentation, syncretic argumentation, argument mining, argument animation**

I. Introduction

Argumentation is a dialectical process of knowing things (inquiry) and justifying them (advocacy) in general. In the last years, argumentation has been accepted as a promising social computing mechanism or paradigm in the multi-agent systems community [1]. It has proven to be particularly suitable for dealing with reasoning under incomplete or contradictory information in a dynamically changing and networked distributed environment.

We have developed a computational argumentation framework that basically consists of EALP and LMA [2]. EALP (Extended Annotated Logic Programming) is an expressive logic programming language we formalized for argumentation. The basic language constituents are literals associated with annotations as truth-values or epistemic states of agents. LMA is a Logic of Multiple-valued Argumentation constructed on top of EALP. It has three notions of negation to yield a momentum or driving force for argumentation. LMA is a generic logic of multiple-valued argumentation that allows us to specify various types of truth values depending on application domains, and to deal with uncertain arguments. Such a feature brings us the extensive applicability of LMA that is considered the most advantageous point in comparison to other approaches to argumentation [1].

In this paper, we specialize such a generic argumentation system to an argumentation system that can deal with Eastern arguments based on the Jaina seven-valued logic. We illustrate this specialization using the implemented argumentation system: PIRIKA (Pilot of the Right Knowledge and Argument) based on EALP and LMA, which is now opened to the public as an open source software.

The paper is organized as follows. In Section 2 and 3, we overview EALP and LMA as background of the paper. In Section 4, we present an overall picture of PIRIKA which provides the basic features and various auxiliary ones for standard uses. Section 5 is concerned with how to specialize PIRIKA so that it can deal with arguments based on Jaina seven-valued logic. Then, we illustrate two intriguing argument examples in which Jaina seven-valued logic may work well. The final section includes concluding remarks and future work.

II. Overview of EALP

EALP is an underlying knowledge representation language that we formalized for our logic of multiple-valued argumentation LMA. EALP has two kinds of explicit negation: Epistemic Explicit Negation '¬' and Ontological Explicit Negation '∼', and the default negation '**not**'. They

are supposed to yield a momentum or driving force for argumentation or dialogue in LMA. We here outline EALP.

A. Language

Definition 1: (**Annotation and annotated atoms**[3]). We assume a complete upper semi-lattice $(\mathcal{T}, \leq)$ of truth values, where $\leq$ denotes the semi-lattice ordering on $\mathcal{T}$. It is often convenient to assume the existence of a greatest element in $\mathcal{T}$, denoted $\top$. The least upper bound operator is denoted by $\sqcup$. An annotation is either an element of $\mathcal{T}$ (constant annotation), an annotation variable on $\mathcal{T}$, or an annotation term. Annotation term is defined recursively as follows: an element of $\mathcal{T}$ and annotation variable are annotation terms. In addition, if $t_1, \ldots, t_n$ are annotation terms, then $f(t_1, \ldots, t_n)$ is an annotation term. Here, f is a total continuous function of type $\mathcal{T}^n \to \mathcal{T}$. If A is an atomic formula and μ is an annotation, then $A : \mu$ is an annotated atom. We assume an annotation function $\neg : \mathcal{T} \to \mathcal{T}$, and define that $\neg(A : \mu) = A : (\neg\mu)$. $\neg A : \mu$ is called the epistemic explicit negation(e-explicit negation) of $A : \mu$.

Definition 2: (**Annotated literals**). Let $A : \mu$ be an annotated atom. Then $\sim (A : \mu)$ is the ontological explicit negation (o-explicit negation) of $A : \mu$. An annotated objective literal is either $\sim A : \mu$ or $A : \mu$. The symbol $\sim$ is also used to denote complementary annotated objective literals. Thus $\sim\sim A : \mu = A : \mu$. If L is an annotated objective literal, then **not** L is a default negation of L, and called an annotated default literal. An annotated literal is either of the form **not** L or L.

Definition 3: (**Extended Annotated Logic Programs (EALP)**). An extended annotated logic program (*EALP*) is a set of annotated rules of the form: $H \leftarrow L_1 \& \ldots \& L_n$, where H is an annotated objective literal, and L_i ($1 \leq i \leq n$) are annotated literals in which the annotation is either a constant annotation or an annotation variable.

For simplicity, we assume that a rule with annotation variables or objective variables represents every ground instance of it. In this assumption, we restrict ourselves to constant annotations in this paper since every annotation term in the rules can evaluate to the elements of $\mathcal{T}$. We identify a distributed EALP with an *agent*, and treat a set of EALPs as a *multi-agent system*.

B. Interpretation

Definition 4: (**Extended annotated Herbrand base**). The set of all annotated literals constructed from an EALP P on a complete upper semi-lattice $\mathcal{T}$ of truth values is called the extended annotated Herbrand base $H_P^{\mathcal{T}}$.

Definition 5: (**Interpretation**). Let $\mathcal{T}$ be a complete upper semi-lattice of truth values, and P be an EALP. Then, the interpretation on P is the subset $I \subseteq H_P^{\mathcal{T}}$ of the extended annotated Herbrand base $H_P^{\mathcal{T}}$ of P such that for any annotated atom A,

1) If $A : \mu \in I$ and $\rho \leq \mu$, then $A : \rho \in I$ (downward heredity);
2) If $A : \mu \in I$ and $A : \rho \in I$, then $A : (\mu \sqcup \rho) \in I$ (tolerance of difference);
3) If $\sim A : \mu \in I$ and $\rho \geq \mu$, then $\sim A : \rho \in I$ (upward heredity).

The conditions 1 and 2 of Definition 5 reflect the definition of the ideal of a complete lattice of truth values. The ideals-based semantics was first introduced for the interpretation of GAP by Kifer and Subrahmanian [3]. Our EALP for argumentation also employs this since it was shown that the general semantics with ideals is more adequate than the restricted one simply with a complete lattice of truth values [2]. We define three notions of inconsistencies corresponding to three concepts of negation in EALP.

Definition 6: (**Inconsistency**). Let I be an interpretation. Then,

1) $A : \mu \in I$ and $\neg A : \mu \in I \Leftrightarrow I$ is epistemologically inconsistent (e-inconsistent).
2) $A : \mu \in I$ and $\sim A : \mu \in I \Leftrightarrow I$ is ontologically inconsistent (o-inconsistent).
3) $A : \mu \in I$ and **not** $A : \mu \in I$, or $\sim A : \mu \in I$ and **not** $\sim A : \mu \in I \Leftrightarrow I$ is inconsistent in default (d-inconsistent).

When an interpretation I is o-inconsistent or d-inconsistent, we simply say I is *inconsistent*. We do not see the e-inconsistency as a problematic inconsistency since by the condition 2 of Definition 5, $A : \mu \in I$ and $\neg A : \mu = A : \neg\mu \in I$ imply $A : (\mu \sqcup \neg\mu) \in I$ and we think $A : \mu$ and $\neg A : \mu$ are an acceptable differential. Let I be an interpretation such that $\sim A : \mu \in I$. By the condition 1 of Definition 5, for any ρ such that $\rho \geq \mu$, if $A : \rho \in I$ then I is o-inconsistent. In other words, $\sim A : \mu$ rejects all recognitions ρ such that $\rho \geq \mu$ about A. This is the underlying reason for adopting the condition 3 of Definition 5. These notions of inconsistency yield a logical basis of attack relations described in the multiple-valued argumentation of Section III.

Definition 7: (**Satisfaction**). Let I be an interpretation. For any annotated objective literal H and annotated literal L and L_i, we define the satisfaction relation denoted by '$\models$' as follows.

- $I \models L \Leftrightarrow L \in I$
- $I \models L_1 \& \cdots \& L_n \Leftrightarrow I \models L_1, \ldots, I \models L_n$
- $I \models H \leftarrow L_1 \& \cdots \& L_n \Leftrightarrow I \models H$ or $I \not\models L_1 \& \cdots \& L_n$

III. OVERVIEW OF LMA

In formalizing logic of argumentation, the most primary concern is the rebuttal relation among arguments since it yields a cause or a momentum of argumentation. The rebuttal relation for two-valued argument models is most simple, so that it naturally appears between the contradictory

propositions of the form A and $\neg A$. In case of multiple-valued argumentation based on EALP, much complication is to be involved into the rebuttal relation under the different concepts of negation. One of the questions arising from multiple-valuedness is, for example, how a literal with truth-value ρ confronts with a literal with truth-value μ in the involvement with negation. In the next subsection, we outline important notions proper to logic of multiple-valued argumentation LMA in which the above question is reasonably solved.

A. Annotated arguments

Definition 8: (**Reductant and Minimal reductant**).
Suppose P is an EALP, and C_i $(1 \leq i \leq k)$ are annotated rules in P of the form: $A : \rho_i \leftarrow L_1^i \& \ldots \& L_{n_i}^i$, in which A is an atom. Let $\rho = \sqcup\{\rho_1, \ldots, \rho_k\}$. Then the following annotated rule is a reductant of P.

$A : \rho \leftarrow L_1^1 \& \ldots \& L_{n_1}^1 \& \ldots \& L_1^k \& \ldots \& L_{n_k}^k$.

A reductant is called a minimal reductant when there does not exist non-empty proper subset $S \subset \{\rho_1, \ldots, \rho_k\}$ such that $\rho = \sqcup S$

Definition 9: (**Truth width [3]**). A lattice $\mathcal{T}$ is n-wide if every finite set $E \subseteq \mathcal{T}$, there is a finite subset $E_0 \subseteq E$ of at most n elements such that $\sqcup E_0 = \sqcup E$.

The notion of truth width is for limiting the number of reductants to be considered in argument construction. For example, the complete lattice $\mathcal{FOUR} = (\{\perp, \mathbf{t}, \mathbf{f}, \top\}, \leq)$, where $\forall x, y \in \{\perp, \mathbf{t}, \mathbf{f}, \top\}$ $x \leq y \Leftrightarrow x = y \vee x = \perp \vee y = \top$, is 2-wide, and the complete lattice $(\Re[0, 1], \leq)$ of the unit interval of real numbers is 1-wide.

Definition 10: (**Annotated arguments**). Let P be an EALP. An annotated argument in P is a finite sequence $Arg = [r_1, \ldots, r_n]$ of rules in P such that for every i $(1 \leq i \leq n)$,

1) r_i is either a rule in P or a minimal reductant in P.
2) For every annotated atom $A : \mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $A : \rho$ $(\rho \geq \mu)$ is head of r_k.
3) For every o-explicit negation $\sim A : \mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $\sim A : \rho$ $(\rho \leq \mu)$ is head of r_k.
4) There exists no proper subsequence of $[r_1, \ldots, r_n]$ which meets from the first to the third conditions, and includes r_1.

We denote the set of all arguments in P by $Args_P$, and define the set of all arguments in a set of EALPs $MAS = \{KB_1, \ldots, KB_n\}$ by $Args_{MAS} = Args_{KB_1} \cup \cdots \cup Args_{KB_n}$ $(\subseteq Args_{KB_1 \cup \cdots \cup KB_n})$.

B. Attack relation

Definition 11: (**Rebut**). Arg_1 rebuts $Arg_2 \Leftrightarrow$ there exists $A : \mu_1 \in concl(Arg_1)$ and $\sim A : \mu_2 \in concl(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A : \mu_1 \in concl(Arg_1)$ and $A : \mu_2 \in concl(Arg_2)$ such that $\mu_1 \leq \mu_2$.

Definition 12: (**Undercut**). Arg_1 undercuts $Arg_2 \Leftrightarrow$ there exists $A : \mu_1 \in concl(Arg_1)$ and **not** $A : \mu_2 \in assm(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A : \mu_1 \in concl(Arg_1)$ and **not** $\sim A : \mu_2 \in assm(Arg_2)$ such that $\mu_1 \leq \mu_2$.

Definition 13: (**Strictly undercut**). Arg_1 strictly undercuts $Arg_2 \Leftrightarrow Arg_1$ undercuts Arg_2 and Arg_2 does not undercut Arg_1.

Definition 14: (**Defeat**). Arg_1 defeats $Arg_2 \Leftrightarrow Arg_1$ undercuts Arg_2, or Arg_1 rebuts Arg_2 and Arg_2 does not undercut Arg_1.

When an argument defeats itself, such an argument is called a *self-defeating argument*. For example, $[p : \mathbf{t} \leftarrow \mathbf{not}\, p : \mathbf{t}]$ and $[q : \mathbf{f} \leftarrow \sim q : \mathbf{f}, \sim q : \mathbf{f}]$ are all self-defeating. In this paper, however, we rule out self-defeating arguments from argument sets since they are in a sense abnormal, and not entitled to participate in argumentation or dialogue. In this paper, we employ defeat and strictly undercut to specify the set of justified arguments where d stands for defeat and su for strictly undercut.

Definition 15: (**acceptable and justified argument** [4]). Suppose $Arg_1 \in Args$ and $S \subseteq Args$. Then Arg_1 is acceptable wrt. S if for every $Arg_2 \in Args$ such that $(Arg_2, Arg_1) \in d$ there exists $Arg_3 \in S$ such that $(Arg_3, Arg_2) \in su$. The function $F_{Args, d/su}$ mapping from $\mathcal{P}(Args)$ to $\mathcal{P}(Args)$ is defined by $F_{Args, d/su}(S) = \{Arg \in Args \mid Arg$ is acceptable wrt. $S\}$. We denote a least fixpoint of $F_{Args, d/su}$ by $J_{Args, d/su}$. An argument Arg is justified if $Arg \in J_{d/su}$; an argument is overruled if it is attacked by a justified argument; and an argument is defensible if it is neither justified nor overruled.

Since $F_{x/y}$ is monotonic, it has a least fixpoint, and can be constructed by the iterative method [4]. Justified arguments can be dialectically determined from a set of arguments by the dialectical proof theory. We give the sound and complete dialectical proof theory for the abstract argumentation semantics $J_{Args, x/y}$.

Definition 16: (**dialogue** [5]).
An dialogue is a finite nonempty sequence of moves $move_i = (Player_i, Arg_i)$, $(i \geq 1)$ such that

1) $Player_i = P$ (*Proponent*) $\Leftrightarrow i$ is odd; and $Player_i = O$ (*Opponent*) $\Leftrightarrow i$ is even.
2) If $Player_i = Player_j = P$ $(i \neq j)$ then $Arg_i \neq Arg_j$.
3) If $Player_i = P$ $(i \geq 3)$ then $(Arg_i, Arg_{i-1}) \in su$; and if $Player_i = O$ $(i \geq 2)$ then $(Arg_i, Arg_{i-1}) \in d$.

In this definition, it is permitted that $P = O$, that is a dialogue is done by only one agent. Then, we say such an argument is a self-argument.

Definition 17: (**dialogue tree** [5]). A dialogue tree is a tree of moves such that every branch is a dialogue, and for all moves $move_i = (P, Arg_i)$, the children of $move_i$ are all those moves (O, Arg_j) $(j \geq 1)$ such that $(Arg_j, Arg_i) \in d$.

We have the sound and complete dialectical proof theory for the argumentation semantics $J_{Args,x/y}$ [2].

IV. Standard Uses of PIRIKA

PIRIKA[1] is an implemented system of EALP/LMA [6]. It is now open to the public as downloadable OSS together with video clips and operation's and users' manual. http://www.cs.ie.niigata-u.ac.jp/Research/PIRIKA/PIRIKA.html

The argumentation scenario of PIRIKA basically consists of the following phases:

- Registering agents (as avatars of humans) with the argument server so that they can commit to argumentation
- Preparing a lattice of truth values for dealing with uncertainty depending on application domains
- Designing knowledge bases under the specified truth values in terms of EALP
- Starting argumentation on submitted issues/claims in LMA (see Figure 1 for the system architecture)
- Visualizing the live argumentation process and diagramming arguments
- Determining the status of an argument
- Storing arguments and their results in the argument repository for the future reuse

In addition, many other unique features proper to the logic of multiple-valued argumentation is integrated with the core part of PIRIKA. They are,

- Neural network argumentation for Dungean semantics [7]
- Pluralistic argumentation (Western and Eastern arguments) [8]
- Syncretic argumentation [9]
- Argument mining [10]
- Argument animation [11]

The overall architecture of PIRIKA is shown in Figure 1.

V. Jaina seven-valued logic

In this subsection, we deal with Jaina seven-valued logic [12][13], which is to be captured as an upper semi-lattice structure of the EALP/LMA framework.

The Jaina logic is said to be an intellectual ahimsa in a word [12], and its doctrines consist of Anekāntavāda, Syādvāda and Nayavāda [14]. Anekāntavāda is one of the most important and fundamental doctrines of Jainism. It refers to the principles of pluralism and multiplicity of viewpoints, the notion that truth and reality are perceived differently from diverse points of view, and that no single point of view is the complete truth.

Syādvāda is the theory of conditioned predication, which provides an expression to anekānta by recommending that

[1]An acronym for PIlot for the RIght Knowledge and Argument

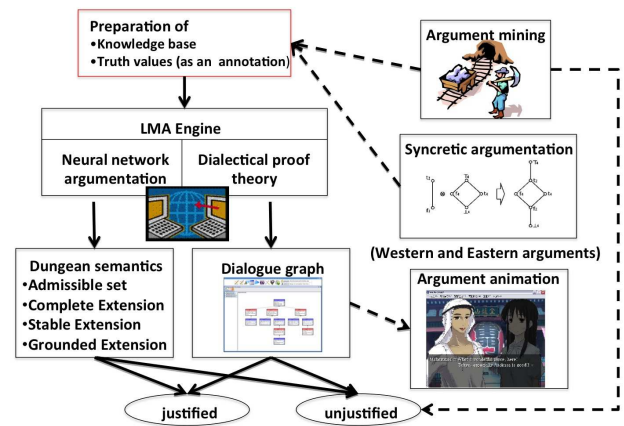**An overall picture of LMA-based argumentation system**

Figure 1.   Architecture of PIRIKA

the epithet Syād be prefixed to every phrase or expression, and Nayavāda is the theory of partial standpoints.

Syād means 'in some ways', 'from a perspective', 'in some aspect', 'somehow', 'maybe', etc. As reality is complex, no single proposition can express the nature of reality fully. Thus the term 'syāt'(in composition 'syād') should be prefixed before each proposition giving it a conditional point of view and thus removing any dogmatism in the statement. Since it ensures that each statement is expressed from seven different conditional and relative viewpoints or propositions, syādvāda is known as the theory of seven conditioned predications. These seven propositions are:

1) syād-astiin some ways, it is.
2) syād-nāstiin some ways, it is not.
3) syād-avaktavyaḥin some ways, it is indescribable.
4) syād-asti-nāstiin some ways, it is, and it is not.
5) syād-asti-avaktavyaḥin some ways, it is, and it is indescribable.
6) syād-nāsti-avaktavyaḥin some ways, it is not, and it is indescribable.
7) syād-asti-nāsti-avaktavyaḥin some ways, it is, it is not, and it is indescribable.

Each of these seven propositions examines the complex and multifaceted nature of reality from a relative point of view of time, space, substance and mode. To ignore the complexity of reality is to commit the fallacy of dogmatism [14].

The vulgar (Aristotelian or Boolean) logic is based on the 'Laws of Thought.' The Jain theory of modes of truth (sapt-abhangivada, 'seven-division-ism,' perfected by the sixth-century Samantabhadra) recognizes seven truth-values [12].

We relate those seven propositions (or the seven modes of truth) to the seven truth-values : t, f̵, i, tf̵, ti, f̵i, tf̵i respectively. Then, we can well capture the structure of the seven truth-values of Jaina logic as the upper semi-lattice as seen in Figure 2, that is,

$$\mathcal{JAINA} = \langle\{\mathfrak{t}, \mathfrak{f}, \mathfrak{i}, \mathfrak{tf}, \mathfrak{ti}, \mathfrak{fi}, \mathfrak{tfi}\}, \leq\rangle.$$
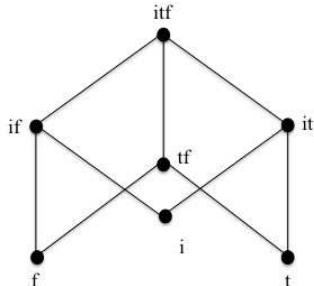
Figure 2.   The upper semi-lattice of seven truth-values in Jaina logic

Figure 3.   Western and Eastern arguments against Aristotle

In what follows, we present argument examples in which intriguing language and logic phenomena can be captured on the basis of the Jaina seven-valued logic.

A. A pluralistic or multicultural argument example

We illustrate a pluralistic or multicultural argument by specializing LMA (Logic of Multiple-valued Argumentation) [2] to the upper semi-lattice $\mathcal{JAINA}$ in Figure 2. Let us consider the Western and Eastern arguments against Aristotle. Aristotle believed that the heavier a body is, the faster it falls to the ground. We simply represent this as $aristotle\_hyp : \mathfrak{t}$. Galileo's logical argument against this proceeds as follows: "Suppose that we have two bodies, a heavy one called H and a light one called L. Under Aristotle's assumption, H will fall faster than L. Now suppose that H and L are joined together. Now what happens? Well, L plus H is heavier than H so by the initial assumption it should fall faster than H alone. But in the joined body, L is lighter and will act as a 'brake' on H, and L plus H will fall slower than H alone. Hence it follows from the initial assumption that L plus H will both faster and slower than H alone. Since this is absurd, the initial assumption must be false." On the other hand, Easterners prefer a more holistic or dialectical argument like this: "Aristotle is based on a belief that the physical object is free from any influences of other contextual factors, which is impossible in reality." [15]

These are well translated into EALP [2] as follows:
[Galileo's knowledge]
$\sim aristotle\_hyp : \mathfrak{t} \leftarrow faster(l + h, h) : \mathfrak{tf}$
$faster(l + h, h) : \mathfrak{t} \leftarrow not\ aristotle\_hyp : \mathfrak{f}$
$faster(l + h, h) : \mathfrak{f} \leftarrow slower(l + h, h) : \mathfrak{t}$
$slower(l + h, h) : \mathfrak{t} \leftarrow brake(l, h) : \mathfrak{t}$
$brake(l, h) : \mathfrak{t}$
[Eastern agent's knowledge]
$\sim aristotle\_hyp : \mathfrak{t} \leftarrow distrust\_decontextualization : \mathfrak{t}$
$distrust\_decontextualization : \mathfrak{t}$

Figure 3 depicts a dialogue tree constructed with the dialectical proof theory for EALP/LMA [2]. Obviously, Aristotle's argument $A_{Aristotle}$ is defeated (rebut) by Galileo's
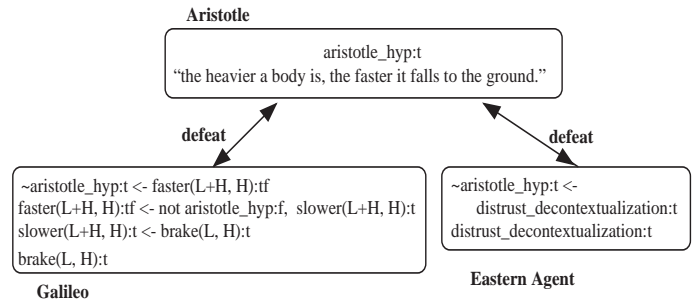
argument $A_{Western}$ and an Easterner's argument $A_{Eastern}$, and turns out not to be justified by two culturally different kinds of counter-arguments (actually defensible): an Western analytic argument and an Eastern holistic one, where the second rule in Galileo's argument is a reductant [3] made from his knowledge base. Note that Galileo made his argument by reductio ad absurdum for which the default negation 'not' has a crucial role in the rule representation. Furthermore, we note that the head $\sim aristotle\_hyp : \mathfrak{t}$ in the first rule of Galileo's argument does not undercut the assumption $\sim aristotle\_hyp : \mathfrak{f}$ of the second rule, that is, Galileo's argument is coherent or not self-defeating, and Eastern agent does not undercut the assumption $\sim aristotle\_hyp : \mathfrak{f}$ of the second rule in Galileo's argument. (Interested readers should refer to [2] for the technical terms used.)

In this example, all the arguments by Aristotle, Galileo and Eastern agent become defensible. Incidentally, let us consider a little modified version of the example. We first change Aristotle's belief as follows:
[Aristotle's belief]
$aristotle\_hyp : \mathfrak{t} \leftarrow not \sim empirically\_factual : \mathfrak{t}$.
And we make one more agent appear on the stage, who is a modern scientist having a firm belief on verificationism.
[Modern scientist's knowledge]
$\sim empirically\_factual : \mathfrak{t} \leftarrow$
    $not\ scientifically\_verified : \mathfrak{t}$.
Then, it is obvious that Aristotle's argument is overruled, and Galileo's, a modern scientist's ones and Eastern agent's one are all justified(see Figure 4).

B. Ethical argument example

We take up an ethical question 'Is homicide evil?' The knowledge bases of two agents: $A1$ and $CA1$ for the argument are shown in Table I. Agent A1 says '$homicide\_is\_evil :: [t\_i]$' with the ground "$when\_we\_say\_no\_we\_mean\_no :: [i]$". Then $[t\_i]$ means 'we cannot explain it, but it is so'. He also believes '$homicide\_is\_evil :: [t]$' with the definite ground. However, both of the assertions turn out to be defeated by the other Agent $CA1$. Agent $CA1$ has a wealth of knowledge compared with A1, quoting the famous words of Charlie Chaplin

Figure 4.   Galileo's argument is justified.

Figure 6. $\sim homicide\_is\_evil : ti \ \ is \ justified$

and Georg Jellinek, and exploiting that there is a scene where homicide is permitted. Furthermore, he also has such a unique assertion that '$millions\_murder\_makes\_a\_hero ::$ $[i\_t\_f]$'. $[i\_t\_f]$ means that if it is stated from a viewpoint of the meaning of the word, it is so, but if it is stated from a viewpoint of one homicide, it is not so, and if it is stated from a viewpoint of a hero's definition, it is indescribable. Actually it is a statement that gets involved in three perspectives.

The arguments on the issues '$\sim homicide\_is\_evil ::$ $[i\_t]$' and '$\sim homicide\_is\_evil :: [t]$' are justified since they are defeated (rebut) from $A1$, but $A1$'s ground are defeated (undercut) too. We can see the winning dialogue trees in Figure 5 and 6 respectively.

Figure 5. $\sim homicide\_is\_evil : t \ \ is \ justified$

Figure 7 displays the result of the neural network argumentation [7] computing Dungean semantics [16] of the Jaina argument example by PIRIKA. It includes the results of other argumentation semantics such as the admissible extention, stable extention and complete extention as well as the grounded extention. The neural network argumentation has such an advantage that it can deal with all the basic Dungean semantics and compute in a uniform way.

Figure 7.   Computing other argumentation semantics by the neural network

VI.   Related Work

PIRIKA is only one argumentation system that allows uncertainty in computational argument system in a full-fledged manner, and is integrated with such unique features as neural network argumentation, syncretic argumentation, argument mining, and argument animation. Chesñevar's possibilistic argumentation might be only one exception, but its argumentation model deals with uncertainty over the real numbers only [17].

VII.   Concluding Remarks and Future Work

EALP is a very generic knowledge representation language for uncertain arguments, and LMA built on top of it also yields a generic argumentation framework so that it allows agents to construct uncertain arguments under truth values specified depending on application domains.

Table I
KNOWLEDGE-BASE OF SEVEN-VALUED ARGUMENTATION

agent A1 knowledge base : KB_{A1}
$homicide\_is\_evil :: [it] <== not \sim when\_we\_say\_no\_we\_mean\_no :: [i].$
$homicide\_is\_evil :: [t] <== homicide\_is\_criminal :: [t]\&not \sim violation\_of\_law\_is\_evil :: [t].$
$homicide\_is\_criminal :: [t] <== true.$
agent CA1 knowledge base : KB_{CA1}
$\sim when\_we\_say\_no\_we\_mean\_no :: [i] <==$
$\qquad not \sim anything\_are\_pardonable\_in\_case\_of\_emergency :: [t].$
$\sim violation\_of\_law\_is\_evil :: [t] <== not \sim law\_is\_ethical\_minimum :: [t].$
$\sim violation\_of\_law\_is\_evil :: [t] <== law\_has\_no\_sense\_of right\_or\_wrong :: [t].$
$the\_law\_have\_no\_sense\_of right\_or\_wrong :: [t] <== true.$
$\sim homicide\_is\_evil :: [it] <== numbers\_sanctify\_murder :: [it].$
$numbers\_sanctify\_murder :: [it] <== not \sim killing\_millions\_makes\_a\_hero :: [itf].$
$\sim homicide\_is\_evil :: [t] <== some\_country\_allows\_die\_in\_a\_duel :: [t].$
$some\_country\_allows\_die\_in\_a\_duel :: [t] <== true.$

In this paper we manifested the standard uses of PIRIKA, an implemented system of EALP/LMA, and revealed its potential usefulness by specializing it to the Jaina seven-valued logic. Particularly, we showed that PIRIKA can deal with not only Western arguments but also Eastern arguments such as somewhat complicated Indian Jaina logic.

In the near future, we will port PIRIKA on Linux, Windows and Mac OX to pervasive personal tools such as iPhone and iPad, in order to attract a wide range of people and allow them to use PIRIKA in their daily lives. We hope that such an attempt will open up a new horizon for computational argumentation research.

REFERENCES

[1] I. Rahwan and G. R. E. Simari, *Argumentation in Artificial Intelligence*. Springer, 2009.

[2] T. Takahashi and H. Sawamura, "A logic of multiple-valued argumentation," in *Proceedings of the third international joint conference on Autonomous Agents and Multi Agent Systems (AAMAS'2004)*. ACM, 2004, pp. 800–807.

[3] M. Kifer and V. S. Subrahmanian, "Theory of generalized annotated logic programming and its applications," *J. of Logic Programming*, vol. 12, pp. 335–397, 1992.

[4] P. M. Dung, "An argumentation semantics for logic programming with explicit negation," in *Proc. of 10th Int. Conference on Logic Programming*, 1993, pp. 616–630.

[5] H. Prakken and G. Sartor, "Argument-based extended logic programming with defeasible priorities," *J. of Applied Non-Classical Logics*, vol. 7, no. 1, pp. 25–75, 1997.

[6] S. Tannai, S. Ohta, T. Hagiwara, H. Sawamura, and J. Riche, "The state of the art in the development of a versatile argumentation system based on the logic of multiple-valued argumentation," in *Proc. of International Conference on Agents and Artificial Intelligence (ICAART2013)*. INSTICC, 2013, pp. 217–224.

[7] Y. Goto, T. Hagiwara, and H. Sawamura, "Neural networks computing the dungean semantics of argumentation," in *IJCCI (NCTA)*, K. Madani, J. Kacprzyk, and J. Filipe, Eds. SciTePress, 2011, pp. 5–14.

[8] H. Sawamura, T. Takahashi, and K. Matsunaga, "An eastern specialization of logic of multiple-valued argumentation to tetralemma originated in india," in *Proceedings of the 2nd Indian International Conference on Artificial Intelligence (IICAI 2005)*, B. Prasad, Ed., 2005, pp. 1274–1291.

[9] Y. Maruyama, T. Hasegawa, T. Hagiwara, and H. Sawamura, "Syncretic argumentation for multi-agents by lattice homomorphism, fusion and sum," in *ArgMAS*, ser. Lecture Notes in Computer Science, P. McBurney, S. Parsons, and I. Rahwan, Eds., vol. 7543. Springer, 2011, pp. 46–65.

[10] Y. Seino, H. Sawamura, and T. Hagiwara, "Towards argument mining," in *Proceedings of 2011 International Conference on Data Engineering and Internet Technology (DEIT 2011)*. IEEE, 2011, pp. 27–34.

[11] T. Narita and H. Sawamura, "An attempt to argument metamorphosis from symbols to moving images," in *Proceedings of 2008 Joint Agent Symposium (JAWS2008)*, 2008.

[12] G. B. Burch, "Seven-valued logic in Jain philosophy," *International Philosophical Quarterly: An Intercultural Forum*, vol. 4, no. 1, pp. 68–93, 1964.

[13] J. Ganeri, "Jaina logic and the philosophical basis of pluralism," *History and Philosophy of Logic*, vol. 23, pp. 267–281, 2002.

[14] Wikipedia, "Jainism," *http://en.wikipedia.org/wiki/Jainism*, 2013.

[15] R. E. Nisbett, *The Geography of Thought: How Asians and Westerners Think Differently... and Why*. The Free Press, 2003.

[16] P. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logics programming and n-person games," *Artificial Intelligence*, vol. 77, pp. 321–357, 1995.

[17] C. I. Chesñevar, G. Simari, T. Alsinet, and L. Godo, "A Logic Programming Framework for Possibilistic Argumentation with Vague Knowledge," in *Proc. of the Intl. Conference on Uncertainty in Artificial Intelligence (UAI2004)*, 2004.

Relative Entropy Minimizing-Based Theory of Intelligent Systems[1]

"ICAI: LATE BREAKING PAPER"

Guangcheng Xi

(State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190,P.R.China)

Abstract: Based on the point of view of neuroethology and cognition-psychology, general frame of theory for intelligent systems is presented by means of principle of relative entropy minimizing in this paper. Cream of the general frame of theory is to present and to prove basic principle of intelligent systems: entropy increases or decreases together with intelligence in the intelligent systems. The basic principle is of momentous theoretical significance and practical significance .From the basic principle can not only derive two kind of learning algorithms (statistical simulating annealing algorithms and annealing algorithms of mean-field theory approximation) for training large kinds of stochastic neural networks,but also can thoroughly dispel misgivings created by second law of thermodynamics on $people's$ psychology ,hence make one be fully confident of facing life.Because of Human society, natural world, and even universe all are intelligent systems.

Key words: Complex systems; Intelligent systems; Relative entropy minimizing; General frame of theory; Basic principle.

1 Introduction

Until now, human brain is the sophisticated creation of natural world and is only place where are produced cognition and intelligence, and hence spirit. Human being as a body carrying life is the most advanced and perfect intelligent system on the planet. It has the features common to general complex system. As we see it[1]， a complex system is a functional system having any structure(including hierarchical and variably hierarchical structure) and consisting of any number of subsystems(≥ 1)capable of particular functions. These subsystems form closed loops of their own based on the feedback mechanism and are capable of self-adjustment according to different optimum criteria and interact with each other in various ways, and are composed of various dynamical, logical, nonlogical, and heuristic links. A complex system integrates at least the following six aspects:

(1) Multi-dimensionality: magnitude of dimension, type of dimension ,and transition each other of the types of dimension;

(2) Multi-parameter;

(3) Multi-relationship: relationships between(among)variables at same level(in particular, nonlinear relationships) and relationships between(among)variables at different level(in particular, nonlinear relationships)or the crossover of the two relationships;

(4) Multi-criterion: multi-component, multi-scales(levels);

---

[1] Having revisited content of the paper "Intelligent control with relative entropy minimizing（Control theory and applications，Vol.16,No.1,PP－27-31.In Chinese.）" ,I have proposed and proved the basic principle of intelligent system.And then further having supplemented and perfected,this article is formed.It is submitted first time in English.

(5) Multi-functionality: emergence, co-operation, competition, adaption, closed-openness, and so on;

(6) Multi-discipline of knowledge used.

Obviously, an intelligent system is complex system.

Intelligent control is the process of performing a task by an intelligent machine[2]. The theory of intelligent control is a composition of mathematic, lingual method and algorithm used in the system and the process and is based on cross disciplines of neuroethology, cognition-psychology, computer science, systematic science, artificial intelligence and information science. Only when we are accustomed   both to introduction and effective use of all knowledge of modern science , in particular, of the systematic science in   scientific fields related to "human being" and to introduction and effective use of all scientific knowledge related to "human bring" in other modern science, in particular ,in the fields of systematic science, in other words, only when human being has full knowledge of itself, can an intelligent control system in true sense be obtained and can an the theory of intelligent system in true sense be established.

Based on the point of view of neuroethology and cognition-psychology, general frame of theory for intelligent systems is presented by means of principle of relative entropy minimizing in this paper. Cream of the general frame of theory is to present and to prove basic principle of intelligent systems: entropy increases or decreases together with intelligence in the intelligent systems. The basic principle is of momentous theoretical significance and practical significance .From the basic principle can not only derive two kind of learning algorithms (statistical simulating annealing algorithms and annealing algorithms of mean-field theory approximation) for training large kinds of stochastic neural networks,but also can thoroughly

dispel misgivings created by second law of thermodynamics on $people's$ psychology ,hence

make one be fully confident of facing life.Because of Human society, natural world, and even universe all are intelligent systems.Human intelligence is in the Brain, intelligence of universe is in Black holes. In particular,highest intelligence of   universe is in the black hole which possesses maximum volume in all black holes.

This paper is organized as follows:Section 2 presents the basic viewpoint and basic construction on intelligent systems,gives block diagram of intelligent control system.Section 3 presents and proves basic   principle of intelligent system: entropy increases or decreases together with intelligence in the intelligent system.Sectiion 4   demonstrates how derive two kind of learning algorithms (statistical simulating annealing algorithms and annealing algorithms of mean-field theory approximation) for training large kinds of stochastic neural networks by means of the basic principle of the intelligent system.

2   Basic viewpoint and basic construction on intelligent systems

The following is a block diagram of intelligent system from the perceptive of neuroethology and cognition-psychology.

It is a description of an abstracted framework of common intelligent control system, including the description of the hierarchical intelligent control system. We will give a brief description of the block diagram of intelligent control system shown in figure1.

The perception system makes characteristic abstraction, transformation, integration and coding of environmental stimuli. The coded environmental stimuli are input to the memory system. The memory is categorized as short-term memory and long-term memory. Information

processing by short-term memory comes from two sources. On one hand, it processes the coded information from environment. At this time, the short-term memory shows up great selectivity: elimination of the unwanted or minor information and preservation and further acknowledgment
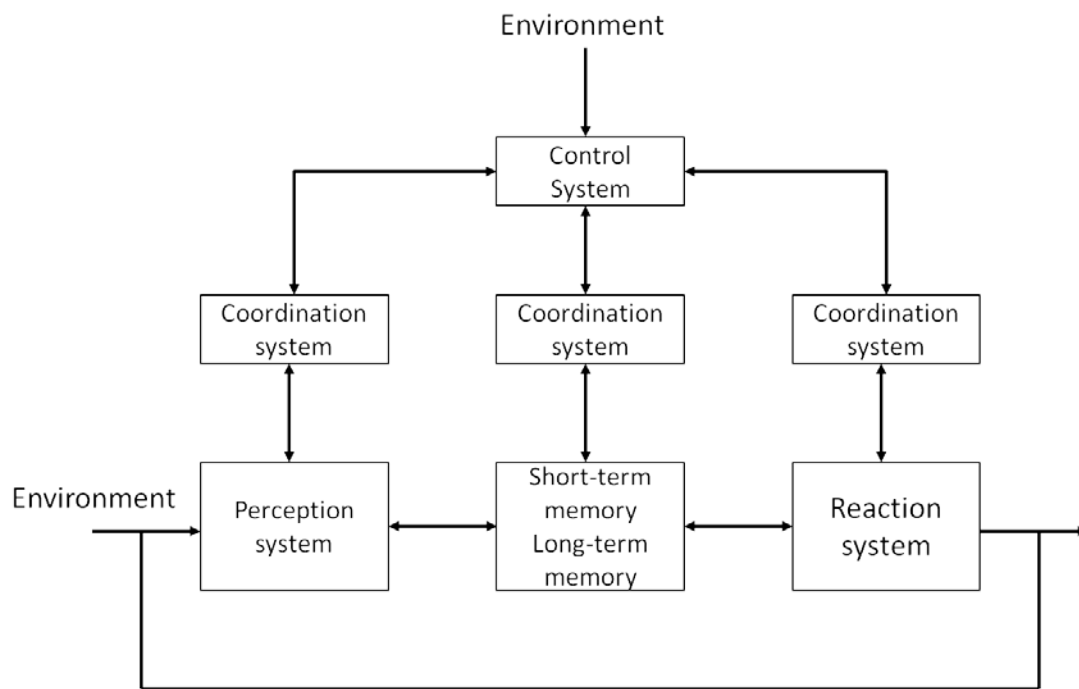

Fig.1 Block diagram of intelligent control system

of the needed information. On the other hand, the short-term memory refines some portion of the information stored in the long-term memory. The long-term memory is a huge information storage-base storing various kinds of information, such as sport skills, grammar information, semantic information, pragmatic information, value, processing procedure, etc. Migrating with current and past input, some portion of the long-term memory is activated. The activated information is called active memory [3].Some portion of the active memory undergoes a refining process in the short-memory which is the place where current cognitive activities take place. It is obvious that there must be a channel for information exchange between the short-memory and long-memory .The information exchange at the same time serves as a coordinator between the control system and reaction system. The control system itself has complete, multi-functional, high-level learning system. The control system determines how the entire intelligent control ssystem works. By means of direct learning from the environment and making use of the information stored in the memory system, the control system implements its organization and management function, processes the objectives and provides policy to achieve these objectives, that is, determines a plan and make a decision. The coordination system is an intermediate structure between the control system and other subsystems and makes decision and coordination based on short-term memory ,"to pass information from the higher level down to the lower level" and "to report information from the lower level to the higher level". The reaction system controls the output of the entire intelligent system ranging from various actions to language and expression.

3 Basic principle in intelligent system

Assume the intelligent system S shown in Figure 1 can be expressed as following triplet with a function

$$S = (X, F, P, H (p_0, p)) \tag{1}$$

Where X is the state space; F is a σ-algebra; P is a probability measure on (X, F); $H(p_0, p)$ is relative entropy defined as

$$H(P_0, p) = \int p_0(x, \theta) \ln \frac{p_0(x, \theta)}{p(x)} dx \tag{2}$$

Where $p_0(x, \theta)$ is determined by observation, $\theta = (\xi, \eta, t)$ is a parameter, where ξ is the action rule of the intelligent system S, η is the event database ,i.e. event set stored in the memory system, and t is time; $p(x)$ is a function of maximum entropy probability density representing a priori knowledge and is given by the following equation

$$p(x) = \frac{1}{Z} \exp(-mU(x)) \tag{3}$$

Where $U(x)$ is a vector potential, m is a vector factor and Z is partition function.

Whenever there is a new sufficiently large input to the system, the system executes a minimization process of formula (2) and when the process ends, the $0-1$ symbol string supplied by the reaction system is the optimal task of the system.

The following theorem demonstrates that the relative entropy can be minimized and the minimum obtained is unique.

Theorem 1. Express the relative entropy in equation (2) as $H(p_0, p)$, we have

$$\lim_{i \to \infty} H(p_o(x_i), p(x_i)) = 0 \tag{4}$$

Proof   Because of $p_0(x_i)$ is estimates to $p(x_i)$ which are obtained by any practicable method to do random experiment (learning) about certain constraint of $p(x)$, for example, because of formula (3), there is a Markov chain of observable sample $x_1 \ldots\ldots x_n$ whose limit distribution is $p(x)$ on state space X .By means of

ergodic theorem, from orbits $x_i(t), i = 1,2,\ldots\ldots ; t \geq 0$; of the Markov chain, estimate

$p_0(x_i)$ of density function $p(x_i)$ can be obtained. The process is producing directive

sequence $p_0(x_i)$ $(i = 1,2,\ldots\ldots)$ and its directive subsequence

$p_0(l(i); l(i) \geq i) = p(x_i) = p(i)$ on complete metric space (D,d), D is set of

probability density function on measurable space (X,F). At this time ,

$$\lim_{i \to \infty} p_0(i) = \lim_{i \to \infty}(p_0(l(i); l(i) \geq i) = \lim_{i \to \infty} p(i).$$

In fact, suppose $\lim_{i \to \infty} p_0(i) = \overline{\eta}$, then for any small positive number ε, there is

positive integer I, such that when $i > I$, $\| p_0(i) - \overline{\eta} \| < \varepsilon$. Because of $\lim_{i \to \infty}$

$\{ l(i) \} = \infty$, thus there is a positive integer i_0, such that when

$i > i_0, l(i) > I$, at this time $\| p_0(l(i) - \overline{\eta} \| < \varepsilon$, hence subsequence $\{ p_0(l(i) \}$

$= \{ p(i) \}$, $(i = 1,2,\ldots\ldots)$ converges and

$\lim_{i \to \infty} p_0(i) = \lim_{i \to \infty} p_0(l(i), l(i) > i) = \lim_{i \to \infty} p(i) = \overline{\eta}$. The proof is completed.

The proving process of above theorem is the process in which $H(p_0, p)$
is minimized. The limit theory dictates that there must be a minimum value
approaching to 0, and the minimum value is unique. The minimum value
corresponds to the uniquely correct $0,1$ character string for
implementation of a given objective(environmental input). At control level,
a $0,1$ character string corresponding to the correct decision—making is

given, and at the reaction system(level), a $0,1$ character string

corresponding to correct task is given.
From the above discussion, we find the following proposition
Proposition The sufficient and necessary condition for the
intelligent system $S = (X,F,P,H(p_0,p))$ to exist is that it can be given
by

$$S = (X,F,P,H(p_0,p)) = \lambda \min_{p_0} \{ \int_X p_0(x,\theta) \ln \frac{p_0(x,\theta)}{p(x)} dx \}, \tag{5}$$

where $\lambda > 0$.

Definition 1. Knowledge on the intelligent system S is defined as the structural information on the intelligent system S, measured by equation (2).

Definition 2. The intelligence of an intelligent system is change ratio of the knowledge acquired by the intelligent system to the linkage coefficient between component units in the intelligent system.

Theorem 2 (Basic principle of intelligent system) . Entropy increases or decreases together with intelligence on the intelligent system S.

To make the discussion simple and sufficient, the concept of sufficiency of relative entropy is introduced [4].

Definition 3. For the intelligent system S, we assume $\mathfrak{A}$ is $\sigma-$algebra on set $\mathscr{X}$ of sub-vectors of $x \in X$ produced by its subsystem—memory system and control system, it is $\sigma-$subalgebra of F, that is, $\mathfrak{A} \subseteq F$. Let $P(X)$ is set of the all probability measures on (X, F) and $P^{'} \subset P(X)$。 By $\sigma-$subalgebra $\mathfrak{A}$ is sufficient with respect to $P^{'}$ is meant there exists $\mathfrak{A}$ —measurable function

$$h = E_\mu(1_A \mid \mathfrak{A}\ ),\ \mu - a.e., \forall \mu \in P^{'} \tag{6}$$

for any $A \in F$.

Assume P_0 and P are probability measures corresponding to probability density function $p_0(x)$, $p(x)$, respectively, $\{P_0, P\} \in P(X)$. From the definition of sufficiency we easily find that $\mathfrak{A}$ is sufficient with respect to $\{P_0, P\}$. Therefore we have :

$$H_{\mathfrak{A}}(p_0, p) = H_F(p_0, p) = H(p_0, p).$$

Since the $\mathfrak{A}$ is sufficient with respect to $\{P_0, P\}$, then the difference between P_0 and P can be determined all by $\mathfrak{A}$ alone. Therefore the conclusion concerning system

$$S = (X, F, P, H(p_0, p)) = \lambda \min_{p_0}\ \{ \int_X p_0(x, \theta) \ln \frac{p_0(x, \theta)}{p(x)} dx \}$$ can be discussed on its

subsystems—memory system and control system.

Following this thinking in the discussion, we can not only prove the basic principle of the intelligent system S, but also produce a special algorithm —algorithm of mean—field theory approximation [5]. The proof of theorem 2 is

given in following

Note that vector m in equation (3) denotes linkage coefficient between component units(nerve cells).For simplicity, we rewrite $p_0(x,\theta)$ as $p_0(x)$ .We will find

$\partial H_\mathfrak{A}(p_0,p)/\partial m_i$ in the following.

$$\frac{\partial H_\mathfrak{A}(p_o,p)}{\partial m_i} = -\int \frac{p_0(x_T)}{p(x_T)}\frac{\partial p(x_T)}{\partial m_i}dx_T \tag{7}$$

Where $x_T \in \mathscr{H}.$ We have:

$$\frac{\partial p(x_T)}{\partial m_i} = [\exp(-\sum_i m_i U_i(x_T))]U_i(x_T)/Z_N - p(x_T)[\sum_{x_T}\exp(-\sum_i m_i U_i(x_T))]U_i(x_T)/Z_N$$

$$= p(x_T)U_i(x_T) - p(x_T)\sum_{x_T}p(x_T)U_i(x_T)$$

Where Z_N is restriction of Z on $\mathscr{H}.$

Substitute above equation into (7) we have

$$\frac{\partial H_\mathfrak{A}(p_0,p)}{\partial m_i} = -\int \frac{p_0(x_T)}{p(x_T)}\frac{\partial p(x_T)}{\partial m_i}dx_T$$

$$= -(-\int p_0(x_T)U_i(x_T)dx_T + \int p_0(x_T)\sum_{x_T}p(x_T)U_i(x_T)dx_T)$$

$$= E_{p_0}(U_i(x_T)) - E_p(U_i(x_T)) \tag{8}$$

Because of $E_{p_0}(U_i(x_T)) = H_{p_0} - \ln Z_N$,equation(8) reads

$$\frac{\partial H_\mathfrak{A}(p_0,p)}{\partial m_i} = H_{p_0}(x_T) - \ln Z_N - E_p(U_i(x_T)),$$

Consequently we have:

$$\ln Z_N + E_p(U_i(x_T)) = H_{p_0}(x_T) - \frac{\partial H_\mathfrak{A}(p_0,p)}{\partial m_i} \tag{9}$$

Where $H_{p_0}(x_T)$ is the maximum entropy determined by the observation sampling set or training sampling set, and is the measurement of uncertainty of task executed by the memory system and the control system. Notice $E_p(U_i(x_T))$ is determinate real number. After the number of component units (nerve cells) of memory system and control system is determined, Z_N(partition function) is determined as well. Therefore the left side of equation(9)can be

regarded as a constant .If the first item at the right side increases(or decreases),the second item must increases(or decreases) .Since $\mathfrak{A}$ is sufficient for $\{P_0, P\}$, the above conclusion is completely applicable to the entire system S. The proof of basic principle is completed.

4 Intelligent behavior of intelligent system

Intelligent behavior of the intelligent system S is accomplished by the subsystems—control system and memory system. The control system and memory system are systems of neural networks. Various intelligent behaviors are shown biologically as the change of linkage weight between nerve cells.

Learning process is the process of acquiring knowledge .The learning process of system of neural network is divided into two phases. The first phase is a processes in which the system of neural systems abstracts the environment and establishes the to be solved problem model—model establishment process. The second phase is a process in which the problem is solved —namely uncertainty caused by environmental regularity in the system is minimized. Corresponding to the two phases, there are the following results.

Theorem 3[6].   Assume the compounded system formed by system N of neural network and the environment E that it is in is isolated from the outside world, and the learning process of the system of neural network is a Markov process that has a given transition probability function .When input information of sufficient magnitude is obtained, the system of neural network starts its learning process. The process will finally arrive at its the equilibrium from far from the equilibrium. The equilibrium is state when the system has maximum entropy. The sufficient and necessary condition of the maximum entropy is that the distribution density function of the system is given by

$$p(x_T) = \frac{1}{Z_N} \exp(-m_N U(x_T)) \tag{10}$$

Where x_T is state variable of the system of neural network, $U(x_T)$ is vector potential function and Z_N is partition function。

Theorem 4[6].   Under the condition of the theorem 3, assume the linkage weight between the i th and j th nerve cells of system of neural network is w_{ij} and $w_{ij} = w_{ji}$, then based on basic principle of the intelligent system, we have

$$\frac{\partial H_{\mathfrak{A}}(p_0, p)}{\partial w_{ij}} = \gamma(P_{ij} - P_{ij}^{'}) \text{ ,where } \gamma = \frac{1}{kT} \text{ and } k \text{ is Boltzmann constant, } T$$

corresponds to the temperature in a physical system , but is a control parameter here;

P_{ij} is the average probability when there is environmental input and the network

arrives at equilibrium with both unit i and unit j conducting; $P^{'}_{ij}$ is the

corresponding probability when there is not environmental input.

Theorem 5[5]. Under condition of theorem 3, assume that the linkage weight between

the i th and j th nerve cells of system of neural network is w_{ij} and $w_{ij} = w_{ji}$, and

that number of nerve cells of the system of neural network is sufficiently large,
then based on the basic principle of the intelligent system and in the sense of mean

square limit, we find $\dfrac{\partial H_{\mathfrak{A}}(p_0, p)}{\partial w_{ij}} = U_i(E_{p_0}(x_T)) - U_i(E_p(x_T))$, where E_{p_0} and E_p

denote expectation operators of $p_0(x)$ and $p(x)$, respectively.

From theorems 4and 5 we find two kinds of learning algorithms for training stochastic
neural networks:

$$\triangle w_{ij} = \beta(P_{ij} - P^{'}_{ij}) \tag{11}$$

Where β is a constant less than but approaching 1.

$$\triangle w_{ij} = \alpha(E[\dot{x}_T(k)]E[\dot{x}_T(l)] - E[\bar{x}_T(k)]E[\bar{x}_T(l)]) \tag{12}$$

Where α is a constant less than but approaching 1, and $E[x_T(k)]$ is given by

$$E[x_T(k)] = \frac{1}{1 + \exp(\dfrac{-U_k}{T})}, \qquad x_T(k) \in \{0,1\} \tag{13}$$

Or

$$E[x_T(k)] = \tanh(\frac{-U_k}{T}), \qquad x_T(k) \in \{1, -1\} \tag{14}$$

Where $U_k = \sum_l w_{kl} x_T(l)$; $E[x_T(k)]$ is mean field when unit k is excited, $E[\dot{x}_T(k)]$

and $E[\bar{x}_T(k)$ represent the excitement value of unit k when the system arrives at
stable state in " – " phase（ "minus" phase) and " + " phase（ "plus "
phase),respectively. "– " phase —— with input unit to be clamping by
environment(input mode),the excitement value of both hidden unit and output unit
evolves to its stable value according to the rule expressed in equation(13) or (14);
 "+" phase ——with the input unit yet to be clamping by environment(input mode) and

the output unit to be clamping by the required output mode, only the hidden unit changes its excitement value till its stable value based on the same rule.

 Directly applying learning algorithms (11) or (12) to the training of the system of neural network is exactly to use the fastest declining method to find the w_{ij}

when $H_{\mathfrak{A}}(p_0, p)$ approaching minimum, which will cause the system of neural system to be trapped into the local minimum, but in some cases, it is undesirable and sometime it is prohibited. Annealing algorithm is a method to avoid the local minimum and to obtain the overall minimum.

1)Statistic simulation annealing algorithm

Statistic simulation annealing algorithm[7] as a common method for approximately solving problem in large-scale combinatory optimization has seen great development in theory and practice. This algorithm can be widely used in the optimization field to provide solution to various optimization problems and the solution can arbitrarily approach to the overall optimization. This algorithm allow cost function related to state to climb the slope randomly simulating the metal annealing process. That is, with the function of the control parameter T, random noise is added to the conventional fastest decline process, thus if "unfortunately" the system of neural network runs into the local minimum trap, it will be able to get out of this local minimum trap until it maximally approaches the overall optimum, in other word, obtains the overall minimum of the cos t function. In this article, the cost function is the relative entropy and the minimum process of relative entropy is synchronized with that of energy function—const function defined in other articles. The procedures for computer implementation of the statistic simulation annealing algorithm are:

(1)   Decide the sufficiently high initial T and randomly determine the linkage weight between nerve cells of the system.

(2)   Givena small perturbation to initial state $x_T^{\ i}$ of the system, we find

   $x_T^{\ i}$ and the relative entropy increment $\triangle H_{\mathfrak{A}}$.

(3)   If $\triangle H_{\mathfrak{A}} \leq 0$, then accept this change , or otherwise if $\exp(-\triangle H_{\mathfrak{A}})$

   $>$random number$\in [0,1)$, then accept this change and $x_T^{\ i+1} = x_T^{\ i}$.

(4)   Compute new temperature: $T(i+1) = T(0) / \ln(i+1), i \geq 1$, where $T(0)$ is the

   initial temperature.

(5)   Repeat steps (2) through to （4）until T approaches zero and the system no longer makes state transfer.

2)Annealing algorithm of mean-field theory approximation

The above statistic simulation annealing algorithm "inherits " the connatural essentiality of the Mote Carlo method —slow convergence rate and high computation complexity, which greatly restricts its application. For this reason, in recent years, specialists and scholars have put forward annealing algorithm of mean-field theory approximation, which has yielded good result. All methods for deducing annealing algorithm of mean-field theory approximation are based on the reduction of the system free energy in statistic physics. Our deduction[5] of this algorithm from the perceptive of relative entropy minimization is determined by the basic principle of the intelligent system and is a natural extension and result of the thinking — using relative entropy minimization to implement the intelligent control.

The annealing algorithm of mean-field theory approximation is as follows:
(1) Randomly choose a high-value parameter T and randomly initialize the mean field

$E[x_T(k)]$ and linkage weight w_{kl} of all free units.

(2) Do following cycle :

a) Randomly choose a mean-field variable $E[x_T(l)], l = 1, \cdots, r$, and compute according to equation (13) or (14) till to arrive at stable state to find

$E[x_T^1(1)]$.

b) Decrease T repeat step a) until a stable solution of equation (13) (or (14))

appears.
(3) Implement the above procedures on "—" phase and "+" phase, respectively.
(4) Modify weight according to equation (12).

The annealing algorithm of mean-field theory approximation has a convergence rate about $50 \sim 100$ times higher than that of statistic simulation annealing algorithm and declines the time complexity from $O(2^n)$ to $O(n)$, where n is the number of nerve cells in the system of neural network.

The above two learning algorithm can be used to train large kinds of stochastic neural network, for example, Markov neural networks[6], neural network based on mean-field theory approximation[5], and so on. Our system of neural network in the intelligent system has learning function[6], memory function[8] and thought function[9]. Having constructed the intelligent system , having given mathematical method an alyzing it and its algorithm,   we have constructed theory of intelligent system, which is implemented through relative entropy minimization.

Reference
[1]Guangcheng, X. (1987), "Entropy—method of partition of complex system" , ACTA AUTOMATICA SINICA (In Chinese) , Vol. 13, No. 3, pp. 216—220.

[2]Mode M C and Saridis G N. (1990), "A Boltzmann machine for the organization of intelligent machines", IEEE Trans. Syst., Man, and Cybern., Vol. 20, No. 5, pp. 1094—1102.

[3]Lindsay PH and Norman DA. (1977), "HUMA INFORMATION PROCESSING An Introduction to Psychology"(Second Edition), Academic Press.

[4] Halmos P R and Savage L T. (1949) "Application of the Radon—Nikodym theorem to the theory of Sufficient Statistics", Ann. Math. Statistics, Vol. 20, pp. 225—241.

[5] Guangcheng, X. (1995), "Neural network Based on Mean-field theory approximation", ACTA ELECTRONICA SINICA(In Chinese), Vol. 23, No. 8, pp. 62—64.

[6]Guangcheng, X. (1991), "A tentative investigation of the learning process of neural network system", ACTA AUTOMATICA SINICA(In Chinese), Vol. 13, No. 3, pp. 311—316.

[7] Kirdpatrick S, Gellat C D and Vechi M P. (1983), "Optimization by Simulation Annealing", Science, Vol. 220, No. 4598, pp. 671—680.

[8] Guangcheng, X. (1998b), "A stochastic theory of associative memory", Control Theory and Application(In Chinese), Vol. 15, No. 5, pp. 688—694.

[9]Guangcheng, X. (2003), "Variability of structure of Abstract Neural Automata and the ability of thought", Kybernetes, Vol. 32, No. 9, pp. 1549—1554.

Machine Learning Approach for Analyzing Health Data

Douglas P. Taggart, Alex I. Waltz, Gabriela Hristescu
Computer Science Department, Rowan University, 201 Mullica Hill Rd., Glassboro, NJ 08028
hristescu@rowan.edu

Abstract   *The availability of large and diverse datasets has lead to many exciting opportunities for the development of tools for knowledge discovery. Artificial Intelligence algorithms have a long history of being used to analyze large and complex datasets. Recent studies have suggested that the commonly used diagnostic measure, the Body Mass Index, may not be the most appropriate indicator for obesity. In this study we use data from a comprehensive study conducted by the Center for Disease Control and a machine learning approach in order to develop a domain independent tool to produce a binary decision tree. We use this tool in order to assess the most important risk factors for obesity as well as to classify an individual's risk for obesity.*

1   Introduction

With the proliferation of large and diverse datasets, the need to develop and expand knowledge-discovery technology is constantly growing. There is a great need to use a large range of tools to extract information from these emerging databases. This paper presents a data mining approach that uses a machine learning algorithm in order to evaluate the most important factors in obesity while also providing a tool for the classification of new data.

The software tool that we have developed analyzes a large dataset provided by the Center for Disease Control (CDC) [1] in order to asses what the most important factors are that influence obesity in the population. Recent studies [2] have advanced the idea that the Body Mass Index (BMI) measure [3], despite its simplicity and wide use for diagnosis, may not

be the best indicator for obesity. The goal of this study is to look beyond the BMI measure in order to find a better understanding of what may be the most influential factors in obesity. This research takes into consideration possible factors of obesity by considering individual's health issues and habits along with regional factors.

The software tool was implemented using a data independent and objected oriented approach and can be used effectively with any other type of data. It uses a machine learning algorithm, Iterative Dichotomiser 3 (ID3) [4], which builds a decision tree based on the data provided. The resulting decision tree can then be used to evaluate an individual's risk for obesity.

2   Background and Related Work

The main algorithms that have been developed for decision tree learning take a top-down, greedy approach through the space of decision trees. They are the Iterative Dichotomiser 3(ID3) [4] and its extension C4.5 [5] and were developed by J. Ross Quinlan. ID3 is popular amongst researchers and academics performing research in health and medical fields.

With the prevalence of Electronic Medical Records (EMR), it is much easier for researchers to gain accesses to large and robust medical databases. This has generated a large amount of research focused primarily on uncovering the causes of various diseases. Researchers in Taiwan used an ID3-based algorithm to produce a tree that was confirmed to be useful for clinical diagnosis of cerebrovascular disease [6]. Another study at Shiyou University in China [7] used the ID3 algorithm for finding hidden classification rules among data relating to

students' physical strengths and sports aptitude levels. The tool has been successfully used to map students' various fitness levels, enabling physical education instructors to improve their course plans and activities. In another study, researchers at the University of Michigan and the Huazhong University developed a workbench [8] for testing the integrity of machine parts. Their tool employs the ID3 algorithm as a basis for testing the tolerance of machine parts commonly used in the automotive industry. A related collection of machine learning algorithms for data mining tasks has been developed in Weka [9].

3 Data

In our research we used a dataset made available by the Center for Disease Control (CDC) [1]. The dataset is the result of a 2009 study into community health indicators. While the dataset contains a large pool of measurements, we have chosen to focus solely on data pertaining to obesity and obesity risk factors. From the CDC dataset the following ten attributes were selected: exercise frequency, fruit and vegetable intake, obesity, blood pressure, smoking habits, diabetes, being insured, access to primary care physicians, access to dental care, and access to community health centers. The data was collected during the 2008 census for each of the 3,141 current counties in the United States.

4 Methods

The Iterative Dichotomiser 3 (ID3) [4] algorithm was implemented and adapted to handle continuous valued attributes and training data with missing attribute values. ID3 was chosen over other comparable algorithms, such as C4.5, as it produces a relativity small decision tree with a boolean outcome. ID3 uses the statistical property called Information Gain. The Information Gain measures how well a given

attribute separates the training data into the targeted classification, by computing the expected reduction in entropy caused by the partitioning of the data according to this attribute.

If the target classification is boolean and the attribute S can have c different values, then the definition of the entropy [10] of a collection of examples S is:

$$Entropy \;\; (S) = \sum_{i=1}^{c} - \, p_i \log_2 p_i \qquad (1)$$

where p_i is the proportion of S belonging to class i. The definition of the information gain, Gain(S, A) of an attribute A, relative to the collection of examples S, is:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} * Entropy(S_v) \;\; (2)$$

where Values(A) is the set of all possible values for attribute A and S_v is the subset of S for which attribute A has value v.

To make the ID3 algorithm work with the CDC data, we had to address two issues. Firstly, the attributes had values in a continuous rather than discrete range. Secondly, some of the attributes from the CDC data had missing values. To work with boolean values for the outcome, we had to find the appropriate threshold against which to compare the values of the continuous attributes. Even though it is common practice to choose the midpoint of the range as the representative threshold, we found that a more meaningful value was the median value of the dataset. To deal with the missing attribute values, we chose to exclude from consideration the data points with missing data.

For example, consider the data in Table 1, which has ten data points with attributes named No_Exercise, High_Blood_Pressure, Smoker, Diabetes, Uninsured, Few_Fruit_Veg, and the target classification is is_Obese. Note that representative thresholds have been used to transform the values for all attributes to boolean values.

An example for the computation of the information gain for the attributes High_Blood_Pressure and Smoker is shown in Figure 1.

Subject	No_exercise	High_Blood_Pressure	Smoker	Diabetes	Uninsured	Few_Fruit_Veg	is_Obese
Sub1	Yes	Yes	Yes	No	Yes	No	Yes
Sub2	Yes	Yes	Yes	No	No	No	No
Sub3	Yes	Yes	Yes	Yes	No	No	Yes
Sub4	Yes	Yes	Yes	No	No	No	Yes
Sub5	Yes	Yes	Yes	No	No	No	Yes
Sub6	Yes	No	Yes	No	No	Yes	Yes
Sub7	Yes	No	Yes	Yes	Yes	No	Yes
Sub8	Yes	No	Yes	Yes	No	No	No
Sub9	Yes	No	Yes	No	No	Yes	Yes
Sub10	Yes	No	No	No	No	No	No

Table 1: Training examples for the target outcome is_Obese. The table only shows a small sampling of the large training dataset used to build the decision tree.

$$S: [115+, 1264-]$$

High_Blood_Pressure

Yes   No

$$[98+, 599-] \qquad [17+, 665-]$$

$$S: [115+, 1264-]$$

Smoker

Yes   No

$$[115+, 1193-] \qquad [0+, 71-]$$

Gain $(S, \text{High\_Blood\_Pressure})$
$= .847 - (115/1379)\,.643 - (1264/1379)\,.105$
$= .697$

Gain (S, Smoker)
$= .847 - (115/1379)\,.416 - (1264/1379)\,.774$
$= .102$
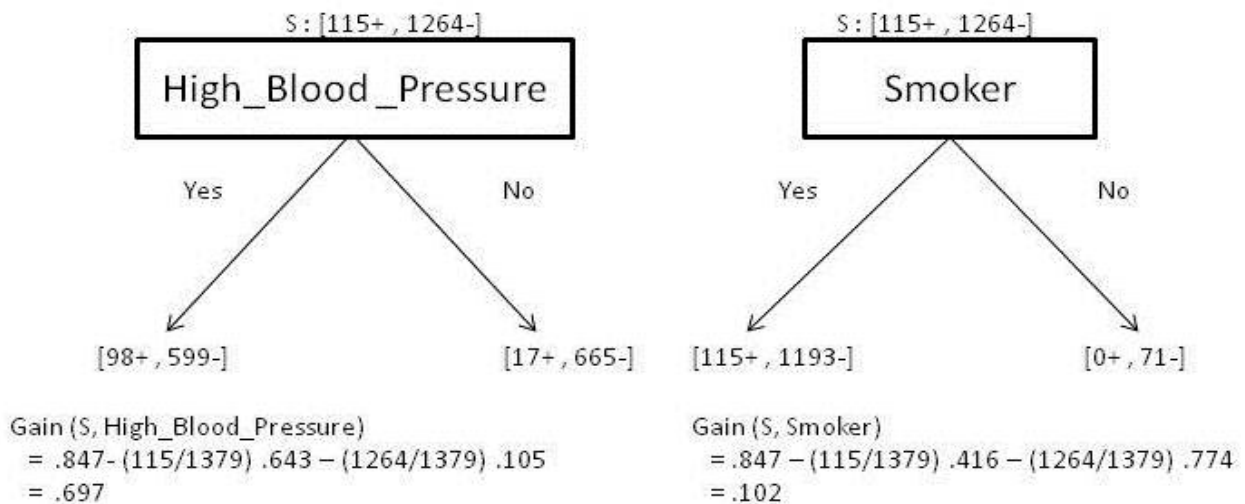
Figure 1: Entropy and Information Gain, as caculated for the High_Blood_Pressure and Smoker attributes at the root level. For High_Blood_Pressure, the data set has 115 positive examples and 1264 negative examples. The dataset results in collections of [98+,599-] (High_Blood_Pressure = true) and [17+,665-] (High_Blood_Pressure = false). The resulting information gain is .697 for High_Blood_Pressure compared to an information gain of only .102 for the attribute Smoker.

The ID3 algorithm uses a top down, greedy approach to building the decision tree. Once a node has been placed in the tree, the algorithm does not revisit it or reconsider its placement. The dataset is divided into two subsets: a training set that is used to build the tree, and a test set that is used to verify the validity of the decision tree. We have chosen a random sampling comprising 90% of the original dataset to serve as the training data set. The remaining 10% is used for the test set. The algorithm has been implemented such that the size of the two sets is variable and therefore can be changed during the course of the experiment.

ID3 uses a recursive algorithm to build the decision tree. The attribute with the highest overall Information Gain is selected to be the root of the tree. The chosen attribute is then removed from the set and the data set is split on the values of this attribute. The same method is then used recursively for each of the subtrees to find the highest information gain attribute from the remaining attribute set and using the remaining training set data for that specific branch. A base case is reached when either all attributes have been placed on the tree or the classification is the same for all the data points in the remaining training set.

5   Experimental Work

The application was built using Java [11] for the framework, MySQL [12] was used as the Database Management System, and the Java Universal Network and Graph Framework (JUNG) [13] was used for visualization of the decision tree.

The system was designed to be data independent, so that the application can handle a diverse range of other datasets. The application connects to a MySQL database and retrieves the data through the Java Database Connectivity (JDBC). Dependency injection, and more specifically constructor injection, was used to implement the database connection. This was done such that the specific database connection can be dynamically changed at runtime and is therefore database independent. By adopting a uniform approach to data handling, the application is not domain bound, and can easily be adapted to be used with a wide range of domain types. The application was used with aggregate health data, but can easily be modified to accommodate data types from a variety of other domains such as finance or material engineering.

Once the application has successfully queried the database and populated the appropriate data structures, the dataset is divided into two subsets, the training data and the testing data. The application uses a randomly chosen 90% of the original data for the training data set to build the decision tree. The application uses the Java shuffle algorithm [14] to randomize the data. During the construction and the testing of the tree, any data point that is found to have a missing attribute value is eliminated from the current process in order to maintain data integrity. The ID3 algorithm then uses a pre-order recursive traversal to build the tree. At each step, the entropy and information gain are computed for the remaining attributes, and the attribute with the highest information gain is placed in the tree. In the computation of entropy and information gain, if a data point has missing values for a given attribute, that entry is removed from consideration for the construction of the subtree.

The Java Universal Network and Graph Framework (JUNG) [13] was used to create a graphical representation of the decision tree. JUNG is a software library that provides a common and extendible language for the modeling, analysis, and visualization of data that can be represented as a graph or network. JUNG is written in Java, which allowed us to extend its libraries and modify them to work with our software. The application uses a depth first

traversal to navigate the tree for visualization. Figure 2 shows the root and the left subtree of the decision tree built using the CDC data. Each node is represented by a circle, with the name of the attribute appearing below it. The left and right edges connecting a node to its children are labeled with Yes and No respectively to allow for easy visual navigation of the tree. Similar to the edges, each outcome of a path is labeled with either a Yes (Y) or a No (N).
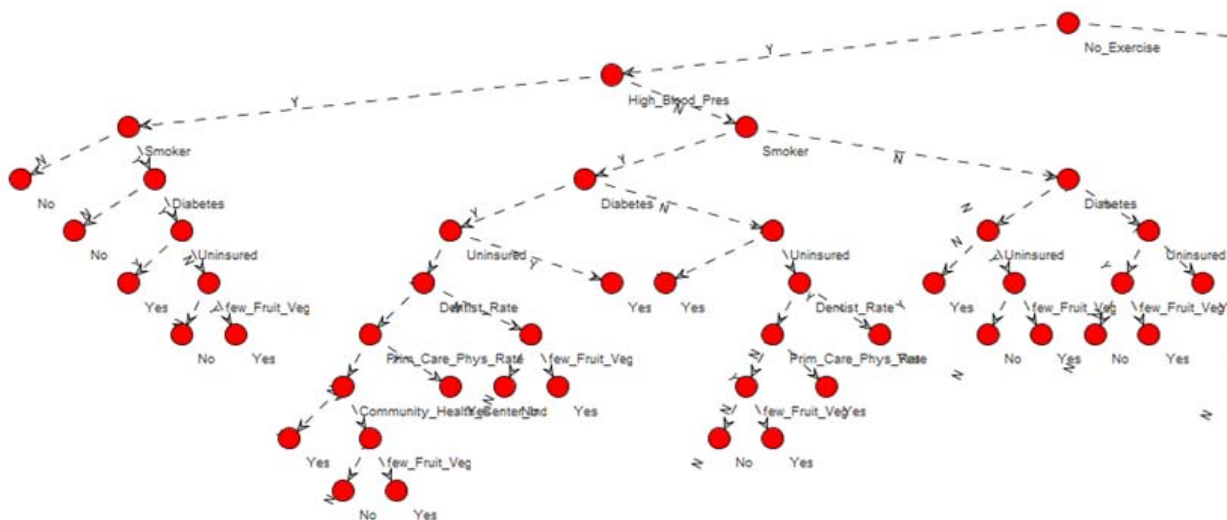

Figure 2: Left half of the resulting decision tree for the target outcome is_Obese. The classification of a new data point follows the path from the root node to a leaf node using the point's attribute values to take either left of right edges.

In addition to providing the user with the visualization of the decision tree, the application uses the remaining 10% of the original dataset to test the validity and accuracy of the tree. The classification attribute in the data that we used was obesity. In order to reduce the continuous valued attribute to a binary one, we used the national average as a threshold. Each data point in the testing set is classified using the decision tree and the final outcome is compared for validation with its own classification. The application keeps track of the total number of correct and incorrect classifications and presents the information to the user.

6 Results and Conclusions

We have used the developed tool on a dataset provided by the CDC [1] that had 3,141 data points each with 10 attributes. We tested the consistency of our application by recording the resulting trees over several runs. During each run a training dataset was randomly selected as a subset of the original dataset. We found that the attribute No_Exercise was placed as the root node 100% of the time. The attribute High_Blood_Pressure was placed on the second level of the tree 95% of the time, while the Smoker attribute was placed on the second level 5% of the time. Similarly, the attribute Few_Fruits_Veg was consistently placed on the last level of the tree. The only slight variation noted was in the middle of the tree where the attributes access to primary care physicians (Prim_Care-Phys_Rate) and access to dental care (Dentist_Rate) were placed interchangeably on levels 6 and 7 in the tree. This, however, does not affect the major findings for this study, as the top level attributes remain consistent. The

results showed that, using the CDC data, out of all the considered factors for obesity, the most important one was not getting enough exercise. The second most important factor was having high blood pressure, closely followed by being a smoker. The least important factor of the ones we have considered was not eating enough fruits and vegetables.

The machine learning application that we have developed has been specifically designed to be data independent and therefore can be used with any type of data sets. Potential areas such as material science, meteorology and banking, where the attributes have boolean values and outcomes, are good candidates for data mining using this tool. Any other data with continued valued attributes and for which threshold values can be chosen, that do not affect the classification of the target outcome, can be used with this tool. The dataset that has been used in this study had missing values for some attributes, a common problem in many large datasets. A further development of this application may be designed to handle the data points that have missing attribute values differently. Instead of excluding these data points from further consideration, the application can replace the missing values with values computed using statistical inference. We expect his approach to increase the robustness of the application.

References

[1] (2009) *Community Health Status Indicators.* [Online]. Available: http://wwwn.cdc.gov/CommunityHealth/Companion_Document/CHSI_DataSet.zip

[2] Jeremy Singer-Vine. (2009) *Beyond BMI: Why doctors won't stop using an outdated measure for obesity.* [Online]. Available: http://www.slate.com/id/2223095/

[3] (1998, Sept.). *National Institutes of Health, National Heart, Lung, and Blood Institute. Clinical guidelines on the identification, evaluation, and treatment of overweight and obesity in adults*, NIH publication No. 98-4083 [Online]. Available: http://www.nhlbi.nih.gov/guidelines/obesity/ob_gdlns.pdf

[4] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning,* vol. 1, pp. 81-106, Mar. 1986.

[5] J. R. Quinlan, *C4.5: Programs for Machine Learning*, San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[6] D. Yeh *et al.*, "A predictive model for cerebrovascular disease using data mining," *Expert Systems with Applications*, vol. 38, pp. 8970-8977, 2011.

[7] Q. Zhang *et al.*, "Applications of ID3 Algorithm in Exercise Prescription," in *Proceedings of the International Conference on Electric and Electronics*, 2011, pp. 669-675.

[8] X. Shao *et al.*, "Application of ID3 algorithm in knowledge acquisition for tolerance design," in *Journal of Materials Processing Technology*, vol. 117, pp. 66-74, 2001.

[9] Mark Hall *et al.*, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations*, vol. 11, issue 1*,* 2009.

[10] T. M. Mitchell, *Machine Learning*, New York, NY: McGraw-Hill, 1997, pp. 55–58.

[11] J. Gosling and H. McGilton, "The Java Language Environment," Sun Microsystems, Mountain View, CA, Tech. Rep., 1996.

[12] *What is MySQL?* [Online]. Available: http://dev.mysql.com/doc/refman/5.1/en/what-is-mysql.html

[13] (2012, Jan.) T. Nelson, *Java Universal Network / Graph Framework.* [Online]. Available: http://jung.sourceforge.net/index.html

[14] *Class Collections.* [Online]. Available: http://docs.oracle.com/javase/1.4.2/docs/api/java/util/Collections.html#shuffle%28java.util.List%29

A Multiobjective Memetic Algorithm for Solving the Carsharing Poblem

Laurent Moalic, Sid Lamrous, and Alexandre Caminada
IRTES-SET, Belfort-Montbliard University of Technology - UTBM
F-90010 Belfort, France
{laurent.moalic, sid.lamrous, alexandre.caminada}@utbm.fr

Abstract— *In this article, we present a method combining a genetic approach with a local search for multiobjective problems. It is an extension of algorithms for the single objective case, with specific mechanisms used to build the Pareto set. The performance of the proposed algorithm is illustrated by experimental results based on a real problem with three objectives. The problem is issued from electric carsharing service with a car manufacturer partner. Compared to the Multiobjective Pareto Local Search well known in the scientific literature, the proposed model aims to improve: the solutions quality and the set diversity.*

Keywords: Memetic algorithm, Local search, Multiobjective optimization, Transportation services, Carsharing, Decision making

1. Introduction

Many real world problems require to optimize several objectives simultaneously, they are called multiobjective optimization problems (MOP).

The general MOP can be formulated as:

$$MOP \begin{cases} \max & z = f(x) \\ \text{where} & f(x) = (f_1(x), f_2(x), ..., f_k(x)) \\ \text{subject to:} & x \in X \end{cases}$$

(1)

In most cases considering a multiobjective context, it does not exist a unique solution optimizing all objectives in an optimal way. We need to find other decisional mechanisms. The Pareto dominance is one of these; for MOP, the Pareto set is composed of all best compromises between the different objectives. The Pareto set is achieved if there are no other dominant solutions in the search space. The Pareto front is defined as the image of the Pareto set in the objective space [1]. The Vector Evaluated Genetic Algorithm proposed in 1985 [2] was at the origin of many works using genetic approaches (GA) for solving MOP called multiobjective evolutionary algorithms (MOEA). Some famous algorithms such NSGA-II[3], SPEA [4] and SPEA2 [5] have shown their efficiency on complex problems. To solve single objective combinatorial optimization problems, local search algorithms provide often efficient metaheuristics. They have also been adapted to multiobjective combinatorial problems like in Pareto Local Search algorithm (PLS)

[6] or more recently in FLSMO [7]. Some other works use strategies based on the neighborhood structure [8] or consider a Tabu Search approach [9][10] . A recent work has been done to unify all local search approaches based on a neighborhood search applied to MOP. This unification is called Dominance-based Multiobjective Local Search (DMLS) [11]. Both local search and GA bring interesting behavior. It is why in the past few years, many multiobjective algorithms have proposed to hybridize GA with local search. These approaches are frequently called Memetic Algorithms or Hybrid Genetic Algorithm. Such algorithms have often outperformed results obtained with simple local search or simple GA, especially in a mono-objective context such for the graph coloring problem [12]. In a multiobjective context, memetic algorithms seems to be very promising [13][14] or [15].

The algorithm we propose is an hybridization of a genetic algorithm based on NSGA-II and a local search. Hybridizing NSGA-II and a local search has been done in some works like in [15]. But in our algorithm the local search is not performed after the GA but inside it, instead of the mutation operator. For each generation of the GA, children are obtained by a crossover mechanism from the selected parents and a local search is applied to the offsprings instead of the mutation. By this way the crossover can be considered as the diversification operator while the local search is the intensification one. A good balance between intensification and diversification allows the algorithm to approach the Pareto set.

The proposed algorithm has been applied on a real problem for locating stations of a Carsharing service for electrical vehicles. The good results we obtained in comparison to PLS shows how much the hybridization is one of the most promising directions of research.

The article is structured as follows. In Section 2, we present the proposed memetic algorithm. Section 3 proposes a detailed presentation of the experimentation performed in Paris area, where the hybrid algorithm was applied on a Carsharing service for electrical vehicles. Then we present in section 4 the decision making tool we develop. Section 5 provides an analysis of the experimental results. The last section presents a conclusion and considers future directions.

2. Memetic algorithm for Multiobjective problems

The proposed algorithm is based on an hybridization of two main approaches: a GA and a local search. It is a population-based algorithm which manages two population sets. The first population with a fixed size is used by the GA for selecting promising parents. The second population is an elitist set used as an archive for collecting all non-dominated solutions reached by the algorithm. During the local search, each acceptable neighbor is stored in the archive if it is not dominated by the solutions found so far.

The GA used in the hybridization is NSGA-II[3] proposed by Deb et al. This algorithm is characterized by:

- a fast non-dominated sorting approach according to the level of non-domination
- a density estimation of solutions surrounding known as the crowding distance
- a sorting function according firstly to the level of non-domination and secondly to the crowding distance

Let's remind that the level of non-domination is computed as fallows:

1) All non-dominated solutions have the level 1
2) The non-dominated solutions of level 1are removed
3) All remaining non-dominated solutions have level 2
4) The non-dominated solutions of level 2 are removed
5) So on...

The crowding distance for a solution s is defined as the average side-length of the cuboid defined with the previous and the next closest solutions.

The algorithm 1 presents these steps.

Algorithm 1 Main Memetic: NSGA-II based approach

Require: population size N, generation number $nbIter$
  {$nbIter$ can be replaced by a time limit}
 1: $P \leftarrow$ init(N) {init population P with N random individuals}
 2: $Q \leftarrow \emptyset$ {init an empty children population}
 3: eval(P) {eval objectives for each individual}
 4: **for** $i = 1$ to $nbIter$ **do**
 5:     $P \leftarrow P \cup Q$
 6:     assignRank(P) {based on Pareto dominance}
 7:     **for** each non-dominated front $f \in P$ **do**
 8:         setCrowdingDist(f)
 9:     **end for**
10:     sort(P) {by rank and in each rank by the crowding dist }
11:     $P \leftarrow P[0 : N]$
12:     $Q \leftarrow$ buildChildren(P)
13: **end for**

Moreover NSGA-II as a GA needs operators for selection/crossover of the parents and mutation. Since here we consider an hybrid algorithm, mutation will be replaced by the local search in a selected direction. These important steps are done by the function called buildChildren(P) as follows:

- selection: as defined in NSGA-II this is done by a binary tournament based on the solutions ranking
- choice of a random direction combining the objective functions
- crossover: an elitist recombination for the chosen direction
- mutation: child improvement with a local search in the chosen direction

The initial population is built randomly, with n random solutions. The result is an approximation of the Pareto set. The proposed approach combines two qualities: a good intensification based on the local search and a good diversification thanks to the crossover.

2.1 Random choice of the direction to explore

At the previous step solutions were ranked according to the Pareto dominance. Now for crossover and local search a direction $\omega = (\omega_1, ..., \omega_k)$ is randomly chosen with k the number of objective functions, $\omega_i \geq 0$ and $\sum_{i=1}^{k} \omega_i = 1$.

The vector direction allows to transform temporarily the MOP to a single-objective problem. Each solution has a unique fitness value f which is the weighted sum of objectives values f_i.

$$f = \sum_{i=1}^{k} \omega_i * f_i \tag{2}$$

This allows to get an ordered relationship between solutions.

2.2 Crossover in the GA

In the memetic approach the crossover may be seen as a diversification operator. In our algorithm we propose to control the diversification with an elitist mechanism.

We suppose here that each variable of an individual is associated to its contribution to every objective functions. The crossover will select from each parent the best variable value according to the selected direction.

2.3 Local Search instead of the mutation

For this operator we propose to use a fast local search algorithm. The selected one is First Improvement Hill Climbing (FIHC) in which the first neighbor with a better quality is chosen (partial neighborhood exploration). Local search is a pair (Ω, V) where Ω is a set of feasible solutions (search space) and V a neighborhood structure $V : \Omega \rightarrow 2^{\Omega}$ that assigns to every $s \in \Omega$ a set of neighbors $V(s)$. Local search is applied until being in a local optimum s^* such that $\forall s \in V(s^*), f(s) \leq f(s^*)$ for a maximization problem.

Algorithm 2 Local Search: FIHC

Require: s the child coming from crossover operator
1: $f_s \leftarrow \sum_{i=1}^{k} \omega_i * f_i(s)$
2: **repeat**
3:     $s' \leftarrow$ selectNeighbor(s) {select randomly the first neighbor of s such that $f'_s > f_s$}
4:     **if** $s' \neq \varnothing$ **then**
5:         $s \leftarrow s'$
6:         $f_s \leftarrow \sum_{i=1}^{k} \omega_i * f_i(s)$
7:         addNotDominated(s) {add s in the archive if not dominated and remove all dominated solutions}
8:     **end if**
9: **until** s is a local optimum (i.e. $s' = \varnothing$)

---

3. Study case: Problem and data

In our study we consider a Carsharing system. Carsharing services were first experimented in 1940 [16]. To deploy the service, we need to locate stations where the people take and return the cars. Here it is not necessary to return the vehicle in its starting station. This approach where vehicles can be return to any station of the service is known as One Way Carsharing. Solving approaches based on exact methods already exist such as [17] but they consider simplified problem. We have applied the proposed memetic algorithm to approximate the Pareto set of this problem. The aim is to locate n stations in a given area to maximize the daily requests of population flows.

As it will be detailed in section 3.3, our model aims to:

- increase the number of people using the service
- reduce the jockeying effort (relocation operation performed by a staff) to alleviate the number of rejected demands
- increase the use of the service during all the day (not only at peak time)

A client's demand is considered as rejected in two cases:

- the client arrives to a station to rent a car and couldn't find any available one
- the client arrives to a station to park the car and couldn't find an empty place

3.1 Mobility model

In this section we will describe the mobility model used in our study and how we used it to generate the input data for our heuristic.

To build the mobility model, several data are necessary to simulate the real environment and people mobility. We used two main types of data:

- GIS Shapefiles describing the geographical entities.
- Survey data and socio-economical information collected by professionals for regional planning needs, describing the main flows of people mobility.

For our study the data are on a 20km * 10km area in Paris region, France. The area is divided into a grid of equal cell size. Each cell is characterized by two properties:

- Terrain type: static information representing the dominant structure type of the area covered by the cell (roads, buildings, houses, company, etc.).
- Attraction weight: dynamic information that varies every 15 minutes during the day. Attraction weights are computed according to cells terrain type and all other survey data.

A model that we have developed before is used to build the people mobility between each cell during the day. All the flows are set in a 3D matrix $F = (f_{i,j,t})$ where $f_{i,j,t}$ represents the number of people moving from the cell i to cell j at time period t.
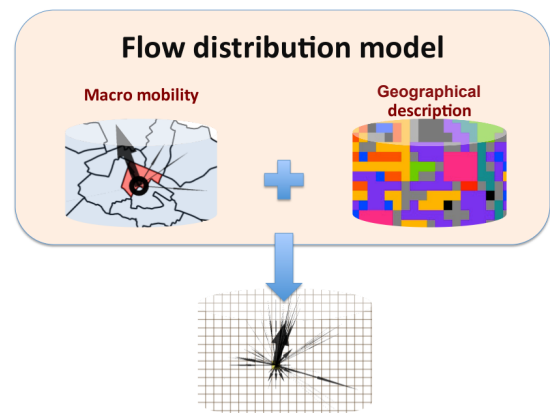

Fig. 1

FLOWS DISTRIBUTION IN CELL

The figure 1 shows the outflows from the selected cell, obtained from macro data and geographical description. The thickness of the arrows reflects the number of people moving between cells. For the purpose of the study we define a cells size of 300m side. Not all the cells can be the origin or destination of a mobility flows such as fields or lakes. For our experimentation nearly 400 cells are considered. That gives 160,000 origin/destination couples. Considering time slots of ¼ hour (96 per day), the flow matrix used contains more than 15,000,000 records describing how people are moving during the day.

3.2 Forecasting on the service users

The algorithm for locating the stations and the evaluation of users' number for each time suppose a good understanding of who will use the service. This part was achieved with the operator who will deploy the Carsharing service.

Among all the mobility flows, some filters are used for selecting the rate of users depending on their profile (age, sex, etc.). Moreover a station is defined by a capture radius which defines the maximum distance the users are ready to

walk to reach the station. Based on this radius value, a station st_i can cover one or more cells. In territory planning it is generally considered that one accept to walk 300m to join a public transportation station.

Fig. 2

STATIONS AND CELLS COVERAGE

In figure 2 are shown stations and their coverage area (colored disc). The inflows / outflows associated to each station are the weighted sum of the flows from the covered cells. The weight depends on the proportion of coverage. If several stations cover the same part of a cell, the associated flows are divided between all of them.

3.3 Stations location

In this section we will discuss how to locate the car stations in the zone of study such that they cover the maximum demand in an appropriate manner.

Stations are located by the algorithm by optimizing three objectives:

f1 : flow maximization i.e. the locations must allow us to maximize the flows between themselves

$$f_1 = \max_{s \in \Omega} \left[\sum_{st_i \in s} \sum_{st_j \in s \setminus \{st_i\}} f(st_i, st_j) \right] \quad (3)$$

f2 : balance maximization i.e. the location must allow us to maximize the balance between inflows and outflows of a station

$$f_2 = \max_{s \in \Omega} \left[\sum_{st_i \in s} \frac{f_r(st_i)}{f_T(st_i)} \right] \quad (4)$$

f3 : minimization of flow standard deviation i.e. the location must allow us to get an uniform flow along the day

$$f_3 = \min_{s \in \Omega} \left[\sum_{st_i \in s} \sqrt{\frac{1}{|T|} \sum_t (f(st_i, t) - \bar{f}(st_i))^2} \right] \quad (5)$$

With,

Ω :   set of feasible solutions

s :   solution element of Ω corresponding to a network of n charging stations

st_i :   charging station i from the solution s

T :   set of time periods of the day

t :   one time period (for instance 15 minutes)

$f(st_i, st_j)$:   number of people moving from st_i to st_j on all time periods

$f(st_i, st_j, t)$: number of people moving from st_i to st_j on time period t

$f(st_i, t)$:   number of people moving from/to st_i on time period t

$\bar{f}(st_i)$:   average number of people moving from/to st_i on all time periods

$$f_r(st_i) = \sum_t min \left[\sum_{st_j \in s \setminus \{st_i\}} f(st_i, st_j, t), \sum_{st_j \in s \setminus \{st_i\}} f(st_j, st_i, t) \right]$$
is the balanced part of the in/out flow throughout the day

$$f_T(st_i) = \sum_t max \left[\sum_{st_j \in s \setminus \{st_i\}} f(st_i, st_j, t), \sum_{st_j \in s \setminus \{st_i\}} f(st_j, st_i, t) \right]$$
is the total flow going through st_i station

The neighborhood of a solution s, needed for the local search, is composed of all the solutions that can be reached by moving any one station of s to any other location. The neighborhood relation is then defined as:

$$v(s) = \{s' \in \Omega | dist(s, s') = 1\} \quad (6)$$

4. Simulation tool and platform of analysis

Good mobility analysis requires an effective tool for representing the results of displacement. A software platform that can run both spatial data such as geographical and time data reflecting the mobility then appears as an essential element.
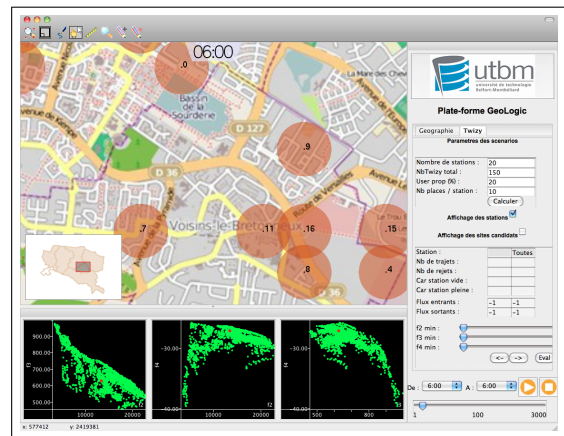

Fig. 3

GEOLOGIC PLATFORM

Many tools exist in the area of GIS (Geographic Information Systems) to represent geo-located data. However, taking into account both space and time is scarce.

So we have developed our own software platform named "GeoLogic". This tool proposes all features found in traditional GIS, such as the display of maps and "shapefiles" vector data, a zoom, etc. Each loaded object, such as roads or buildings delimitations, is a layer that can be shown or hidden.

Moreover this tool reveals in the same view the decision space (geographical part) and the criteria space (plot part). Associated tools allow a decision maker to view the evaluation of a solution, and for a selected location the associated evaluations.

Fig. 5
A MULTIOBJECTIVE MAXIMISATION PROBLEM

Fig. 4
VECTOR INFORMATION: GIS SHAPEFILE

The data can be static or dynamic. While topographic features such as roads or buildings do not vary over time, their properties can change during the day. As an example a supermarket will always be in the same place but will have a different power of attraction at 9:00, 16:00 or 23:00. Similarly the location of the population changes over time. Our software allows taking into account this dynamicity over time. It is possible to vary the speed of scrolling or to pause the application.

5. Performance analysis

This section deals with the performance measurement and the results. In the context of multiobjective optimization the comparison of algorithms is quite more difficult than in single-objective case. Indeed the result is not only one solution but a set of solutions, and each solution has an evaluation on several axes. As long as there is no fitness value associated to the population one can't say which population is better than the other.

In the general case for two approximations of the Pareto front one front can be better for a criteria but worst for another one. As an example figure 5 shows two uncomparable sets since they cannot be totally ordered. The only situation in which a direct comparison is possible is when a set is totally dominated by another one. Then we will use specific indicators explained in the following section.

5.1 Results analysis

For analyzing the results we performed 20 runs of the memetic algorithm, and we will compare the results to a reference set obtained after 20 runs of PLS. The average time before being in a local optimum for PLS is 2 hours. It is the computation time we have chosen for each run of our algorithm. By this way one can consider that the results of both algorithm are comparable.

Figures 6, 7 and 8 presents the projection of the non-dominated solutions for a typical run. Each color/shape presents the solutions for our memetic algorithm and the local search PLS.

The first important result we can see is that the memetic approach produces solutions in zones of the criteria space which are never explored by PLS.
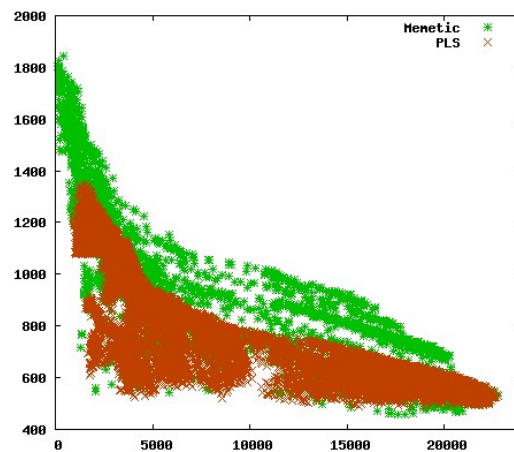

Fig. 6
PROJECTION ON F1 (X) AND F2 (Y) - MAXIMIZING

For a more precise analyze of the results we will use indicators. Indeed a comparative indicator is the most common way to distinguish two sets or to assign a performance measure. Since an indicator gives a restrictive information we propose for this work to consider three indicators:

- the additive ϵ-indicator [18]
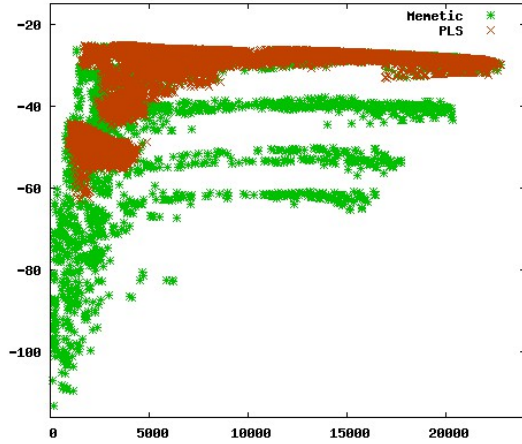- the contribution indicator [19]
- the number of solutions found
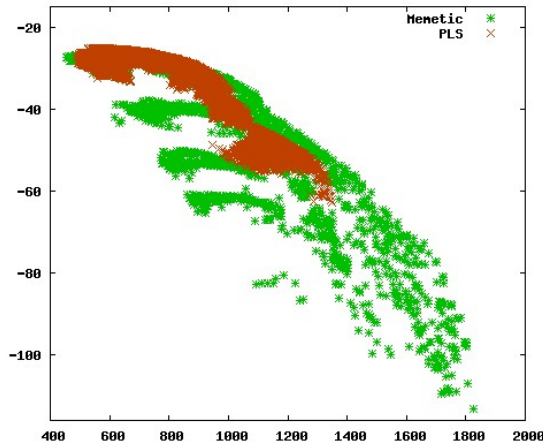
Fig. 7

PROJECTION ON F1 (X) AND F3 (Y) - MAXIMIZING

Fig. 8

PROJECTION ON F2 (X) AND F3 (Y) - MAXIMIZING

additive ε-indicator

The additive ε-indicator gives the minimum factor ϵ by which a set A has to be translated to dominate the set B.

$$I_{\epsilon+}(A, B) = \min_{\epsilon \in \mathbb{R}}\{\forall x \in B, \exists x' \in A : x \preceq_{\epsilon+} x'\} \quad (7)$$

In the equation 7 the $\preceq_{\epsilon+}$ operator represents the weak ε-dominance. For a maximization problem we have the relation:

$$x \preceq_{\epsilon+} x' \Leftrightarrow \forall i \in \{1, 2, ..., n\}, x_i \leq \epsilon + x'_i$$

$I_{\epsilon+}$ is sensitive to the dimension of each objective function. Before comparing solutions in any computation, a normalization of the axes is done.

Table 1
MEMETIC RESULTS

	min	max	mean
f_1	22787.801	22787.801	22787.801
f_2	1783.280	1874.630	1826.831
f_3	-25.217	-25.047	-25.150
NbSol	4515	4915	4766
$\epsilon+$	5.135	2.921	3.719
contrib	0.104	0.117	0.109

Table 2
PLS RESULTS

	min	max	mean
f_1	22787.801	22787.801	22787.801
f_2	1024.540	1347.700	1076.144
f_3	-25.042	-25.039	-25.041
NbSol	8249	12969	9094
$\epsilon+$	76.621	17.232	66.050
contrib	0.202	0.300	0.226

contribution indicator

The contribution indicator computes the proportion of solutions from a set A in $ND(A \cup B)$, where ND represents the non dominated solutions.

$$I_C(A, B) = \frac{\frac{|A \cap B|}{2} + |W_A| + |N_A|}{|ND(A \cup B)|} \quad (8)$$

The indicators $I_{\epsilon+}$ and I_C are used to compare the results of algorithms to the same reference set R. In an ideal situation R would be the optimal Pareto set. For our real problem the optimal Pareto set is unknown. It is why we have built R with all non-dominated solutions found on many runs of PLS and FLSMO.

An $I_{\epsilon+}$ value near to 0 or being negative shows a very good result. In the same way the bigger I_C value is, the better the set is.

The results presented in these tables 1 and 2 give information on the fitnesses evaluation and the three indicators. For each we indicate the best value on the 20 runs, the worst of the max values, and the mean.

A significant result is that our memetic approach can't find as many solutions as PLS. But the found solutions are better spread in the criteria space. The evaluation of f_2 confirms that the results are significantly better with the memetic algorithm than with PLS.

5.2 Progression of the algorithm

Moreover we propose to study how the archive increase during the program run. This allows to compare the way the algorithms converge during the process. To do that we consider the ε-indicator evolution. After each offspring has been improved by the local search, the archive is compared to the reference set. By this way it is possible to compare not only the results of an algorithm but the way to reach these results.
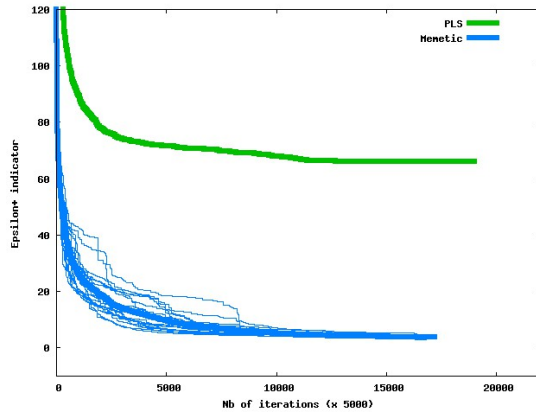
Fig. 9

ε-INDICATOR EVOLUTION FOR THE MEMETIC AND THE REFERENCE PLS

Figure 9 shows the comparison on 20 runs between our Memetic algorithm and the reference algorithm PLS. The results given by the Memetic algorithm seem to be very promising. The curves revel that after only few minutes of computation the hybrid algorithm give better results than PLS after 2 hours according to the selected indicator.

6. Conclusion and future works

We have proposed in this article three main contributions. The first one is about a memetic approach for solving multiobjective problems. This algorithm hybridizes a GA based on NSGA-II and a local search like FIHC. The second contribution is a new way for modeling the problem of locating stations for a Carsharing service. This model is based on a dynamic knowledge of how people move on the territory during the day. To solve this problem in a real context we have proposed three objectives to optimize:

- to increase the number of people using the service
- to reduce the jockeying effort to alleviate the number of rejected demands
- to increase the use of the service all during the day

The last main contribution is about a decision making tool which shows in a same view the decision and criteria spaces. This interactive tool allows the users to understand interactions between the solutions and their evaluation.

The evaluation of our algorithm was compared to the PLS algorithm results for three indicators. The main important aspect is that even if PLS provides nearly twice more non-dominated results, these solutions are in a small part of the criteria space. The algorithm we proposed provides a very good diversification and intensification, in a comparative computation time.

These results confirms that hybridizing GA and local search is a very promising way for future researches.

As a perspective it seems interesting to apply a Pareto based local search such as PLS or FLSMO on the archive set at the end of the memetic algorithm. This new hybridization could complete the archive by finding new solutions in the neighborhood of the non-dominated solutions.

References

[1] C. Coello and G. Lamont, *Applications of multi-objective evolutionary algorithms*, vol. 1. World Scientific Publishing Company Incorporated, 2004.

[2] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proceedings of the 1st International Conference on Genetic Algorithms*, (Hillsdale, NJ, USA), pp. 93–100, L. Erlbaum Associates Inc., 1985.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-2," *Evolutionary Computation, IEEE Transactions on*, vol. 6, pp. 182 –197, apr 2002.

[4] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *Evolutionary Computation, IEEE Transactions on*, vol. 3, pp. 257 –271, nov 1999.

[5] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," *TIK-Report 103*, 2001.

[6] L. Paquete, M. Chiarandini, and T. Stützle, "Pareto local optimum sets in the biobjective traveling salesman problem: An experimental study," in *Metaheuristics for Multiobjective Optimisation* (X. Gandibleux, M. Sevaux, K. Sörensen, V. T'kindt, G. Fandel, and W. Trockel, eds.), vol. 535 of *Lecture Notes in Economics and Mathematical Systems*, pp. 177–199, Springer Berlin Heidelberg, 2004.

[7] L. Moalic, A. Caminada, and S. Lamrous, "A fast local search approach for multiobjective problems," in *LION*, (Catania, Italie), january 2013.

[8] Z. Wu and T. S. Chow, "A local multiobjective optimization algorithm using neighborhood field," *Structural and Multidisciplinary Optimization*, pp. 1–18, 2012.

[9] X. Gandibleux and A. Freville, "Tabu search based procedure for solving the 0-1 multiobjective knapsack problem: The two objectives case," *Journal of Heuristics*, vol. 6, pp. 361–383, 2000. 10.1023/A:1009682532542.

[10] M. P. Hansen, "Tabu search for multiobjective optimization: Mots," in *MCDM'97*, Springer-Verlag, 1997.

[11] A. Liefooghe, J. Humeau, S. Mesmoudi, L. Jourdan, and E.-G. Talbi, "On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems," *Journal of Heuristics*, vol. 18, pp. 317–352, 2012. 10.1007/s10732-011-9181-3.

[12] P. Galinier and J.-K. Hao, "Hybrid evolutionary algorithms for graph coloring," *Journal of Combinatorial Optimization*, vol. 3, no. 4, pp. 379–397, 1999.

[13] J. Knowles and D. Corne, "M-paes: a memetic algorithm for multiobjective optimization," in *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, vol. 1, pp. 325 –332 vol.1, 2000.

[14] A. Jaszkiewicz, "Genetic local search for multi-objective combinatorial optimization," *European Journal of Operational Research*, vol. 137, no. 1, pp. 50 – 71, 2002.

[15] K. Deb and T. Goel, "Controlled elitist non-dominated sorting genetic algorithms for better convergence," in *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, EMO '01, (London, UK, UK), pp. 67–81, Springer-Verlag, 2001.

[16] S. A. Shaheen and A. P. Cohen, "Worldwide carsharing growth: An international comparison," 2008.

[17] G. H. de Almeida Correia and A. P. Antunes, "Optimization approach to depot location and trip selection in one-way carsharing systems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 1, pp. 233 – 247, 2012.

[18] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, and V. da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *Evolutionary Computation, IEEE Transactions on*, vol. 7, pp. 117 – 132, april 2003.

[19] H. Meunier, E.-G. Talbi, and P. Reininger, "A multiobjective genetic algorithm for radio network optimization," in *In Proceedings of the 2000 Congress on Evolutionary Computation*, pp. 317–324, IEEE Press, 2000.

Self-Adaptive Feature Selection Strategy for Multi-type Object Classification Problems

Namdar Mogharreban,   **Tianyang Wang**

namdar@cs.siu.edu,   toseattle@siu.edu

Department of Computer Science, Southern Illinois University Carbondale, Illinois 62901

Abstract *This paper introduces the main idea and method of self-adaptive feature selection strategy in multi-type object classification problems. The limitations of the traditional methods in their application to the real situations are discussed. A new strategy for feature selection combining the traditional methods and a heuristic approach is presented. The result of feature selection using the new strategy and the traditional ones in a real world agricultural setting are compared. The new strategy not only improves the feature selection and classification, it is also applicable to a variety of objects classification. This paper also discusses the future application of this feature selection strategy.*

Keywords: Feature selection, Multi-types object classification, heuristic method, real application

1 Introduction

The concept of Classification is used very broadly in many fields. While the traditional classification methods using a single subset of features to classify objects have been very successful and improved, less attention has been paid to classification of objects into multiple categories where each category has a different efficient set of features associated with it. In this paper, we propose and demonstrate using real data from agriculture, a unique and novel approach to combining a modified heuristic method for classifying objects of multiple categories with differing subset of features.

1.1 Feature selection methods

Feature selection methods are categorized into two approaches, Filter and Wrapper [1]. Filter methods choose the features independent of the classification process whereas wrapper methods do the feature selection depending on the result of classification process. The objective of the proposed approach is to enhance the classification accuracy; therefore, a strategy based on the wrapper method is appropriate since the awareness of the classification result, during the feature selection, is essential.

There are several traditional heuristic feature selection methods in practice, which includes forward, backward, and hybrid [2]. These methods are usually used with corresponding evaluation function [3]. This is

the criterion of removing or adding features to the subset. In general, the pure heuristic methods will not get the global optimal solution; however, sometimes it will obtain a local optimal solution. With our proposed method, we intend to avoid this local optimization. So we combine a modified backward method with our new strategy to achieve optimum classification based on multiple subsets of features.

1.2 Limitation of the traditional methods

There is several probability and information theory based methods utilized in feature selection. For example, the Gain Ratio method, the Taguchi method, the correlation based method and the Chi-square feature evaluation [4, 5, 6, 7, 10, 11, 19]. The concept "entropy" is critical in information theory describing the pure level of classification. An entropy value is determined that will be changed by removing some features, smaller the entropy value, the better. Gain Ratio is an example of a method using entropy to calculate the information gain. In the Taguchi method, the average improvement of performance is obtained by removing each specific feature. Based on this average value, inclusion or exclusion of a feature is determined. Many of the above mentioned probability based methods, as well as correlation-based and principle component analysis [12] methods, are not suitable for the kind of problems at hand [14]. For these methods to be effective, the features are expected to have some internal relationships, based on which correlation probability can be calculated. However, in the type of problems we are addressing the features are independent and no internal relationship exists.

We propose combining a modified heuristic backward method with an accuracy evaluation to select a subset of features for each specific type for the classification.

2 Implementation

In this implementation, there are 33 types of soybeans obtained from six different provinces of China. For the purpose of classification, each category is labeled by the region it is from, as depicted in Table 1. For instance, "Liaoning-1" represents the soybeans from specific area 1, in the Liaoning province, thus creating a total of 33 different types.

Liaoning-	1	2	3	4	5	6	
Heilongjiang-	1	2	3	4	5	6	7
Jilin-	1	2	3	4	5	6	
Henan-	1	2	3	4	5	6	7
Shandong-	1	2	3	4	5	6	7

Table 1: Different types of soybeans from different areas

The quality characteristics of the cultivated soy beans depend on a variety of factors. These can be broadly categorized as environmental factors such as soil temperature or soil acidity; intrinsic factors such as hardness or diameter of the bean, and cultivation factors such as irrigation cycle or fertilization level.

Each combination of specific factors influences the type of soybean differently and each factor is considered as a feature of the soybeans. While a particular combination of features might be desirable for one type, a different subset of features might be considered important in another type of soybeans.

In the proposed application, we need to separate out types of soybeans from a big group

which contains several types of soybeans, each type being defined by a different set of factors. One approach can be to calculate the similarity between the new given soybean and the training sample soybean. However, if we calculate it directly, the accuracy would not be satisfactory since not all the features contribute the same to each category. For example, if feature A is imperative for classifying "Liaoning-1" soybean, that means feature A must be kept for classification of the "Liaoning-1" type, however, this feature may not be significant in classification for the "Henan-5" type. That is to say feature A is a negative feature because it can reduce the classification accuracy for the "Henan-5" type. This indicates that for a more accurate classification, we should remove some features for each type of soybean, resulting in a subset of features that potentially is different for each type of soybean. In contrast to the traditional methods where classification is done based on only one subset of features, the proposed method applies classification based on a differing subset of features for each object. This part will be discussed in section 3, in detail. Once a suitable subset of features has been identified for each object, a similarity degree can be calculated using a variety of methods such as Naive Bayes [16], Knn, and ML, which will produce a better result for the classification.

2.1 Self adaptive feature selection strategy

In light of the limitations of the traditional methods to resolve the multi-type classification with differing subsets of features, we propose the new strategy to pick out the features which bring negative influence on classification and subsequently remove these features from the subset for each type. From [13], we determine that a negative influence means a feature that

reduces the accuracy of the classification. By identifying the negative features and removing them, a different subset is created for each type for the purpose of classification. This strategy displays the function as being self-adaptive, meaning the feature selection result is different among different types, and each type will choose its own feature subset to carry on the classification operation. The new strategy is expressed formally below,

For each T_i ,

    1. for each $f_i \in FS_i$:

$$Acc > ori\_Acc \text{ with } FS_i - \{f_i\},$$

$$ori\_Acc = Acc, \text{ go to step 2,}$$

otherwise go to step 3;

    2. $FS_i = FS_i - \{f_i\}$,go to step 1;

    3. selection is done for T_i .

Where T_i is the type, f_i is the feature in the current feature subset FS_i , which is initialized to be the set containing all the features, Acc represents the classification accuracy with the current feature subset FS_i and ori_Acc is the classification accuracy with the feature subset we have in the last step, that is initialized to be the original feature subset containing all the features.

On the other hand, in order to improve the feature selection efficiency, we incorporate a modified backward method to do the feature selection for each type. We express it formally on the following page.

For each f_i in FS_i :

1. calculate Acc with each

$FS_i = FS_i - \{f_i\}$;

2. remove f_i that make $FS_i = FS_i - \{f_i\}$

lead to the largest Acc

3. if the largest $Acc > ori\_Acc$,

$ori\_Acc = Acc$, go to step 1, otherwise,

selection is done for the current type.

3   Experiment and Results

Soybean data from five different provinces in China was used in our experiments. The experiment was performed in two steps. In the first step, we treat the 33 data sets as the training sets which contain 50 individual samples of each type. For example, the "Liaoning-1" type has 50 individual soybeans. In other words, for each type of training set, we have 50 samples and 18 features. In order to test our new strategy, in the second step, we create new prediction sets which are generated based on the mean value and covariance value of the training sets. The experiments were run 20 times under each different condition and the average result of correct classification for each type was recorded. The standard ML classifier was used for this process. The features are extracted from all of the soybeans and saved in corresponding matrix in mat-lab, version 2010 b. Since this paper only focuses on feature selection, we will not discuss the sample collection and feature extraction part.

There are 18 features extracted and these are depicted in table below in different categories.

Environmental		Intrinsic		Cultivation	
1	Temperature	7	Hardness	13	irrigation cycle
2	soil acidity	8	Volume	14	Irrigation level
3	Precipitation	9	Diameter	15	fertilization cycle
4	frost degree	10	maturity period	16	fertilization level
5	insect suffering	11	water absorption	17	insecticides cycle
6	sunshine	12	Translucency	18	insecticides level

Since we have a total of 33 training sets, we generate a corresponding prediction set for each training set, resulting in 33 prediction sets. Then we set the number of individual soybeans in each prediction set to 100. Theoretically speaking, all the individual soybeans in prediction sets are supposed to be of the same type with the corresponding training sets. However, during the course of running the experiment, it was discovered that several soybeans are not recognized as the correct type. This leads to an unsatisfactory classification result, totaling to 3059 correct classifications. The experiment was repeated applying a heuristic backward method presented by [15] with only one less feature subset applied. The features 10, 12, or 13 were removed, resulting in 3131 correct individual soybean classifications. Applying the proposed self-adaptive method and repeating the experiment resulted in 3266 correct soybean classifications. Figure 1, on the following page, represents the application of the three different methods and the accuracy of the classification.
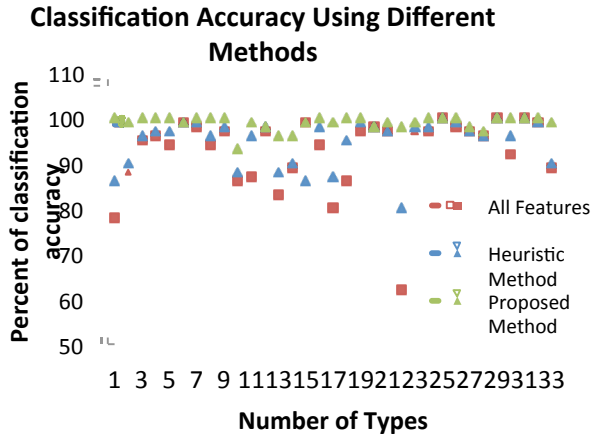
Figure 1: Accuracy of Classification for each type

Table 2 shows the features that were removed for each type in the purpose of classification which arrives at the optimal result obtained by our method. Cells indicating "none" mean no features were selected for removal.

Liaoning-	1 e3,c15	2 e3,e4,i10	3 e1,e3,i12	4 c18	5 e3	6 none	
Heilongjiang-	1 i8	2 e1,i12	3 e3,i8	4 c18	5 i12,c13,c14	6 e2	7 c18
Jilin-	1 i7,c16,c18	2 none	3 e6	4 e3,i11,i12	5 e6	6 e2,i12	
Henan-	1 none	2 e3, i11	3 e2,e3,i7,i10	4 e1	5 e5,i12	6 none	7 i11
Shandong-	1 i11	2 c18	3 none	4 i7	5 none	6 e2,e3	7 e3,i11,c13

Table 2: Types and the corresponding features removed for optimal classification

4   Discussion

Some existing feature selection methods are effective for selecting out the unique feature subset for the specific type of objects, such as the SVM based kernel-penalized method and the Taguchi method. To obtain an improved classification based on a multiple subset of features, we presented a new method combining a modified backward process to do the feature selection with the ML classifier in the classification. The results indicate an improvement in the accuracy of the classification moving from applying the traditional method with ML to applying our self-adaptive approach. By selecting a different subset of features for classification, the proposed method produces much higher accuracy. An added significance to the new strategy is that it can be applied in different kinds of problems with multiple types or classes. For each type, or class, a feature subset is built which is different from those associated with other types or classes. This ensures that the feature selection process for each type will select the most effective features for the

current type, removing any influence of the current feature subset on the other types being classified.

5   References

[1] Ron Kohavi, George H. John. Wrappers for Feature Subset Selection. Artificial Intelligence. 1997

[2] Wang Juan, Ci lin-lin, Yao Kang-ze. A Survey of Feature Selection. Computer Engineering & Science, Vol. 27, No.12, 2005

[3] Andreas G.K. Janecek, et al. On the Relationship Between Feature Selection and Classification Accuracy. JMLR proceedings 4, 90-105. 2008

[4] Shyamala Doraisamy, et al. A study on Feature Selection and Classification Techniques for Automatic Genre Classification of Traditional Malay Music. ISMIR proceedings-Session 3a. 2008

[5] Nojun Kwak, et al. Input Features Selection for Classification Problems. IEEE Trans on Neural Networks, Vol.13, No.1. 2002

[6] Lei Yu, Huan Liu. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. Proceedings of ICML. 2003

[7] Huiqing Liu, et al. A Comparative Study on Feature Selection and Classification Methods Using Gene Expression Profiles and Proteomic Patterns. Genome Informatics 13 : 51-60. 2002

[8] Sebastian Maldonado, et al. Simultaneous feature selection and classification using kernel-penalized support vector machines. Information Sciences 181, 115-128. 2011

[9] Jianfeng Cai, et al. A Singular Value Thresholding Algorithm for Matrix Completion. 2008.

[10] Krzysztof Michalak, et al. Correlation-based features selection strategy in classification problems. Int. J. Appl. Math. Comput. Sci., Vol. 16, No. 4, 503-511. 2006

[11] Huan Liu, Lei Yu. Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE trans on knowledge and data engineering, Vol. 17, No. 4. 2005

[12] Lindsay I Smith. A tutorial on Principal Components Analysis. 2002.

[13] Hanchuan Peng, et al. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy, IEEE Trans on pattern analysis and machine intelligence, Vol 27, No.8. 2005

[14] Daniel P. W. Ellis, et al. Using Mutual Information to Design Feature Combinations.

[15] Shigeo Abe. Modified Backward Feature Selection by Cross Validation. Proceedings of European Symposium on Artificial Neural Networks, 27-29 April 2005,

[16] M.L.Zhang, et al. Feature Selection for Multi-label Naïve Bayes Classification. Information Sciences, 179(19) (2009) 3218-3229.

[17] J. Neumann, C. Schnörr, G. Steidl, Combined SVM-based feature selection and classification, Machine Learning 61 (1–3) (2005) 129–150.

[18] Yi-Wei Chen and Chih-Jen Lin. Combining SVMs with Various Feature Selection Strategies. Proceedings of NIPS. 2003.

[19] Krzysztof Michalak, Halina Kwasnicka. Correlation-Based Feature Selection Strategy In

Syntactic Parameters in the Phrasal Machine Translation

Elena Kozerenko, Alexander Khoroshilov, Alexei A. Khoroshilov
Institute of Informatics Problems of the Russian Academy of Sciences, Moscow, Russia

Abstract - *The paper deals with the methods and techniques of representing syntactic parse rules within the system of parameters of the example-based machine translation framework largely based on automatically compiled set-phrase dictionaries and translation memory. The syntactic structures are introduced into the vocabulary entries of the machine translation system.*

Keywords: machine translation, syntax, semantics, set-phrase dictionaries, machine learning, translation memory

1   Introduction

The main objective of the research and development presented in the paper is establishing the mechanisms for syntactic parsing in the systems based on previously translated texts, i.e. translation memory and example-based machine translation.

The concept of translation memory (sentence memory) appeared as an alternative to traditional machine translation. That concept can be regarded as an attempt to realize the idea of the Japanese computer scientist Makoto Nagao, that in the process of machine translation it is necessary to use the large corpora of parallel texts, earlier translated by humans. A more adequate approach is based on the concept of statistical machine translation (statistical-based machine translation), which is defined by some authors as a "sort of machine translation of texts, based on comparing of large corpora of language pairs". In contrast to traditional machine translation, statistical approach is based on statistical computation of matching probability and does not use the linguistic algorithms. Large corpora of parallel texts are necessary for operation of this system. A statistical mechanism of text analysis is used in the process of translation. This mechanism allows to select the variant for the word combination translation based on matching frequency of the language pair elements. A weak point of statistical systems is partial or total absence of a mechanism of grammatical rules analysis for source and target languages. Therefore, it is hard to imagine that the system, which does not analyze the text from the point of view of grammar, is able to release the correct translation of semantically complex texts. The systems of machine translation of texts simulate operation of a human translator. Their efficiency depends on how the nature of language operation and cognition are taken into account, and this nature has not yet been adequately studied. Therefore, the developers of machine translation must take into account the experience in international communication and translation activity, that was accumulated by mankind.

The starting point of machine translation was marked by the word-wise approaches. In this case, single words were considered to be basic units of sense expressing the concepts, and the sense of larger speech units (word combinations, phrases and utterance-length units) was supposed to be determined on the base of the sense of words comprising them. In dictionaries the use of word combinations along with single words was also admitted. But these combinations were mainly the idiomatic expressions, and their amount in dictionaries of machine translation systems was negligible in comparison with the amount of single words. That experience testifies that in the process of text translation the phraseological word combinations expressing the concepts rather than single words are the basic units of sense. The concepts are the elementary intellectual images, by the use of which it is possible to create more complex intellectual images corresponding to translated text.

2   Methods and limitations of the set-phrase machine translation and statistical approaches

The main thesis of the set-phrase translation is a statement that the concept names in texts are determined by word combinations rather than single words [1-6]. The meaning of units of higher level cannot be fully reduced to the sum of meanings of lower level units comprising them.
Set-phrase translation employs single-step compiling of bilingual frequency dictionaries of words and set-phrase word combinations. The orientation at the semantic-syntactic and mainly word-by-word translation alone could not lead to the solution of the basic problems of machine translation, because within language and speech the sense of units of higher level, as a rule, cannot be reduced or fully reduced to the sense of the lower level units comprising them. Almost all known systems related to traditional machine translation systems, developed in that direction. Later the developers of traditional systems began to include more terminological word combinations into their dictionaries.
A weak point of statistical systems is partial or total absence of a mechanism of grammatical rules analysis for source and target languages. A system which does not analyze the text from the point of view of grammar is unable to release the correct translation of semantically complex texts. Statistical machine translation allows to select the variant for the word combination translation based on matching frequency of the language pair elements. The main thesis of the set-phrase translation is a statement that the concept names in texts are determined by word combinations rather than single words, which is not always true. Set-phrase translation is mainly founded on a single-step compiling of bilingual frequency dictionaries of words and set-phrase word combinations.

The idea to create machine translation systems on the base of previously translated texts can be realized in different ways. The *first way* provides text translation with the use of statistic analysis of large corpora of bilingual texts in the

process of translation. This way is known as "statistical machine translation" [7-14, 17,18].

In statistical machine translation (SMT) the task of translating from one natural language into another is treated as a machine learning problem. This means that via training on a very large number of hand-made translation samples the SMT algorithms master the rules of translation automatically. The first SMT developments were presented in [7,8]. The existing methods basically employ either sentence alignment or word alignment some experiments are made with phrase alignment and recently a mixed sentence-word approach has been developed to explore the paraphrases in the aligned parallel corpora. These attempts to consider linguistic information mark a step forward to acknowledging the intricate character of natural language if compared with other types of data. The mixed approach employs both sentence and word alignments. However, all these methods deal with the structural elements without considering the semantic aspects of the aligned language units.

The *second way* is connected with a single-step compiling of bilingual frequency dictionaries of words and phraseological word combinations. The creators of the systems of phraseological machine translations follow the second way [1-6]. This way excludes the fatal dependence of the translation process on availability of large volumes of parallel texts and quality of their translation.

Since the systems of phraseological machine translation are based on the theoretical concept, the main thesis of which is a statement that the concept names in texts are determined by word combinations rather than single words. Therefore, in the process of text translation from one language into another, it is necessary to use the phraseological combinations expressing the concepts, relationships between concepts and the typical situations rather than single words as basic units of sense. The single words may also be used if the translation with the help of the phraseological word combinations fails.

In compliance with this thesis, the system of phraseological machine translation must comprise the knowledge base of translation equivalents for most frequent phrases, phraseological combinations and single words. In the process of text translation the system should use the translation equivalents stored in its knowledge base in the following order: at first, an attempt to translate the successive sentence of the source text as the integral phraseological unit is made; then, if this attempt fails, words combinations being a part of the sentence should be translated; and, finally, if both above-mentioned attempts fail, word-by-word translation of the text fragments is performed. The fragments of the target text translated with the use of all three approaches, must grammatically agree with one another with the help of procedures of morphological and syntactic synthesis). Let us give consideration to this concept in detail. It is necessary to apply the following principles in the process of development of phraseological machine translation systems:

1. The phraseological units (word combinations and phrases) are basic language and speech units, which should be primarily included in the computerized dictionary.

2. Along with the phraseological units composed of continual word sequences, so called "speech models" - phraseological units with blank spaces, that may be filled with different words and word combinations, generating meaningful segments of speech, may be used in machine translation systems.

3. Real texts, without regard to their subject area, tend to be polythematic, if they have sufficiently large size. These texts differ from each other not so much by word stock as by probability distribution of occurrence of different words and word combinations from national word stock in them. Therefore, the computerized dictionary designed for translation of the text belonging to a single subject area must be polythematic, not to speak of translation of texts belonging to different subject areas.

4. Systems of phraseological translation need high-volume computerized dictionaries. Such dictionaries should be created on the base of computer-aided processing of parallel texts - bilingual texts, which are translations of each other, and in the process of translation system operation .

5. Along with the main high-volume polythematic dictionary, it is also reasonable to use a set of additional small-volume highly specialized dictionaries in systems of phraseological machine translation. The additional dictionaries should only contain information missing from the main dictionary (for example, data on priority translation equivalents of word combinations and words for different subject areas, if these equivalents are not equal to priority translation equivalents of the main dictionary).

6. The main means for solution of the problem of words polysemy in phraseological translation systems is their use in phraseological word combinations. The additional means is a set of additional specialized dictionaries , where the priority translation equivalent specific for subject area in question is identified for each multiple-meaning word or word combination.

7. The procedures of morphological and syntactic analysis and synthesis of texts, that are built on the base of linguistic analogy may play a major role in the systems of phraseological machine translation of texts.. These procedures allow to give up storing large amounts of grammatical information in dictionaries and generate it automatically in the process of translation when the need arises. They make the translation system open and capable of processing the texts with "new" words.

Along with text translation in automatic mode, it is reasonable to provide for an interactive mode of operation for the systems of phraseological machine translation. In that mode the user should have potentiality to intervene in the translation process and adapt the additional computerized dictionaries for the subject area of the translated text.

The phrase-based translation model, or the alignment template model and other similar approaches have greatly advanced the development of machine translation technology due to the extension of the basic translation units from words to phrases, i.e. the substrings of arbitrary size.

However, the phrases of the statistical machine translation model are not the phrases in the meaning of any existing syntax theory or grammar formalism, thus, for example, a phrase can be like "alignments the", etc. A real challenge is the cross-level (e.g. morphology-to-syntax) matching of language structures in parallel texts. New research and development results demonstrate the growing awareness of the demand for enhancing linguistic motivation

in statistical translation models and machine learning techniques [17,18].

3   Adult learning memory metaphor

In contrast to the approaches on the basis of "translation memory" that provide the increase of a machine translation system language competence by accumulating the previously translated text fragments and mainly based on regular expressions, Cognitive Transfer Grammar - CTG [15,16] is intended for the realization of the mechanism of structural memory, which simulates language competence of an adult learner ("Adult Learning Memory"). Thus, structural memory comprises the following components:

1) The initial basic collection of grammar rules represented in the formalized form (CTG);

2) The mechanisms of expansion and refinement of the system of rules, implemented by means of the methods of machine learning on parallel texts.

Our studies are based on the concepts of the functional approach, which we have used for the multilingual situation. With the development of the linguistic processor, which ensures English - Russian and Russian - English transfer, we introduced the concept of functional transfer fields (FTF) [16] that served the basis for the segmentation of language structures for the solution of machine translation problems. The basic idea of FTF consists in the adoption of the hypothesis about the fact that at the basis of grammatical structures there lie the cognitive structures (mental frames); a functional transfer field reflects the interaction of elements from different language levels.

"Adult learning memory" (ALM) means the employment of the "adult rule kit" a starter set of about 300 rules stating the structural semantic correspondences between source and target languages.

The machine translation technique comprises analysis, transfer and generation across the functional – categorial values of language units [15,16].

The process of structural patterns recognition is performed basing on the multiple transfer rule set and the probabilistic functional tree substitution grammar.

Translation activity involves the search for equivalence between structures of different languages. However, to establish whether the structures and units are equal or not, we need some general equivalent against which the language phenomena would be matched. Our approach based on the principle "from the meaning to the form" focusing on Functional Syntax would yield the necessary basis for equivalence search.

Consider some statistically relevant examples. Sometimes, a word may be translated by a word of another part-of-speech in the target language, a word combination, or even a clause, as the English word *implementable* is best translated into Russian as *kotoryi vozmozhno realizovat* (*which can be implemented*). To overcome these differences the categorial and functional features of the two languages were considered, and the structures of the input were made conformed to the rules of the target language by applying contrastive linguistic knowledge for implementation of the transfer model. A suitable formalism is indispensable for an algorithmic presentation of the established language transfer

rules, and the language of Cognitive Transfer Structures (CTS) was developed based on rational mechanisms for language structures generation and feature unification [15].

4   Syntactic rules in digital dictionaries

First of all, the systems of phraseological machine translation should be aimed at translation of texts on business, science, technologies, politics and economy. Translation of fiction, artistic prose and poetic texts is a more complex and challenging task. But success can be also achieved in this area in future, if modern technological means are used to compile huge phraseological dictionaries for these texts.

Set-phrase units (word combinations and phrases) are basic language and speech units which should be primarily included in the computerized dictionary.

Along with the set-phrase units composed of continual word sequences, so called "speech models" - set-phrase units with slots that may be filled with different words and word combinations, generating meaningful segments of speech, can be used in machine translation systems.

At first glance, the machine translation concept offered by professor Makoto Nagao in 1984, fundamentally differs from the concept, formulated by professor G. G. Belonogov nine years earlier. But this is not true. Indeed, in the process of practical realization of Makoto Nagao's concept, it is difficult to imagine that the text written in any language is completely the same as another text written earlier and translated into foreign language. It is not to be expected that this text contains long fragments (chapters, paragraphs and etc.), that are the same as the fragments of the text written and translated earlier. But, as our investigations showed, the continuous texts fragments including over ten words repeat on rare occasions - their total frequency doesn't exceed 1%. It is necessary to use only short sentences, single words and text fragments (word combinations) including less than 10-12 words. This is the semantic-syntactic phraseological translation.

Of course, along with the translation equivalents of the relatively short fragments of texts, it is possible to include the translation equivalents of longer fragments in the computerized dictionaries. But in this case one should keep in mind, that the computerized dictionaries will be filled with "dead" ballast, i.e. with the dictionary entries, which will be used on rare occasions or will not be used at all in the process of text translation.

When developing the systems of phraseological machine translation, the most difficult and time-consuming problem appears to be the on of compiling sufficiently high-volume computerized dictionaries. The quality of translation depends on the volume of these dictionaries and on the quantity of the phraseological word combinations in them. And those volumes needs to be sufficiently large to provide the good covering of texts.

It is known, that in modern languages of the world (for example, in Russian or English) the amount of different words exceeds one million, and the amount of concept names determined by word combinations exceeds hundreds of millions. The authors of this article came to this conclusion on the basis of many years' experience of the statistical study of texts. Confirmation of such viewpoint is the report of the All-

European terminological centre "Infoterm" (Vienna, Austria, 1998), in which it was found that in modern languages of the world, such as English and German, a total amount of different terms exceeds 50 million, and nomenclature of goods exceeds 100 million. It is well known, that the connected texts consist of not only terms and names of goods.

The computerized dictionaries of such volume cannot be created quickly, but as experience shows, it is possible to achieve satisfactory quality of translation at the first stage in the presence of only several million entries in dictionaries, at least 80% of which should be word combinations. In this case the polithematic texts have the coverage of about 99, 7%.

Thereafter, the volume of dictionaries must be constantly increased and with the growth in amount of phraseological combinations, the quality of machine translation should improve. This problem cannot be solved by manual methods. For its solution, a system of computerized compiling and maintenance of the computerized dictionaries was created.

5   Semantic-syntactic set-phrase translation

It is necessary to use only short sentences, single words and text fragments (word combinations) including less than 10-12 words .

Along with the main high-volume polythematic dictionary, it is also reasonable to use a set of additional small-volume highly specialized dictionaries

The procedures of morphological and syntactic analysis and synthesis of texts, that are built on the base of *linguistic analogy* may play a major role in the systems of set-phrase machine translation of texts.

The implementation of the computerized phraseological text translation from one language into another must have three stages. At the first stage, the semantic-syntactic analysis of the source text is carried out. During that analysis the text is split into sentences, and then their conceptual and syntactic structure is determined. At the second stage (at the transfer stage) the concept names of the source text are substituted by the concept names in target language and the information on the syntactic structure of the source text is transformed into information required for the target text synthesis. At the final stage (the stage of semantic-syntactic synthesis of the target text) the text in the target language is formed.

The stages listed above are present in the process of translation of texts from any language to any other language, but their particular content for different pairs of languages has a specific character. This specific character can be seen in procedures of semantic-syntactic analysis and synthesis of texts, which include the procedures of morphological, syntactic and conceptual analysis and synthesis [1-6].

The set-phrase machine translation based on the multilingual dictionaries will operate in the same way, but these systems should be complemented with the procedures of semantic-syntactic and conceptual analysis and synthesis of all languages, which will be included in the system. The authors of this paper developed the effective technology based on the use of principles of linguistic analogy for creation of these procedures.

The computerized dictionaries are the most important part of the systems of phraseological machine translation. They should have sufficiently large volume, in order to cover texts, and should contain mainly word combinations. The authors developed the original methods, algorithms and programs for automated compiling and maintaining dictionaries for the system of phraseological machine translation. In cooperation with other specialists, the large-volume Russian-English and English-Russian phraseological computerized dictionaries containing 2, 6 million dictionary entries each were compiled. These dictionaries cover 99, 7% of the lexical content of modern texts and they represent the powerful bilingual conceptual model for a wide range of fields of human activity. Phrase structures rules are incorporated into the vocabulary entries.

6   Well-formed nonterminals and dynamic rules

The main method for syntactic model enhancement in the set-phrase machine translation is including well-formed non-terminals in the general system of sentence analysis.

The non-terminals constitute the complete parse tree of a sentence comprising set-phrase models.

The dynamic formation of syntactic structures is supported by alternative categorial grammar parse on the basis of the rules dynamically extracted from parallel corpora.

Actually the process of transfer goes across the functional – categorial values of language units. A language structure which can be subjected to transfer has to be semantically complete from the point of view of its function. The cases of categorial shifts, in particular, when the technique of conversion is employed, require special treatment: the categorial shift of a syntax unit is determined by the functional role of this unit in a sentence (e.g. noun as a modifier is transformed into adjective).

The experience in creation of the large-volume Russian-English and English-Russian computerized dictionaries convinced the authors that Russian and English texts which are translations tino each other (for example, bilingual titles of the documents), can serve as the most reliable source for dictionaries compiling.

The compiling of the computerized dictionaries with the use of bilingual texts was carried out both manually and with the assistance of computers. The manual dictionary making requires huge expenditures of human labour. Therefore the authors of the article developed the procedure for automated dictionary making [3]. This procedure is based on the hypothesis, that in numerous bilingual pairs of sentences, which are translations of each other and which contain the same word or word combination of one of the languages, the word or word combination of another language, which is the translation of this word or word combination has maximal frequency. of occurence.

The procedure was used for processing bilingual (Russian and English) titles of the documents from the databases of VINITI (All-Union Scientific and Technical

Information Institute). In this case more than one million pairs of the document titles were processed. The computerized dictionaries of MetaPhrase system can be corrected and completed in the process of text translation in the interactive mode. In that mode there is an opportunity to identify the words and word combinations, which have no translation equivalents in the dictionary or these equivalents do not comply with the context or several equivalents are given, but the first equivalent does not comply with the context. These equivalents can be replaced by the equivalents complying with the textual context.

7   System performance

In compliance with the method described above, the large-scale experiment on compiling English-Russian frequency dictionaries on the base of the automated concept analysis of English and Russian titles of the documents, which are translations to each other, was carried out. For this purpose, the corpus of English titles of the polythematic documents and their Russian translations having the volume of about 2 million pairs of sentences from the VINITI's databases (1994-1999) were processed. The total volume of the corpus of texts is 390 Mb.

In the process of research three English-Russian frequency dictionaries were created:

1) the dictionary comprising the items which are the combinations of fragments of English and Russian titles of documents between which the translation equivalents were determined with the assistance of MetaPhrase system;

2) the dictionary comprising the items which are the fragments of titles of documents between which the translation equivalents have not been determined, but they are surrounded by the other fragments, between which such equivalents have been determined or by the signs of the beginning or the end of the title;

3) the dictionary comprising the items which are fragments of the English and Russian titles of documents between which the translation equivalents have been determined on the initial stage of titles processing.

The first frequency dictionary includes bilingual phraseological word combinations containing 2 to 16 words. It had 3.127.363 dictionary entries.

The value of the dictionary in question is that it contains translation equivalents between English and Russian fragments of titles of documents, which are longer than their fragments selected at the first stage of conceptual analysis of the titles. Each of the newly formed dictionary entries practically has just one translation version of an English word combination (the percentage of dictionary entries having more than one version of translation is less than 0.1).

The second frequency dictionary includes translation equivalents between fragments of titles of documents, that were not found at the initial stage of conceptual analysis of these titles. This dictionary contains 1.825.612 dictionary entries. The most frequent dictionary entries have the frequency of 1.008, and infrequent dictionary entries have-the frequency equal to one. 87% of dictionary entries have the frequency equal to one. A spot-check of the dictionary showed that about 50% of translation equivalents were incorrect. The quantity of such translation equivalents can be reduced at the final stage of dictionary making, if the procedure of semantic-syntactic checking is applied. After that, the dictionary must be edited by humans.

The third frequency dictionary contains translation equivalents between fragments of English and Russian titles of documents that were found at the initial stage of processing of these titles. It contains 387.025 dictionary entries. The most frequent dictionary entries have the frequency of 4.985, and most infrequent dictionary entries have- the frequency equal to one. 56% of dictionary entries had the frequency equal to one.

8   Digital set-phrase dictionaries

The authors developed the original methods, algorithms and programs for automated compiling and maintaining of the dictionaries for the system of set-phrase machine translation. Russian-English and English-Russian set-phrase dictionaries containing 2, 6 million dictionary entries each have been compiled. These dictionaries cover 99, 7% of the lexical content of modern texts and they represent powerful bilingual conceptual model for a wide range of fields of human activity.The compiling of the digital dictionaries with the use of bilingual texts was carried out both manually and with the assistance of computers. The procedure for automated dictionary making has been developed.

This procedure is based on the hypothesis, that in numerous bilingual pairs of sentences which are translations of each other and which contain the same word or word combination the translation of this word or word combination has maximal occurrence frequency.

The process of establishing concept equivalents is organized in the following way. At first stage, the semantic-syntactic analysis of the source text is carried out: the text is split into sentences and then their conceptual and syntactic structure is determined.

At the second stage (transfer) the concept names of the source text are substituted by the concept names in target language and the information on the syntactic structure of the source text is transformed into information required for the target text synthesis.

At the final stage (semantic-syntactic synthesis of the target text) the text in the target language is formed.

A system which does not analyze the text from the point of view of grammar is unable to release the correct translation of semantically complex texts.

Therefore, it proved to be necessary to introduce the well-formed non-terminals and parse rules into the representation mechanism of the set-phrase translation. It extended the compiling procedure of the dictionary as a means for automated linguistic processing.

9   Conclusion

In conclusion, it should be noted that this article describes the experience in creation of modern multilingual machine translation systems – the systems of phraseological translation. The extensive application of means of automation allowed to essentially reduce expenditures of human labour in

the process of creation of this system, and therefore, to reduce the creation cost of such systems.

The modern multilingual machine translation systems should be based on set-phrase translation enhanced by cognitively and functionally motivated grammar.

The extensive application of means for automation allowed the authors to essentially reduce expenditures of human labour in the process of creation of the machine translation system, and therefore, to reduce the total creation cost of such systems.

Further research and development is connected with the dictionaries expansion and semantic-syntactic structuring.

10 Acknowledgements

11 References

[1] Belonogov, G. G., Khoroshilov, Alexander A., Khoroshilov, Alexei. A. Phraseological Machine Translation of Texts from Natural Languages to Other Natural Languages. . Col **:**"Scientific-Technical Information", Series 2. - M.: VINITI, 2010, № 10.

[2] Belonogov, G. G., Khoroshilov, Alexander A., Khoroshilov, Alexei A. Automatization of compiling of English-Russian bilingual phraseological dictionary using the corpora of bilingual texts. Col**: "**Scientific Technical Information**"**, Series 2. - M **,** VINITI, 2010, № 5.

[3] Belonogov, G. G., Gilyarevskij, R. S., Khoroshilov, A.A. On the nature of information. Col.**:**"Scientific Technical Information**"**, Series 2. - 2009. - № 1.

[4] Belonogov, G. G., Kalinin, Yu. P., Khoroshilov, Alexander A., Khoroshilov, Alexei A.. Systems of Phraseological Machine Translation of Texts. Theoretical Preconditions and Experience in the Development. - M. 200**7.**

[5] Belonogov, G. G., Kalinin, Yu. P., Khoroshilov, A. A. Computational linguistics and Advanced Information Technologies. Theory and Practice of Constructing of Automatic Text Processing Systems. - M. 2004**.**

[6] Belonogov, G. G., Bystrov, I. I., Kozachuk, M V., Novoselov, A. P., Khoroshilov A.A. Automated Conceptual Text Analysis. Col. :**"** Scientific Technical Information**", S**eries 2. - 2002. № 10.

[7] Brown P. F., Cocke J., Della Pietra S. A., Della Pietra V. J., Jelinek F., Lafferty J. D., Mercer R. L., Roossin P. S. A statistical approach to machine translation // Computational Linguistics, 1990. Vol. 16. P. 79–85.

[8] Brown P. F., Della Pietra S. A., Della Pietra V. J., Mercer R. L. The mathematics of statistical machine translation: Parameter estimation // Computational Lin- guistics, 1993. Vol. 19. No. 2. P. 263–311.

[9] Marino J. B., Banchs R. E., Crego J. M., de Gispert A., Lambert P., Fonollosa J. A.R., Costa-Jussa M. R. N-gram-based Machine Translation // Computational Linguis-
tics, 2006. Vol. 32. No. 4. P. 527–549.

[10] Chen S. F. Aligning sentences in bilingual corpora using lexical information //
Proceedings of the 31st Annual Conference of the Association for Computational
Linguistics, 1993. P. 9–16.

[11] Callison-Burch C. Syntactic Constraints on Paraphrases Extracted from Parallel
Corpora // Proceedings of EMNLP-2008. 2008.

[12] Callison-Burch C., Koehn P., Monz C., Schroeder J. Findings of the 2009 Workshop on Statistical Machine Translation // Proceedings of Workshop on Statistical Machine Translation (WMT09), 2009.

[13] Och F. J., Ney H. The alignment template approach to statistical machine transla- tion // Computational Linguistics, 2004. Vol. 30. P. 417–449.

[14] Koehn P. and Hoang H. Factored translation models // Proceedings of the 2007Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). 2007. P. 868–876.

[15] Kozerenko E. Features and Categories Design for the English-Russian Transfer Model // Advances in Natural Language Processing and Applications Research in Computing Science, 2008. Vol. 33. P. 123–138.

[16] Kozerenko E. B. Cognitive Approach to Language Structure Segmentation for Machine Translation Algorithms // Proceedings of the International Conference on Machine Learning, Models, Technologies and Applications, Las Vegas, USA, 2003. — CSREA Press, 2003. P. 49–55.

[17] Wang W., May J., Knight K., and Marcu D. Re-Structuring, Re-Labeling, and Re- Aligning for Syntax-Based Statistical Machine Translation // Computational Lin- guistics, 36(2), 2010.

[18] Zhang H., Gildea D., and Chiang D. Extracting Synchronous Grammar Rules from Word-Level Alignments in Linear Time // Proceedings of the 22nd Interna- tional Conference on Computational Linguistics (COLING-08), Manchester, UK, August 2008.

Functional and Cognitive Aspects in Linguistic Modelling

E. Kozerenko

Institute of Informatics Problems of the Russian Academy of Sciences, Moscow, Russia

Abstract - *The paper focuses on the issues of establishing transferable language phrase structures. The approach employed is based on generalized cognitive entities manifested in the categorial systems of the English and Russian languages and functional roles of language units in a sentence. The formalism developed for presentation of syntactic structures for the English-Russian machine translation is a variant of unification grammar and comprises about three hundred rules. A number of declarative modules of linguistic processors were designed and implemented within the framework of machine translation system "Cognitive Translator" and knowledge extraction systems.*

Keywords: machine translation, syntax, semantics, set-phrase dictionaries, machine learning, translation memory

1   Introduction

To face the problems of language structures transferability for machine translation (MT), it is necessary to consider human translation experience. Translation is a creative and sophisticated human activity, hence, producing automatically a high-quality translation of an arbitrary text from one language to another is a task too far from its complete implementation. However, for simpler tasks, such as acquiring information on the Web, getting acquainted with subject domain information, etc., rough translation output without post editing can be quite adequate. One of the domains where MT works best is scientific discourse. Perhaps, it can be accounted for the regularity of syntactic structures which is required by the functional style of scientific prose.

Of the three forms of translation performed by man: written translation, consecutive interpretation and simultaneous interpretation, the one which is nearest to the real-time machine translation is simultaneous interpretation (SI). Therefore, the recommendations for SI are of prime interest to MT designers, as they propose more implementable solutions for lexical grammatical transformations than the first two forms.

Another important consideration is that some features of human language appear to be of universal character, for example, every language has nouns and verbs. Even the differences of human languages often have systemic structure [1]. Syntactically languages are most different in the basic word order of verbs, subjects, and objects in declarative clauses. English is an SVO language, while Russian has a comparatively flexible word order. The syntactic distinction is connected with a semantic distinction in the way languages map underlying cognitive structures onto language patterns, which should be envisaged in MT implementations [2]. Besides, there exist syntactic constructions specific of a given language (such as, for example, English constructions with

existential "there" and "it" as formal subjects). Sometimes, a word may have translation to a word of another part-of-speech in the target language, a word combination, or even a clause, as the English *implementable* is best translated into Russian as *kotoryi vozmozhno realizovat'* (*which can be implemented*). To overcome these differences the categorial and functional features of the two languages were considered, and structures of the input were made conformed to the rules of the target language by applying contrastive linguistic knowledge for implementation of the transfer model. A suitable formalism is indispensable for an algorithmic presentation of the established language transfer rules, and the language of Cognitive Transfer Structures (CTS) was developed based on rational mechanisms for language structures generation and feature unification.

The application of statistical models has considerably advanced the area of machine translation since the last decade of the previous century, however now new ideas and methods appear aimed at creating systems that efficiently combine symbolic and statistical approaches comprising different models. Both the paradigms move towards each other: more and more linguistics is being introduced into stochastic models of machine translation, and the rule-based systems include statistics into their linguistic rule systems. The procedures of analysis and translation are enhanced by the statistical data, which are taken into consideration by the "translation engine" for disambiguation of language structures. The paper is also focused on discovering the ways of the two research paradigms combination, namely, introducing statistical methods into the rule-based systems of machine translation and employment of the methods and presentations capturing human language intuition in statistical translation models with the view of enhancing the existing language processing technologies.

In statistical machine translation (SMT) the task of translating from one natural language into another is treated as a machine learning problem. This means that via training on a very large number of hand-made translation samples the SMT algorithms master the rules of translation automatically.

2   Establishment of cross-language matches and inter-structural synonymy

Segmentation and unification of utterances in the course of translation is a major task for human professional interpreters. They would even say that syntax is "interpreter's enemy". The selectivity of languages as to the choice of specific characteristics of description of one and the same situation results in numerous distinctions, and one of the most crucial of them is the degree of particularity in conveying a referential situation. Therefore, a situation which in one

language is described by means of one specific feature, in another language may require two or more characteristics. Thus, in many cases the English language is more economical (about thirty percent, according to the reports of simultaneous interpreters) [3,4] in expressing a thought than Russian. A very good illustration of this phenomenon is attributive word combinations of the "stone wall" type which when being translated into Russian in many cases require numerous additions. On the other hand, Russian input in some cases may result in an expanded English translation.

In practice the technique applied to overcome this problem is utterance segmentation which consists in sectioning a source Russian sentence into two or more utterances in the resulting English sentence.

Another important rule is the least possible change of word order. But this inflicts other unavoidable transformations, and not all of them are implementable within the framework of machine translation. For example, the general rule for interpreters: a Russian noun which appears at the very beginning of a sentence and has the form of an oblique case, i.e. indirect object standing at the beginning of a Russian sentence, should be transformed into the subject of an English sentence notwithstanding its initial syntactic role

e.g. *Na vstreche dogovorilis'…(At the meeting agreed…)*

should be translated as -

The meeting reached an agreement…

This transformation performed in the course of human simultaneous interpretation appears to be unattainable to a machine translator at the present state of the art. The requirement of denotational equivalence involves numerous lexical grammatical shifts which cause transformations of the semantic structure of an utterance [3,4]. Another regular semantic shift, that of substituting a predicate of action by the predicate of state.

e.g. *He is a member of the college team. (A predicate of state).*

On igraet v studencheskoi komande. (He plays in the students' team. A predicate of action).

Moreover, the existence of such shifts within the real text corpora inflicts complications for one more computational linguistics problem, that is text alignment, which in some cases may appear even intractable.

The following SI techniques appeared to be of use for MT design in the course of our development.

(1) Full translation of lexical grammatical forms is applied when these forms completely correspond to each other both in the source and the target languages as to their form, function and meaning.

(2) Null translation is applied when a grammatical form exists in the source and target languages but is used differently for explicating a certain referential situation.

(3) Partial translation is used when one and the same grammatical form has several content functions which differ in the source and target languages.

(4) Functional substitution is employed when the functions and meanings of grammatical forms in the source and target languages differ. In that case the source form can be substituted by a form of another type in the target language on the basis of their functional identity.

(5) Assimilation is a device applied for translating grammatical forms constituting compound structure, and the combinability features of these forms differ in the source and target languages.

(6) Conversion is used for substituting a form of one category by a form of another category, and is conditioned by the combinability rules difference in the source and target languages.

(7) Antonyms employment is used for eliminating a conflict between lexical and grammatical combinability of language units in the source and target languages.

Thus it is obvious that the search for equivalence should be carried out starting with the establishment of semantic equivalence of patterns notwithstanding their structural dissimilarity. Pattern-matching approach for the English – Russian transfer was assumed, and the segmentation of structures of the source language was performed on the basis of functional transfer fields which were established via contrastive study of the two languages.

The transformations in focus comprise the following statistically important cases:

- Nominalization;
- Passivization;
- Adjectival – Adverbial structures transformations;
- Subject – Object transformations;
- Indirect Object transformation into Subject;

e.g. *Ser'oznymi raznoglasiyami byla otmechena vstrecha storon – Serious disagreements arose during the meeting of the sides*

Direct Object transformation into Subject;

Prepositional phrase transformation into Subject:

Na vstreche dogovorilis' – The meeting reached the conclusion.

3   Cross-level focus

The machine translation technique employed presupposes three stages: analysis, transfer and generation. The stage of analysis results in parse representing the structure of the input sentences. Transfer is a bridge between the parse structure of the source language and the input to the generation procedure for the target language. At this stage the transformation is performed of one parse tree (applicable for the source language presentation) into another tree (presenting the target language). Thus syntactic transformations imply the mapping of one tree structure to another.

It is very important that a parse for MT differs from parses required for other purposes. Thus the grammar formalisms developed for a unilingual situation (phrase stucture rules systems for the English language) [5] would give an untransferable parse in many crucial situations. For example, just one English phrase structure rule for simple sentence would suffice for grammar parse without translation, but for the English – Russian transfer a multiple structure of possible parses is required depending on the specific finite verbal form constituting the sentence. And to overcome this, an accurate scheme for all the particular verbal form cases should be designed.

The segmentation of phrase patterns used for the input language parse was carried out with the consideration of semantics to be reproduced via the target language means. Both the most important universals such as enumeration, comparison, modality patterns, etc., and less general structures

were singled out and assigned corresponding target language equivalents.

Consider an example of a phrase structure conveying the modal meaning of obligation: "…the task to be carried out…". In other words, the meaning of this phrase can be rendered as "…the task that should be carried out…". The Infinitive phrase in the Engish language gives the regular way of expressive means compression without the loss of semantic value. A literary translation in Russian requires the second way of presenting the same idea of obligation. However in this specific case a "reduced" translation variant is also possible which consists in the introduction of the subordinate conjunction "chtoby" – "so that", between the noun and the modifying Infinitive. The parse rule would look like:

NP(to) -> NP VPto

And the generation rule would be presented as:

NP(to) -> NP Punct.{comma} Conj.(chtoby) VPto

Special attention is required for the problem of passive constructions transfer. As in the phrase "was considered". The rules for simultaneous translation (which in many cases is similar to the real time machine translation performance and can be a source of compromise decisions for phrase structure design) requires the transformation of the English Subject into the Direct Object (Russian, Accusative Case) standing in the first position in a sentence and the passive verbal form would produce an impersonal verbal form in Russian. However such transformation proved to be of considerable danger to the whole sentence structure and might cause an unpredictable generation result. Hence, for many cases a more clumsy, though robust method of a passive construction generation was accepted: the one similar to the English "be + Past Participle":

V(aux_ppt) -> V(aux) PPt

For any MT design scheme there exist major concerns such as verb subcategorization presentations, discontinuous structures treatment, phrasal units adjustment. In the English-Russian transfer these concerns are aggravated by the high productivity of the English phrasal verbs (and other units) and their derivatives.

a) An example of a phrase structure rule for the verb subcategorization:

V/np_p_inf --> Vinf NP Pt Vto_inf

get the sample down to observe

b) An example of a discontinuous structure:

Or watch the things, you gave your life to, broken

c) Phrasal units:

later on; over there; what a {good idea}.

Our approach employs both phrase structure rules and vocabulary-driven methods for dealing with these problems.

4   Generalized Cognitive Structures Underlying Transferable Syntaxemes

Actually the process of transfer goes across the functional – categorial values of language units. A language structure which can be subjected to transfer has to be semantically complete from the point of view of its function. The cases of categorial shifts, in particular, when the technique of conversion is employed, require special treatment: the categorial shift of a syntax unit is determined by the functional role of this unit in a sentence (e.g. noun as a modifier --> adjective). Only by creating the centaur concepts.. 'constituency-dependency', 'linearity-nonlinearity', 'form-function', etc. can we get a reasonably clear picture of linguistic reality [6].

The starting idea for the language structures segmentation strategy was the notion of functional semantic fields. The system of grammar units, classes and categories with generalized content supplementary to the content of lexical units, together with the rules of their functioning, is a system which in the end serves for transmission of generalized categories and structures of mental content which lie the foundation of utterance sense, and constitute the basis of language grammar formation [7].

As it was exhibited in [8] language coding technique is to a great extent determined by the deep semantic structure, and of considerable advantage is such a presentation method which takes for the starting point the semantic level, and particular semantic units are confronted with the coding devices expressing them. The approach of functional semantics concords in many aspects with the categorial grammar. The system of sentence members (functional roles) is being modified, but its essence is preserved in the new facts qualification via the traditional categories [9].

The transferability of phrase structures is conditioned by the choice of language units in the source and target languages belonging to the same functional transfer fields (FTF), notwithstanding the difference or coincidence of their traditional categorial values. A set of basic FTF was singled out and language patterns employed for conveying the functional meanings of interest were examined.

• Nomination and Relativity FTF: language structures performing the nominative functions (including the sentential units) comprise this field.

• Primary Predication FTF (non-inverted) bearing the Tense – Aspect – Voice features; this field mainly includes all possible complexes of finite verbal forms and tensed verbal phrase structures.

• Secondary Predication FTF bearing the features of verbal modifiers for the Primary Predication FTF. Included here are the non-finite verbal forms and constructions, and subordinate clauses comprising the finite verbal forms. All these are united by the functional meanings they convey, e.g. qualification, circumstance, taxis (ordering of actions), etc.

• Modality and Mood FTF: language means expressing modality, subjunctivity and conditionality are included here. Here the transfer goes across the regular grammatical forms and lexical means (modal verbs and word combinations) including phrasal units.

• Connectivity FTF: included here are lexical – syntactic means employed for concatenation of similar syntactic groups and subordination of syntactic structures.

• Attributiveness FTF: adjectives and adjectival phrases in all possible forms and degrees comprise the semantic backbone of this field; included here are also other nominal modifiers, such as nominative language units and structures (stone wall constructions, prepositional genitives – of – phrases), and other dispersed language means which are isofunctional to the backbone units.

• Metrics and Parameters FTF: this field comprises language means for presenting entities in terms of parameters and values, measures, numerical information.

• Partition FTF: included in this field are language units and phrase structures conveying partition and quantification (e.g. some of, part of, each of, etc.).

• Orientation FTF: this field comprises language means for rendering the meaning of space orientation (both static, and dynamic).

• Determination FTF: a very specific field which comprises the units and structures that perform the function of determiner (e.g. the Article, which is a good example for grammar – lexical transfer from English into Russian, since in Russian there exist no such grammatical category; demonstrative pronouns, etc.).

• Existentiality FTF: language means based on be-group constructions and synonymous structures (e.g. sentential units with existential there and it as a subject: there is…; there exists…; etc.).

• Negation FTF: lexical – syntactic structures conveying negation (e.g. nowhere to be seen, etc.).

• Reflexivity FTF: this field is of specific character since the transfer of reflexivity meaning goes across lexical - syntactic – morphological levels.

• Emphasis – Interrogation FTF: language means comprising this field are grouped together since they employ grammar inversion in English.

• Dispersion FTF: individual language structures specific for a given language are included here; these are presented as phrasal templates which include constant and variable elements.

The set of functional meanings together with their categorial embodiments serve the source of constraints for the unification mechanism in the formal presentation of our grammar. The formalism developed employs feature-based parse, and head-feature inheritance for phrase structures which are singled out on the basis of functional identity in the source and target languages. To implement the feature-valued inheritance sometimes broader contexts are taken.

5 The Existing Formalisms Influence

Our implementation formalism was developed taking into account the apparata of phrase structure and unification grammars: Head-Driven Phrase Structure Grammars (HPSG) [10], Generalized Phrase Structure Grammars (GPSG) [11], Revised Generalized Phrase Structure Grammars (RGPSG) [12]. Categorial and Dependency [13] grammars were also considered. Important for us was the strict lexicalism principle of the HPSG, i.e. word structure and phrase structure are governed by independent principles. Roles are determined by verbal valences; utterances are a blend of categorial meanings and role meanings and their structural projections which are specific for every particular language.

The technique of categorization, i.e. generation of a backbone grammar of atomic categories from distinct sets of feature bindings was first suggested by [14]. It was used for building a shift-reduce table for the Alvey grammar [5], but there it was necessary to subsume each category into its most general unifying category, so reducing the overall number of categories.

Generally, for the Russian language, dependency grammars have been applied. And phrase structure approach seemed to be less applicable here. Hence, of particular interest for us was the study and comparison of both the formal approaches, so that practical algorithmic solutions could be worked out.

A certain key was suggested in the coexisting systems of Immediate Dominance (ID) rules and phrase structure (PS) rules in ANLT [5] based on a variant of GPSG. The ID rules encode unordered dependency relations and further are subjected to linearization to be applied for the parse. GPSG may be thought of as a grammar for generating a context-free grammar. The generation process begins with immediate dominance (ID) rules which are context-free productions with unordered right-hand sides. An important feature of ID rules is that nonterminals in the rules are not atomic symbols (e.g. NP). Rather, GPSG nonterminals are sets of [feature, feature-value] pairs. For example, [N +] is a [feature, feature-value] pair, and the set {[N +], [V -], [BAR 2]} is the GPSG representation of a noun phrase. Next, metarules apply to the ID rules, resulting in an enlarged set of ID rules. In the RGPSG the finite closure problem is used to determine the cost of metarule application. Principles of universal feature instantiation (UFI) apply to the resulting enlarged set of ID rules, defining a set of phrase structure trees of depth one (local trees). One principle of UFI is the head feature convention which ensures that phrases are projected from lexical heads. Finally, linear precedence statements are applied to the instantiated local trees. The final result is a set of ordered local trees, and these are equivalent to the context-free productions in a context-free grammar. The process of assigning structural descriptions to utterances consists of two steps in GPSG: the projection of ID rules to local trees and the derivation of utterances from nonterminals, using the local trees.

In GPSG there are three category-valued features : SLASH which marks the path between a gap and its filler with the category of the filler; AGR which marks the path between an argument and the functor that syntactically agrees with it (between the subject and matrix verb, for example); and WH which marks the path between a wh-word and the minimal clause that contains it with the morphological type of the wh-word. In RGPSG the revision is unit feature closure: to limit category-valued features to containing only 0-level categories., i.e. 0-level categories do not contain any category-valued features. GPSG's ID/LP format models the head parameter and some free word order facts. The HPSG formalism is based on phrase structure rules, but dominance relations are implemented via head elements. Phrasal types are also treated in terms of multiple inheritance hierarchies that allow generalizations about diverse construction types to be factored into various cross-cutting dimensions.

In fact, each non-linear dependency rule is an encoded potential for actualization of a set of possible linear phrase structures. Therefore, we assumed a more computationally practical approach (to our knowledge, never used before in a bilingual situation), that of feature-valued head-driven phrase structures for both English and Russian.

6   Implementation Techniques

In conclusion,   it should be noted that this article describes the experience in creation of modern multilingual machine translation systems – the systems of phraseological translation. The extensive application of means of automation allowed to essentially reduce expenditures of human labour in the process of creation of this system, and therefore, to reduce the creation cost of such systems. The primary purpose in introducing feature structures and unification has been to provide a way to express syntactic constraints that would be difficult to express using the mechanisms of context-free grammars alone. The next step was to design a way to integrate feature structures and unification operations into the specification of a grammar.

This was performed by augmenting the rules of the hybrid grammar comprising context-free and context-dependent rules with attachments that specify feature structures for the constituents of the rules, along with appropriate unification operations that express constraints on those constituents. These attachments were used to associate complex feature structures with lexical items and instances of grammatical categories; to lead the composition of feature structures to larger grammatical constituents based on the feature structures of their component parts; to lay compatibility constraints between specific parts of grammatical constructions. Functional meanings of units were encoded in functional tags for phrase structures, and the feature-value types were determined by functional – categorial semantics, for example:

[Feature,EnumVerb]; [Category,bePlus]; [Category,toPlusInfinitive]; [Feature,verbModal] [Feature,verbComplex];], etc.

Such major problems as reference resolution and long distance dependencies are also treated within the framework of feature-valued phrase structures.

The demand for practicality, quick implementation and low computational cost were of prime concern.

The principle of effort economy was observed: if something could be represented by weaker means, no stronger instruments were applied. We acquired the "blow-up" strategy for language structures simulation, which means that the most functionally relevant subsystems were introduced first, and then these were expanded, specifying structures being gradually included.

A constraint-based formalism comprising some features of the HPSG was developed. The formalism provides representation mechanisms for the fine-grained information about number and person, agreement, subcategorization, as well as semantics for syntactic representations. The system of rules based on this formalism can be called the Cognitive Transfer Grammar and consists of transferable phrase structures together with the transfer rules which are combined within the same pattern. Such patterns, or Cognitive Transfer Structures (CTS), are constitutional components of the declarative [15] syntactical processor module and encode both linear precedence and dependency relations within phrase structures. The CTS presentation was worked out under a certain influence of the content-based attribute structuring approach assumed in dataflow basic components [16].

The syntax of a CTS can be given as follows:

CTS -> CTS<identifier> CTS<token> <Input Phrase Structure & Feature-Value Set> <Head-Driven Transfer Scheme> <Generation Feature-Value Set & Phrase Structure >

The Cognitive Transfer Grammar provides translation of phrase structures within one CTS,

e.g. him to come -> chtoby on prishel.

A CTS rule is either a context-free or context-dependent production, and the derivational process may alternate between an AND-transition and OR-transition, these two devices introduce lexical and structural ambiguity, which is a central property of natural languages. Disambiguation techniques are based on learning methods [17].

"Abstract" structures are avoided wherever possible, in favor of constituent structures. Linguistic information is hierarchically organized in such a way as to predict the impossibility of certain kinds of linguistic phenomena. The head features inheritance is widely used. Needed feature structures are copied from children to their parents, which turns out to be a specific instance of a much more general phenomenon in constraint-based grammars. Specifically, the features for most grammatical categories are copied from one of the children to the parent. The child that provides the features is called the head of the phrase, and the features copied are referred to as head features.

In our approach the direct encoding of possible subcategorization features is made via a verbal CTS. Since the verbs can subcategorize for quite complex frames composed of many different phrasal types, we first established a list of possible phrasal types that can make up these frames, e.g. VPto "I want to know"; VPing "He contemplates using them"; Sto "feel themselves to be relatively happy". Each verb allows many different subcategorization frames.

If compared with the existing phrasal subcategorization frames [18,19], in our system the emphasis is laid on functional motivation

7   Rule set for training data: cognitive semantic approach

In conclusion,   it should be noted that this article describes the experience in creation of modern multilingual machine translation systems – the systems of phraseological translation. The extensive application of means of automation allowed to essentially reduce expenditures of human labour in the process of creation of this system, and therefore, to reduce the creation cost of such systems. In contrast to the approaches on the basis of "translation memory" that provide the increase of a machine translation system language competence by accumulating the previously translated text fragments and mainly based on regular expressions, Cognitive Transfer Grammar is intended for the realization of the mechanism of structural memory, which simulates language competence of an adult learner ("Adult Learning Memory"). Thus, structural memory comprises the following components:

1) The initial basic collection of grammar rules represented in the formalized form (CTG);

2) The mechanisms of expansion and refinement of the system of rules, implemented by means of the methods of machine learning on parallel texts.

Our studies are based on the concepts of the functional approach, which we have used for the multilingual situation. With the development of the linguistic processor, which ensures English - Russian and Russian - English transfer, we introduced the concept of functional transfer fields (FTF) that served the basis for the segmentation of language structures for the solution of machine translation problems. The basic idea of FTF consists in the adoption of the hypothesis about the fact that at the basis of grammatical structures there lie the cognitive structures (mental frames); a functional transfer field reflects the interaction of elements from different language levels.

The basic design unit of the spaces of cognitive transfer is a transfeme.

Definition. Transfeme is a unit of cognitive transfer the, i.e. a semantic element embodied in a translatable semantically relevant language segment taken in the unity of its categorial and functional characteristics, that establishes the semantic correspondence between the language structures, which belong to different language levels and systems. The types of transfemes are determined by the rank of transfemes.

We distinguish the following ranks of transfemes:
• rank 1: lexemes as structural signs, i.e., a word, considered as a categorical - functional unit without taking into account the specific lexical value of this word;
• rank 2: a word combination, i.e., the syntactic structure, which consists of two and more syntactically connected words, but never a complete sentence (clause);
• rank 3: a clausal unit, i.e., dependent (subordinate) clause;
• rank 4: a sentence (either a simple sentence or the main clause of a complex sentence);
• rank 5: a scattered structure, i.e., a word group, which is characterized by a syntactic and semantic unity, but is discontinuous, i.e., between the members of the group there appear other language objects, which are not the members of this group;
• rank 0: the morphological units, which are not independent words, but which form a part of a lexeme of a source language, and in the language of transfer can be expressed by a clause and the units of other ranks, for example: the suffixes – ible, - able which are synonymous to the construction "which can be", e.g. extensible – which can be extended.

he key idea of our linguistic framework is cognitive cross-linguistic study of what can be called configurational semantics, i.e. the systemic study of the language mechanisms of patterns production, and what meanings are conveyed by the established types of configurations. We explore the sets of meanings fixed in grammar systems of the languages under study. Our studies are focused on the types of meanings outside the scope of lexical semantics, and we consider the lexical semantics when the meanings which we denote as configurational, have expression at the lexical level. The importance of this aspect is connected with the fact that natural languages are selective as to the specific structures they employ to represent the referential situation. However, it is always possible to establish configurations which perform the same function across different languages (i.e. isofunctional structures). The parse aimed at transfer procedures requires a

semantic grammar and cannot be efficiently implemented through a combination of monolingual grammars.

In the Cognitive Transfer Grammar (CTG), the functional meanings of language structures are determined by the categorial values of head elements. The probability characteristics are introduced into the rules of the unification grammar as weights assigned to the parse trees.

For the alignment of parallel texts the transfemes are given as the rewrite rules in which the left part is a nonterminal symbol, and the right part are the aligned pairs of chains of terminal and nonterminal symbols which belong to the source and target languages :

$$T \rightarrow \langle \rho, \alpha, \sim \rangle,$$

where T is a nonterminal symbol, ρ and α are chains on terminal and nonterminal symbols which belong to the Russian and English languages, and ~ is a symbol of correspondence between the nonterminal symbols occuring in ρ and the nonterminal symbols occuring in α. In the course of parallel texts alignment on the basis of the CTG the derivation process begins with a pair of the linked starting symbols and , then at each step the linked nonterminal symbols are rewritten pairwise with the use of the two components of a single rule.

For automatic extraction of the rules on the basis of CTG from parallel texts these texts should be previously aligned by sentences and words. The extracted rules base on the wordwise alignments in such a way that at first the the starting phrase pairs are identified with the use of the same criterion as the majority of statistical models of translation employing the phrase-based approach, which means that there should be at least one word inside a phrase in one language aligned with some word inside a phrase in another language, but no word inside a phrase in one language can be aligned with any word outside its pair phrase in another language.

8   Acknowledgements

9   Conclusion

The urgency of the new hybrid methods of language objects presentation is caused by the demand for the optimal combination of advantages of the two research paradigms: logical linguistic modelling employing the designed rules and stochastic approach based on machine learning [19-21]. This development is of special importance for the tasks of structural analysis and computer modelling of the full text scientific and patent documents. The work with patent documents requires the introduction of specific features of patent texts: such as employment of certain language constructions, the syntax of patent formulae, the extensive use of templates, domain-oriented lexicons. The Intertext base comprises a collection of scientific and patent texts in the Russian and English languages from the areas of Computer Science, Social Monitoring, Chemical Technology and other areas. One of the latest developments is connected with implementing the natural language web service for the multilingual search and analysis of financial information. The objectives of the prospective research and development efforts consist in the inclusion of parallel texts and language

processing features for the French, German and Italian languages, and evolving the Intertext into a multilingual knowledge base. Our focus on configurations provides high portability to the language processing software designed under these principles: we can operate with a lexicon which has only standard linguistic information including morphological characteristics, part of speech information and the indication of transitivity for verbs. The approach taken would be important in further development of educational programs for computer science and computational linguistics courses. Educational relevance of the methods discussed in the paper lies in deeper understanding of uniform cognitive mechanisms employed in particular language embodiments of semantic structures.

10   References

[1]   Comrie, B. Language Universals and Linguistic Typology. Basil Blackwell, Oxford. Second edition. 1989.

[2]   Nirenburg, S., Carbonell, J., Tomita, M., and Goodman, K. Machine Translation: A Knowledge-based Approach. Morgan Kaufmann. 1992

[3]   Visson, L. From Russian Into English: An Introduction to Simultaneous Interpretation. Ann Arbor, Michigan: Ardis, 1991.

[4]   Visson, L. Syntactical Problems for the Russian-English Interpreter. No Uncertain Terms, FBIS, vol. 4, N 2, 1989, 2-8.

[5]   Grover, C., Carroll, J. and Briscoe, T. The Alvey Natural Language Tools Grammar (4-th Release). Technical Report, 1993, Computer Laboratory, University of Cambridge, 1993.

[6]   Shaumyan, S. A Semiotic Theory of Language. Indiana University Press, 1987. .

[7]   Bondarko A.V. Printsipy Funktsional'noi Grammatiki I Voprosy Aspektologhii. Moskwa, URSS, 2001 / Functional Grammar Principles and Aspectology Questions. Moscow, URSS, 2001 (In Russian).

[8]   Kibrik A.E. Ocherki po Obstchim I Prikladnym Voprosam Yazykoznaniya. Moskwa, URSS, 2002. / Studies in General and Applied Linguistics Issues. Second Edition. Moscow, URSS, 2001 (In Russian).

[9]   Zolotova G.A. Kommunikativnye Aspekty Russkogo Sintaksisa. Moskwa, URSS, 2001/ Communicative Principles of the Russian Syntax. Moscow, URSS, 2001 (In Russian).

[10] Pollard, C. and Sag, I.A. Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago, 1994.

[11] Gazdar, G., Klein, E., Pullum, G. and Sag,I. Generalized Phrase Structure Grammar. Oxford: Basil Blackwell, 1985.

[12] Ristard, E.S. Computational complexity of current GPSG theory. Proceedings of the 24-th Annual Meeting of the Association for Computational Linguistics. Columbia University, New York: Asociation for Computational Linguistics. 1986, pp. 30-39.

[13] Mel'cuk, I.A. Dependency Syntax: Theory and Practice, State University of New York Press, 1988.

[14] Gazdar, G. and Mellish, C. Natural Language Processing in Prolog. Wokingam, UK: Addison-Wesley, 1989.

[15] Kozerenko E.B. Portable Language Egineering Solutions for Multilingual Processors // Proceedings of the International Conference on Artificial Intelligence IC-AI'02// CSREA Press, 2002, pp. 447-453

[16] Arlazarov V.L., Emelyanov N.E. Document Processing Systems. Basic Components. /Data Flow Management. Ed. Prof. Arlazarov V.L., Prof. Emelyanov N.E. – Moscow: Editorial URSS, 2002. (in Russian).

[17] Missiourev A. Hand-printed Character Recognition by Neural Networks.// Proceedings of the 5-th German-Russian Workshop on Pattern Recognition and Image Understanding (GRWS98), 1999.

[18] Baker, C.F., Fillmore, C.J., and Lowe, J.B. The Berkeley FrameNet project. In COLING/ACL-98, pp. 86-90, 1998.

[19] B*rown P.F., Cocke J., Della Pietra S.A., Della Pietra V.J., Jelinek F., Lafferty J.D., Mercer R.L., Roossin P.S.* A statistical approach to machine translation // Computational Linguistics, 1990. Vol. 16. P. 79–85.

[20] *Och F.J., Ney H.* A Systematic Comparison of Various Statistical Alignment Models // Computational Linguistics, 2003. Vol. 29. No. 1. P. 19–51.

[21] *Koehn P. and Hoang H.* Factored translation models // Proceedings of the 2007Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). 2007. P. 868–876.

New Ensemble Combination Scheme

Namhyoung Kim, Youngdoo Son, and Jaewook Lee, *Member, IEEE*

Abstract— Recently many statistical learning techniques are successfully developed and used in several areas. However, these algorithms sometimes are not robust and does not show good performances. The ensemble method can solve these problems. It is known that the ensemble learning sometimes improves the generalized performance of machine learning tasks as well as makes it robust. However, the combining weights of the ensemble model are usually pre-determined or determined with the concept that the ensemble model is a superposition of individual ones. Thus we proposed a new ensemble combination scheme which consider the ensemble model is a factor affects the individual predictors. Through experiments, the proposed method shows better performance than other existing methods in the regression problems and shows competitive performance in the classification problems.

I. INTRODUCTION

Several learning models such as clustering, classification, regression, etc., are successfully developed in these days and they are used in many areas [1], [2], [3], [4]. However, sometimes these algorithms show ill performances or are not robust. Ensemble can be a solution to overcome these problems. Ensemble combines multiple models together in specific ways. The ensemble learning is known to improve the generalized performance of a single predictor and it has won some success. For example, the famous one million dollars Netflix Prize competition, the team named *The Ensemble* won the first place. They were known to ensemble several algorithms. In other data mining competitions, most winning teams employed an ensemble of learning models. Also, the ensemble method has been widely used in several practical area [5], [6], [7]

Along with these practical successes, many different ensemble algorithms have been proposed and developed [3], [8], [9]. However, some of them use the pre-determined weights. Most of the rest find weights with the concept that the ensemble model is a superposition of individual predictors. Unlike existing ensemble models, we propose a new ensemble combination scheme which consider individual predictors as realizations of the ensemble model. In other words, in the view of factor analysis, the ensemble model becomes a factor which affects individual models.

The organization of this paper is as follows. In the next section we summarize related literature and algorithms. In Section 3 we describe the proposed method. Next we present the results of an empirical analysis and comparison with other ensemble methods in Section 4. Then we conclude in Section 5.

Namhyoung Kim is with the Department of Statistics, Seoul National University, Seoul, Korea (email: namhyoung@snu.ac.kr).

Youngdoo Son and Jaewook Lee is with the Department of Industrial Engineering, Seoul National University, Seoul, Korea (email: hand02, jaewook@snu.ac.kr).

II. LITERATURE REVIEW

Most of the contemporary ensemble methods develop ensemble model based on the following equation.

$$f = \sum_{i=1}^{m} w_i C_i \qquad (1)$$

where f is a real function predicted, C_i is an individual base model and w_i is weight. It is illustrated in Figure 1, that is, the predictions of the individual models $C_i, i = 1, 2, ..., m$, are combined using the weight w_i. The weights can vary depending on combination scheme. According to combination scheme, ensembles are divided into two main classes, combine by learning and combine by consensus. Boosting combines multiple base predictors by learning. It utilizes the information of the performance of previous predictors so this kind of ensemble is also known as supervised scheme. In bagging, the ensemble is formed by majority voting. Bagging doesn't consider the performance of each predictor. It averages the predictions of individual models in unsupervised scheme.

A. Bagging

Bagging is the simplest algorithm to construct an ensemble[10]. It creates individual predictors by training randomly selected training set. Each training set is generated by randomly with replacemeth, n exampels. In bagging, the weights are uniform, i.e., the ensemble prediction is given by

$$f = \frac{1}{m} \sum_{i=1}^{m} C_i. \qquad (2)$$

In the case of classification, majority voting is used to predict the class of data.

B. Boosting

Boosting algorithms learn iteratively weak classifiers using a training set selected according to the previous classified results then combine them with different weights to create an ensemble[12]. The weights are determined by the weak learners' performance.

Our method uses both supervised and unsupervised scheme. In the following section, we describe our proposed method.

III. PROPOSED METHOD

In this paper, we propose a new ensemble combination scheme. We assume that the prediction of each individual predictor C_i reflects the real function value f by λ_i and a specific error term δ_i. In equation form, we would write the following.
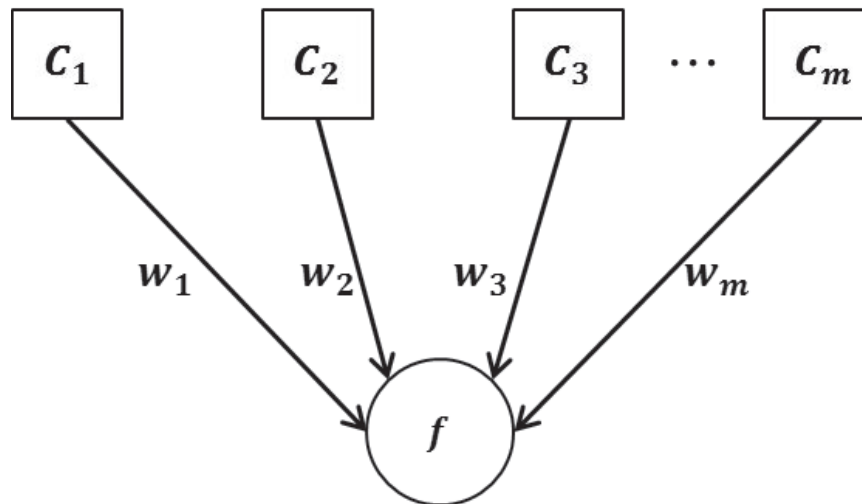
Fig. 1.   Ensemble of Classifiers

$$C_i = \lambda_i f + \delta_i, \quad i = 1, 2, ..., m \tag{3}$$

We can rewrite the above equation in matrix form as following.

$$C = f\Lambda' + \Delta \tag{4}$$

$$\text{where} \quad C = \begin{bmatrix} C_1^{(1)} & C_2^{(1)} & \cdots & C_1^{(1)} \\ C_1^{(2)} & C_2^{(2)} & \cdots & C_m^{(2)} \\ \vdots & & & \\ C_1^{(n)} & C_2^{(n)} & \cdots & C_m^{(n)} \end{bmatrix}, f =$$

$$\begin{bmatrix} f^{(1)} \\ f^{(2)} \\ \vdots \\ f^{(n)} \end{bmatrix}, \Lambda = [\lambda_1 \lambda_2 \cdots \lambda_m]', \quad \text{and} \quad \Delta =$$

$$\begin{bmatrix} \delta_1^{(1)} & \delta_2^{(1)} & \cdots & \delta_1^{(1)} \\ \delta_1^{(2)} & \delta_2^{(2)} & \cdots & \delta_m^{(2)} \\ \vdots & & & \\ \delta_1^{(n)} & \delta_2^{(n)} & \cdots & \delta_m^{(n)} \end{bmatrix}.$$ n is the number of cases

of data set. In addition, we have the following three assumptions about the proposed method.

Assumption 1:   1) Variance of f is 1.

$$(n-1)f'f = 1 \tag{5}$$

2) Specific factors δ are mutually uncorrelated with diagonal covariance matrix:

$$\Theta = \frac{1}{n-1}\Delta'\Delta = diag(\theta_1^2, \theta_2^2, \ldots, \theta_m^2). \tag{6}$$

3) Common factor f and specific factors δ are uncorrelated.

approximation of the correlation matrix S is

$$\begin{aligned} S &= 1/(n-1)C'C \\ &= 1/(n-1)(f\Lambda' + \Delta)'(f\Lambda' + \Delta) \\ &= 1/(n-1)(\Lambda f'f\Lambda' + \Delta'f\Lambda' + \Delta f'\Delta + \Delta'\Delta) \end{aligned}$$

By assumptions, the correlation matrix becomes simplified as following

$$S = \Lambda\Lambda' + \Theta \tag{7}$$

ML solution : $S(\Lambda'\Lambda + \Theta)^{-1}\Lambda = \Lambda$
$f_{(i)} = \Lambda'(\Lambda\Lambda' + \Theta)^{-1}C_{(i)}$ or $f_{(i)} = (1 + \Lambda'\Theta^{-1}\Lambda)^{-1}\Lambda'\Theta^{-1}C_{(i)}$.

Let $f_{(i)}^* = \Lambda'(\Lambda\Lambda' + \Theta)^{-1}C_{(i)} + \epsilon_{(i)}$ and $C_{(i)} = \Lambda f_{(i)}^*$
Then $\Lambda f_{(i)}^* = C_{(i)} = \Lambda\Lambda'(\Lambda\Lambda' + \Theta)^{-1}C_{(i)} + \Lambda\epsilon_{(i)}$
$\Lambda'\Lambda\epsilon_{(i)} = \Lambda'[I_p - \Lambda\Lambda'(\Lambda\Lambda' + \Theta)^{-1}]C_{(i)}$
$\therefore \epsilon_{(i)}^* = \Lambda'[\frac{1}{\Lambda'\Lambda}I_p - (\Lambda\Lambda' + \Theta)^{-1}]C_{(i)})$

Bayesian Latent Ensemble
$f_{(i)}^* = \frac{\Lambda'\Theta^{-1}}{1 + \Lambda'\Theta^{-1}\Lambda}C_{(i)} + \epsilon_{(i)}^*$
$f^* = \frac{\sum_{k=1}^m \lambda_k \theta_k^{-2}C_k}{1 + \sum_{l=1}^m \lambda_l^2 \theta_l^{-2}} + \epsilon$
$f^* = \sum_{k=1}^m \frac{\lambda_k \theta_k^{-2}}{(1 + \sum_{l=1}^m \lambda_l^2 \theta_l^{-2})}C_k + \epsilon$

In supervised scheme, we assume the given output of training sets as real function value. Then, the process becomes much simpler.

IV. EXPERIMENTAL RESULTS

A. Data Sets

To evaluate the performance of the proposed method, we use both regression and classification data sets from UCI Machine Learning Repository[11]. Table I shows the summary of the data sets used. The data shown in Table I is widely used real-world problems in the machine learning
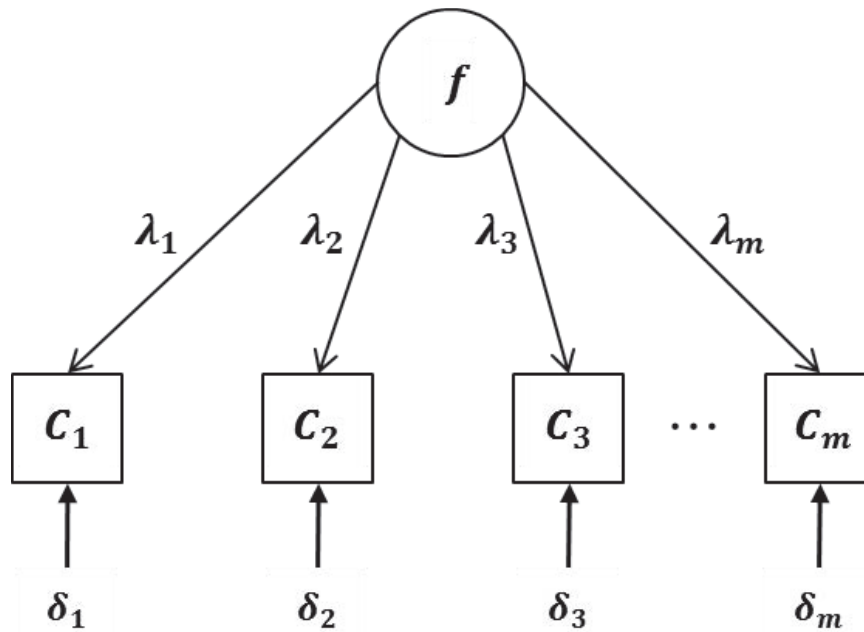
Fig. 2. Proposed method

TABLE I

SUMMARY OF THE DATA SETS

Data set	cases	class	Features		Neural Network			
			cont	disc	inputs	ouputs	hiddens	Epochs
breast-cancer-w	699	2	9	-	9	1	5	20
Credit-a	690	2	6	9	47	1	10	35
ionosphere	351	2	34	9	34	1	10	40
Statlog_german	1000	2	7	13	45	1	10	20
Statlog_aust	690	2	6	8	40	1	10	35
ailerons	13750	regression	40	-	40	1	10	20
Housing_boston	506	regression	13	-	13	1	5	35
Pole telecom	5000	regression	48	-	48	1	10	20
abalone	4177	regression	7	1	9	1	5	20

community. The data sets vary in terms of both the number of cases and features.

B. Experiments settings

In this study, neural networks are used as a base model. Neural networks consist of a series of input, hidden and output layers. It is needed to set the size of hidden layer. Some parameters are also required for the neural networks. We trained the neural networks with a backpropagation function, a learning rate of 0.15, a momentum constant of 0.9 and randomly selected weights between -0.5 and 0.5. The size of hidden layer was determined according to the number of input and output units having five hidden units as a minimum. The number of epochs was based on the number of cases. We used a larger value of epochs as the size of data increases. The training parameters used in the experiments are shown in Table I.

For comparison, other popular ensemble algorithms, bag-ging and boosting, and single model are applied to the same data sets. We averaged test error of each method over five standard 10-fold cross validation experiments. The data set is divided into 10 equal-sized sets, then each set is used as the test set in turns then the other nine sets are used for training. In all ensemble methods we used an ensemble size of 25 for each fold.

C. Results

Table II and Table III show the classification error for classification problems and root mean squared error for regression problems, respectively. The columns indicate the different methods, a single model(Single), Bagging(Bagging), Arcing(Arc), Ada-boosting(Ada) and the proposed method(FA). Each row shows the results for each data set. From the tables we can notice that the proposed method usually shows good performances for several data sets.

In classification problems(Table II), the proposed method

does not shows the worst performance and its results are often the best or close to the best. In regression problems(Table III), the proposed method shows the best performances except for the housing_boston data set. In that data set, bagging algorithm performs the best but the proposed method showed competitive result with the bagging. Therefore, we can say that our proposed method performs better than other existing ensemble methods in general.

V. Conclusion

In this study, we presented novel methods for ensemble combination scheme with a new perspective on a relationship between real function and each classifier. We assumed predicted value of each classifier is composed of the real function value and a specific error term. According to this assumption, we solved weights for ensemble combination in both supervised and unsupervised scheme. The experimental results show the proposed methods are superior to the existing methods. They also show self-correction skills, so an ensemble can adjust weights noticing corrupt classifiers in learning process.

The proposed ensemble method seems to be robust with some corrupting predictors unlike other ensemble schemes whose performances may decrease with corrupting predictors. In other words, the proposed method has self-correction reliability. Thus, for future work, the analogous proof and experimental results for this intuition are necessary. We also need to prove and show the statistical stability of the proposed model using statistical tests like Friedman test, t-test, or sign-test.

References

[1] D. Lee and J. Lee, "Domain descried support vector classifier for multi-classification problems," *Pattern Recognition*, vol. 40, pp. 41–51, 2007.

[2] Y. Son, D-j. Noh, and J. Lee, "Forecasting trends of high-frequency KOSPI200 index data using learning classifiers," *Expert Systems with Applications*, vol. 39, no. 14, pp. 11607–11615, 2012.

[3] N. Kim, K-H. Jung, Y. S. Kim, and J. Lee, "Uniformly subsampled ensemble (USE) for churn management: Theory and implementation," *Expert Systems with Applications*, vol. 39, no. 15, pp. 11839–11845, 2012.

[4] H. Park and J. Lee, "Forecasting nonnegative option price distributions using Bayesian kernel methods," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13243–13252, 2012.

[5] T. Gneiting and A. E. Raftery, "Weather forecasting with ensemble methods," *Science*, vol. 310, no. 5746, pp. 248–249, 2005.

[6] D. Morrison, R. Wang, and L. C. De Silva, "Ensemble methods for spoken emotion recognition in call-centres," *Speech communication*, vol. 49, no. 2, pp. 98–112, 2007.

[7] L. K. Hansen, C. Liisberg, and P. Salamon, "Ensemble methods for handwritten digit recognition," *Neural Networks for Signal Processing [1992] II., Proceedings of the 1992 IEEE-SP Workshop*, IEEE, 1992.

[8] T. G. Dietterich, "Ensemble methods in machine learning," *Multiple classifier systems*, pp 1–15, Springer Berlin Heidelberg, 2000.

[9] H. Drucker, *et al.*, "Boosting and other ensemble methods," *Neural Computation*, vol. 6, no. 6, pp. 1289–1301, 1994.

[10] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[11] A. Frank and A. Asuncion, *UCI Machine Learning Repository* [http://archive.ics.uci.edu/ml], Irvine, CA: University of California, School of Information and Computer Science, 2010.

[12] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," *In Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 148–156, 1996.

TABLE II

CLASSIFICATION RESULTS (CLASSIFICATION ERROR)

*Data set	Neural Network					
	Single	Bagging	Boosting		FA	
			Arc	Ada	Unsupervised	Supervised
Breast-cancer-w	3.59%	3.15%	3.35%	2.91%	3.50%	3.56%
credit-a	14.17%	13.94%	15.04%	14.52%	14.03%	13.88%
ionosphere	12.91%	11.71%	10.17%	10.86%	8.46%	8.51%
statlog_aust	15.48%	13.83%	15.22%	14.75%	13.57%	14.26%
statlog_german	26.66%	24.68%	27.48%	24.96%	23.78%	23.96%
hepatitis	36.80%	34.13%	36.40%	36.00%	33.87%	34.00%
sonar	17.40%	16.60%	14.40%	15.70%	16.60%	16.70%
diabetes	24.16%	23.21%	24.82%	23.16%	24.13%	24.16%

TABLE III

REGRESSION RESULTS (RMSE)

Data set	Neural Network				
	Single	Bagging	Boosting	FA	
				Unsupervised	Supervised
Abalone	2.0989	2.0733	7.6227	2.0702	2.0711
housing_boston	3.8296	3.2316	17.3399	3.2743	3.2705
ailerons	0.0005	0.0003	0.0021	0.0002	0.0002
pole telecom	13.6437	10.5535	31.5003	10.3371	10.0061

Finding semantic correlation in structured data from unstructured notes

C. Anantaram, Shailja Khurana, Vuppala Gowri Dev
Tata Consultancy Services Limited, 249 D&E Udyog Vihar Phase IV, Gurgaon, Haryana, India

Abstract - *In financial reports, sometimes, unstructured notes explain the nuances of the financial data described in structured form. The problem of interest here is to parse such unstructured notes and find the semantic correlation that it implies in the structured data. While techniques like Named Entity Recognition have been used for extracting information from unstructured text, it has been observed that in order to discover meaningful semantic correlation in structured data, it is important to parse the unstructured notes in accordance with the terms and relations specified in the ontology of the domain. In this paper we examine this problem and present a mechanism to address the extraction of relations from unstructured notes and determine the semantic correlation of such relations with the structured data. Once such correlations are established, it becomes easier to understand the structured data, or have an automated question-answering system respond to user queries.*

Keywords: Unstructured notes, semantic correlation, information extraction, structured data

1   Introduction

Reports, such as financial reports, medical reports, event reports or military reports, are a mechanism to describe the status of an entity / entities in a domain, together with the relevant data about the entity [7, 8]. Typically, such reports consist of structured data in the form of tables or fields that report the values of the measurements/ observations of the entities of interest. In some reports, unstructured notes are sometimes added to aid the explanation of structured data in the report [9]. For example, in financial reports, many unstructured notes may appear in Company Annual Reports, Reports to Shareholders, or Corporate Balance Sheets, to help explain the modes and mechanisms of accounting practices that have been followed [8,9]. Primarily targeted for *human analyst* consumption, there is no formal structure of such unstructured notes. With increasing volumes of reports required to be processed by human analysts, the need for automated processing becomes very important. While structured data from such reports may be easily retrieved and presented, the automatic processing of unstructured notes has posed a great challenge, especially when it has to be used to determine the relations in structured data that are implied in the unstructured notes.

We examine the problem of parsing unstructured notes to determine the semantic correlation that they imply between the various elements in the structured data. Such parsing and semantic correlation would then help in automated question-answering on the report. We restrict our work to closed-domain reports, that is, where the domain is clearly specified by a limited formalized ontology. In our architecture, the unstructured notes are parsed to identify the relevant ontology terms that the note entails, and then the formal ontology is used to determine the possible set of relationships between the terms. Once the set of possible relationships are identified, the most likely relationship from this set is chosen from the values that are present in the notes. The identification of the relationship then helps us to semantically correlate the values in the structured data that is present in the report. This can then help a user understand the correlations among structured data or form the backbone of an automated question-answering system.

2   Related work

Some attempts have been made in parsing unstructured notes. The MUSING project [10] extracts data from financial statements by identifying the concepts from such statements, and populates the concepts in XBRL form. Further, it identifies the structure of the balance sheets using patterns developed in JAPE (regular expression engine), and maps each line of the balance sheet into the appropriate XBRL concept, which is then validated by an expert. While the focus has been to convert unstructured notes into structured format, the framework does not extract the relations between concepts that are implied in the notes. Our work addresses this aspect.

Mansuri and Sarawagi [1] proposed a system for automatically integrating unstructured text into multi-relational databases using state-of-the-art statistical models for structure extraction and matching. Firstly, cosine similarity is used to find entity at word level since entity in unstructured data may be represented in slightly different form in database. Next, a set of regular expressions are built from entities present in database which is used to build a pattern dictionary of entities. A set of features such as: typical length of entities, characteristics words and multiple patterns per tokens are given to Semi-CRF to identify entity in unstructured data. The focus in this work has been on entity extraction for huge

datasets. However, when parsing reports one may not have access to such huge datasets.

Several machine learning techniques have been used for entity extraction from un-structured data. Borthwick [2] examines how knowledge resources play an important role in identifying entities. External dictionaries are used to improve accuracy of NER tasks. They have used conditional models having Boolean features based on exact match of a word with terms that appear in an entity dictionary. Cohen et.al [3] have improved Borthwick approach to handle noisy dictionaries through features that capture various forms of similarity measures with dictionary entries. They have used various distance measures (such as *edit distance, soundex, n-grams overlap, jaccard, jaro-winkler* and *subsetmatch* at attribute level) as features.

3   Overview of correlation

Typically, an unstructured note may consist of one or more paragraphs written in free-format text, which provides a description of some of the key points that one should take into consideration while going through a report. We process each note to identify the concepts in the domain that the note refers to, determine if there are any valid values for these domain terms, and extract the relationship implied among the terms and/or values specified in the note. We assume that structured data is represented in the form of a table in the report. Such a table contains various domain terms and their measured values that form the basis of the report. In our method we strive to identify the relevant tables and the fields in those tables that are related to the concepts and relationships identified during the parsing of the unstructured note. Based on this identification our method then tries to construct the semantic correlation between the identified concepts and the structured data that is present in the report. As an example, consider an unstructured note in a Company Balance Sheet that states: `Intangible assets include customer contracts and relationships ($2.9 billion) with an estimated weighted average life of 14.7 years, favorable leaseholds ($12.7 million) with an estimated weighted average life of 6.2 years, covenants not to compete ($9.0 million) with an estimated average life of 2 years and trade names ($6.4 billion), which are indefinitely lived.`
This note indicates those account heads that are included in the computation of Intangible Assets in the Balance Sheet and reported. On parsing the note, our method identifies the concepts "Intangible assets", "customer contracts and relationships", "favorable leaseholds", "covenants not to compete", "trade names", and the relationships "estimated weighted average life", "indefinitely lived", "include", "estimated average life". The values such as "($2.9 billion)" etc are also identified.

Table1: Sample from Balance Sheet

Assets: *(In millions, except share and per share amounts)*	Sept 2008	Dec 2007
Cash and cash equivalents	$ 1,282.3	$1,056.6
Short-term investments	—	27.5
Accounts receivable, net	4,799.5	4,579.6
Inventories	8,345.4	8,008.2
Deferred income taxes	360.2	329.4
Other current assets	181.4	148.1
Total current assets	14,968.8	14,149.4
Property and equipment, net	6,489.5	5,852.8
Goodwill	23,925.0	23,922.3
Intangible assets, net	10,197.2	10,429.6
Other assets	371.0	367.8
Total assets	$ 55,951.5	$54,721.9

Once this is done, we correlate these concepts with the figures reported in the datasheet (Table 1). This also points out that Intangible assets include some more account-heads as it does not add up to USD 10.197 billion as given above. We can then look up other structured data in the report (Table 1) to identify the possible set of tables and fields that could form a part of this correlation. Thus in the example above, one is able to find the correlation between elements of the structured data based on the information provided in the unstructured notes. Our semantic correlation mechanism comes up with the equation:

```
Intangible assets = 10.197 billion = 2.9
(customer contracts and relationships)
+0.0127 (favorable leaseholds) +
0.009(covenants not to compete) + 6.4
(trade-names) + 0.8753 (account-head-to-
be-identified)
```

3.1   Semantic correlations

Relations that are identified between Concepts that are semantically valid in the domain are called *semantic correlations*. We define correlations of the following types:

a) *Equations*: This involves domain terms with an equality of parameters on both sides. The example illustrated in section 3 above of `Intangible assets` is an equality relation between predicate-value pairs.

b) *Rules*: This specifies an 'if-then-else' form of relation between the concepts. For example the note "`Intersegment eliminations relate to intersegment revenues and accounts receivables that occur when a Pharmacy Services Segment customer uses a Retail Pharmacy Segment store to purchase covered products. When this occurs, both segments record the revenue on a standalone basis`" connotes a rule between various concepts.

c) *Comparison*: This specifies the comparison of values over various time intervals. For example, in a Payroll a note such as "`employee benefits and occupancy costs increased $48.0 million to`"

$306.0 million, in the first quarter of 2009, compared to the first quarter of 2008" will connote a comparison type of correlation.

d) *Constraints*: This specifies a constraint for the values of the predicates that appear in the note. For example, "Inter-company revenues that occur when a Caremark customer uses a pharmacy retail store to purchase covered merchandise were eliminated" connotes a constraint for "Inter-company revenues".

4 Challenges

Building a natural language mechanism to process unstructured notes in order to find correlations in structured data is a challenge. We discuss some of the challenges below.

4.1 Identifying valid domain terms and relationships in the note

One of the challenges is to identify terms and relationships mentioned in the note that are *valid* in the domain. Terms may be multi-word expressions. Specifically,

a. Domain terms may be written in various permutations, sometimes using some of their synonyms. For example, the term "Fixed Assets" may be referred as "Assets (immovable)"; or "Long term investments" could be written in the note as "Investments, long term".

b. Missing words can make it difficult to find the valid domain term. For example, "Unutilized money investmented" actually refers to the Domain term "Balance of unutilized money investmented".

c. Identifying valid relationships from the sentence may also be a difficult task as terms need to be semantically interpreted. For example, "Gross profit includes net revenues less cost of revenues" implies the equation: Gross profit = net revenues - cost of revenues.

4.2 Identifying concepts

The second challenge is to identify semantically related domain terms and then identify the possible set of concepts that it connotes, rank them and select the most probable concept from the list. For example, if we process the text "customer contracts and relationships" we need to identify one concept in the domain consisting of two domain terms "customer contracts" and "customer relationships".

4.3 Identifying semantic chunks of the note

Chunking parts of the text is an important step in processing a note. However, standard chunkers may fail to produce a chunk that is semantically valid in the domain. For example, "Long term investments ($3456.7million) included long-term debt-securities traded mutual funds and current debt securities" standard chunkers may chunk "traded" and "mutual funds" as separate chunks, whereas "traded mutual funds" is one chunk.

4.4 Identifying semantic correlations between the concepts

Identifying the semantic correlations between the concepts is another challenge as the correlations can be implied and stated in various forms. Further, the domain ontology may not permit certain types of correlations, or only correlations of a particular type may be valid. Correlations may be implied in the form of equations, rules, or constraints and the mechanism will have to identify this classification and categorize them appropriately.

5 Architecture

To tackle the challenges discussed above, our architecture takes unstructured notes as input to find correlations among structured data. The main component is a CORE engine that does the overall processing and produces semantically correlated data in form of equations, rules, comparisons or constraints. The Engine follows a four step process for processing each note: pre-processing, identification of concepts, semantic rearrangement of chunks, and relationship extraction.

5.1 Pre-processing of unstructured data

Each note is pre-processed to identify the terms and relationships. We use sentence splitter to extract the sentences from the notes. Stanford core NLP tool is used to parse the sentences. The parsed information is used to identify the concepts.

5.2 Identification of concepts

General-purpose natural language parsers may sometimes be inadequate because even though they give syntactically a correct parse, they lose out on the semantics of the sentence. Because such parsers lack domain knowledge, we need to post-process the output of a general purpose NL parser in order to obtain semantically valid chunks for the input sentence. Once all concepts are identified, rearrangement of chunks is done using domain ontology. We find the domain concepts as follows:

1. *Synonym replacement-* Using a synonym dictionary we replace synonyms in the sentence to their counterpart domain terms in the input sentence.
2. *Direct match using Boyer Moore algorithm* -Using Boyer-Moore and KMP string search algorithms [6], we look for direct matching of words in input sentence to concepts in the ontology. We call this resultant set of concepts as C2.
3. *Partial/indirect match* – Since domain terms may be multi-word expressions and can be referred in any fashion, an indirect match mechanism with domain ontology is required. This is achieved using "partial-match" search mechanisms such as Apache SOLR, after removing direct matched concepts and noise words. The domain ontology is indexed for fast search. Using "N-gram" algorithm at word level we generate pair of words from the sentence .The n-gram matched word is identified as the concept. N is incremented until match by the search engine fails. This gives the maximally matched set of domain terms that forms the concept. We mark this result set of concepts as C3 and maintain a dictionary, composed of keywords and corresponding domain terms found by the search engine. Thus we have a set of partial-matching concepts, C3. This set is combined with the set C2 to create a set C4 consisting of direct-matched concepts as well as the indirect-matched concepts.

5.3   Relationship Extractor

Role of relationship extractor is to extract relationship among domain terms specified in the input sentence. Since relations could be of any type, e.g. equations, rules, comparisons or constraints (as explained in section 4), relations are extracted in many phases (Figure 1). These steps are explained below:
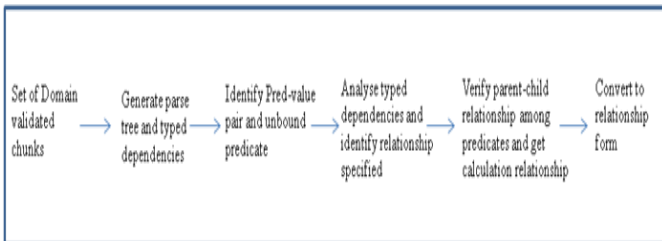
Figure 1 Relationship extractor

5.3.1   Identifying predicate-value pair and unbound predicate

Predicates are domain terms, tagged as noun (/NN/) or as noun phrases (/NP/). We look at phrase structure to extract noun phrases, along with associated value with numerical tag (/CD/). Predicates that have an associated 'value' makes up a 'predicate-value' pair. Values may be simple numerical values present in the note. Noun phrases that do not have any value associated with them are marked as the Unbound predicate.

5.3.2   Analyze typed dependencies and identify relationship types

Dependency parser is used to find out the dependent nouns and their relationships through dependency labels such as (nsubj, pobj, prep, number, etc). English language words like 'include", "deducted from", "involve", "comprises of" etc. imply certain relationship with respect to predicates preceding or succeeding them. We maintain a dictionary of such words, with the corresponding domain operation that each one implies. When we find one, we look for nearby Noun phrases to get the correlated predicates.

5.3.3   Verify relationship among predicates

Sometimes identifying existence of a relation between predicates is not enough; there needs to be a mechanism to find and confirm the association between these predicates. We do this with the help of domain ontology which has information about association of various domain terms in the presentation-view along with their calculation-view.

Presentation view (parent–child relationship): Domain terms are represented in hierarchically structure. The diagram below shows one such view in the financial domain (figure 2).
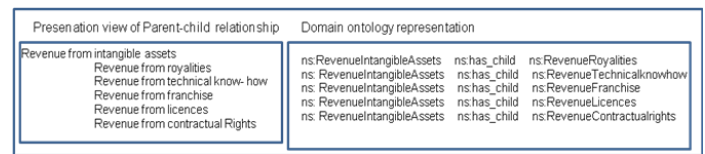
Figure 2: Presentation view of domain terms in Ontology

As shown in diagram, domain terms like "`Revenue from intangible assets`" has an association of parent-child with other domain term "`Revenue from royalties`". In domain ontology such relationship is expressed as "`has_child`" with parent and child domain terms.

Calculation view(Mathematical form of relationship): In financial reports, domain terms either add up to or subtracted from other domain terms, and is indicated by words as "`include`", "`comprises of`" etc. Such mathematical relationship is expressed in Calculation view. For example in Figure 3, "`Tangible assets net`" has children "`Tangible assets gross`" , "`Accumulated depreciation`" and "`Accumulated impairment`"; out of which first child needs to be added and rest two will be subtracted.
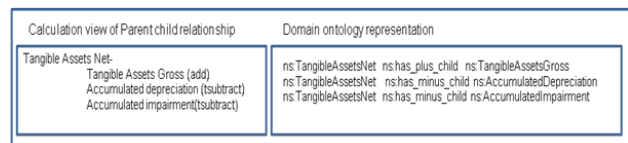
Figure 3 Domain terms, calculation view and their representation in Ontology

The identified concepts (or predicates) are searched over domain ontology to get the Presentation as well as Calculation view, in order to know which concepts are correlated and how. Once found, association from domain ontology can be verified from the relationship identified from Step 2. The result is the basic correlations among the concepts.

5.4 Finding Semantic correlation in structured data

Having identified the basic correlations among the concepts the next task is to determine the semantic correlations between the concepts. As discussed before, the correlations can be in the form of equations, rules, comparisons and constraints. Based on the verbs and the typed-dependencies in the parse, we construct the full semantic correlation among the concepts and structured data. Verbs such as "include" help form the *equation* kind of correlation. Verbs such as "when" and "in the situation" help form *rules* or *constraints*. This step is then carried out to identify all semantic correlations with the values and terms in the structured data

6   A detailed example

To illustrate the approach, we consider the unstructured notes in the financial report of an XYZ company, for the fiscal year 2008-09 (Figure 4). Let us consider Note2. Using Stanford Chunker we get the following chunks:

Note2: *"Long term investments($3456.7 million) include long-term debt-securities($786.9million),traded mutual funds($ 804.1 million),current debt securities($1540.3million) and the unutilized money investmented which was $324.5million."*
Chunk set N2C1=*[[[Long term investments][$3456.7million]] [(include)[(long-term debt-securities($786.9million))] [traded][(mutual funds($804.1million))]][,][current debt securities($1540.3 million))][and][[the unutilized money investmented [which was $324.5 million]]]].* The concepts are:

Table2: Concepts terms identified

Concepts present in Note2	Actual Domain Term
Long term debtsecurities	Debt securities, Long term
Traded mutual funds	Mutual funds, trade
Unutilized money investmented	Balance of Unutilized money investmented
Long term investments	Long term investments

Figure 4 Snippet of financial report of XYZ Company for fiscal year 2008-09

Using our approach for identification of Domain terms in the input sentence we get Direct-match set N2C2 = ["Long term investments"] and Indirect match set N2C3= ["long term debt securities", "traded mutual funds", "unutilized money investmented"]. The combined set of concepts, N2C4=["long term investments", "long term debt securities", "traded mutual funds", "unutilized money investmented"].
Since chunk set from N2C1 for the example Note2 does not match ("Traded mutual funds" is chunked separately in N2C1), we consider the concept identified from Step 2 and re-arrange the chunks in N2C1, giving us the set N2C5.
    N2C5 = [[(Long term investments $ 3456.7 million)] [(include) [(long-term debt-securities ($786.9 million))] ([traded mutual funds ($804.1million))] (,) (current debt securities ($1540.3million)(and) (the unutilized money investmented which was $324.5 million)]]]
Following our steps we find that every domain term in the sentence (Note2) is followed by some numerical value, which led us to the identify such predicate-value pairs as:
Long-term investments   = $3456.7 million
Long-term debt securities   = $ 786.9 million
Traded mutual funds   = $804.1 million
Current debt securities   = $1540.3 million
Balance of unutilized money investmented=$324.5 million
As another example, taking the sentence from Note 1, "The intangible assets ($10365.3 million) included accumulated amortization expense deducted from gross carrying amount". The identified predicate-values are:
Bound predicate: Intangible assets = $10365.3 million
Unbound predicates: Accumulated amortization expense, Gross carrying amount
Typed dependencies and parse tree structure of Note 1 reveals that "include" type of relationship that occurs between

predicates identified during the parse, interprets to *addition* of all identified predicates. From domain ontology we come to know that in Presentation view, predicate "Long-term investments" is Parent of all predicates found in the sentence, having a parent-child relationship. The Calculation view shows that predicates have "has_plus_child" relationship among them with parent predicate. Thus we are able to derive the relationship:

```
Long-term investments = Long-term debt
securities + traded mutual funds +
current debt securities + unutilized
money investmented
```

In above examples, we have considered equation comprising of Equality operator, including addition/subtraction of predicates associated.

7   Experimental results

Considering the various financial reports, such as Balance sheet, Income statement and Cash-flow statements, we asked a financial domain expert to review and evaluate the result. Our evaluation had almost 24 such reports, each one having structured table as well as underlying unstructured data, with notes section having at least 5-6 statements. When compared with actual relationship existing between Domain terms, the following observations were made, out of 144 statements (Table 3).

Table 3: Evaluation results

Input statements	Output analysis	Possible reasons
Set of 78 simple statements, implying direct relationship among domain terms and easily identified predicate-value binding	63 statements were successfully converted. Precision- 80.7% Recall - 92% F-measure – 0.85%	Poor parsing and syntactic analysis of complex and long sentences. Irrelevant domain validation of chunks.
45 statements comprised of multiple domain terms, having intricate relationship among elements.	22 statements were successfully converted. Precision – 48.8% Recall – 87% F-measure – 0.63%	Comparison of concepts with time period, amounted to confusion in understanding of domain terms and incorrect predicate-value binding.
Remaining 21 statements were long ones , referencing more	8 statements gave partial correlations .	Domain terms referred from
than one table , syntactically ambiguous and not easily interpreted	Precision- 38.09% Recall – 85% F-measure -0.53%	multiple tables were correlated, which seemed complex to identify and resolve.

We are currently working on finding a feasible solution to these problems.

8   Future work

Our future work will be focused on finding temporal relationships between concepts in structured data. As temporal relationships are of two types- 1)Causal (if an event takes place due to certain constraints) and 2) Purpose (if result of action is realized in future for a particular decision. Such temporal relationships play a major role in financial reports. We would also like to identify discourse connectives at inter-sentence level in the unstructured data. It will help improve our system accuracy.

9   Conclusions

We have described an approach to obtain semantic correlations among structured data from unstructured notes in reports. To extract correlations from the notes, the system should be robust enough to parse the sentence with respect to the domain ontology that consists of concepts and relations among the concepts. Identification of concepts is achieved through direct and indirect match with the domain ontology. Interaction between these concepts is determined with help of dependency relations and relationships among terms in the domain ontology. These relationships help in determining the semantic correlations between terms in the structured data described in the reports. Correlations can then help a user to understand the structured data or act as the backbone of an automated question-answering system.

10   References

[1]   Imran R. Mansuri and Sunita Sarawagi. Integrating Unstructured Data into Relational Da-tabases, Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), p.29, April 03-07, 2006.

[2]   Borthwick, J. Sterling, E. Agichtein, and R. Grishman. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In Sixth Workshop on Very Large Corpora New Brunswick, New Jersey. Association for Computational Linguistics. 1998.

[3]   W. W. Cohen and S. Sarawagi. Exploiting dictionaries in named entity extraction: Com-bining semi-markov extraction processes and data integration methods. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004.

[4] Shailly Goyal, Shefali Bhatt, Shailja Gulati and C Anantaram. Ontology-Driven Approach to Obtain Semantically Valid Chunks for NL-Enabled Business Applications,pages 105-117, CICLing 2010.

[5] Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Gene-rating Typed Dependency Parses from Phrase Structure Parses. In LREC 2006.

[6] R.S. Boyer and J.S. Moore, A fast string searching algorithm, Comm. ACM 20 (1977) 762-772.

[7] Hynes, G. E., & Bexley, J. B. 2004. "The contribution of banks' annual report writing qual-ity to investor decision-making", Journal of Commercial Banking and Finance, vol. 3, no 2, pp. 113-122.

[8] Courtis, J. K. 1998. "Annual report readability variability: tests of the obfuscation hypo-thesis",Accounting, Auditing & Accountability Journal, vol. 11, no. 4, pp. 459-471.

[9] Subramanian, R., Insley, R. G., & Blackwell, R. D. 1993. "Performance and Readability: A Comparison of Annual Reports of Profitable and Unprofitable Corporations", Journal of Business Communication, vol. 30, no. 1, pp. 49-61.

[10] Diana Maynard, Horacio Saggion, Milena Yankova, Kalina Bontcheva, Wim Peters. Natural Language Technology for Information Integration in Business Intelligence. Lecture Notes in Computer Science, Volume 4439, pages 366-380, Springer, 2007

Emotion Control of Induction Motor Using Disturbance Acceptance and Bacteria Foraging

Dong Hwa Kim

Dept. of Electronic and Control Eng., Hanbat National University
16-1 San Duckmyong-Dong Yuseong-Gu
Daejon City, Korea
e-mail: koreahucare@gmail.com

Abstract— **Abstract. In this paper, emotion control approach of PID controller with disturbance acceptance function against external emotion in control system is proposed using bacterial foraging based optimal algorithm. Up to the present time, PID Controller has been used to operate for disturbance reject because disturbance is not used in practice structure. However, this paper deals with disturbance positively for emotion control. Also, to obtain optimal control gain, this paper use bacterial foraging algorithm. This paper illustrate disturbance acceptance conditions and ITSE (Integral of time weighted squared error) as the performance index. The parameters of PID controller are selected by bacterial foraging based optimal algorithm to obtain the required response for emotion control.**

Keywords: **PID control; Disturbance control; Bacterial algorithm, Optimal algorithm; Emotion control.**

I. INTRODUCTION

A Proportional – Integral – Derivative (PID) controller has been widely used in the most industrial processes despite continual advances in control theory. This is not only due to the simple structure which is theoretically easy to understand but also to the fact that the tuning technique provides adequate performance in the vast majority of applications. However, it cannot effectively control such a complicated or fast running system such as disturbance control system, since the response of a plant depends on only the three parameters (P, I, and D) and its gain has to be manually tuned by trial and error in the industrial world. Most of the PID tuning rules developed in the past years use the conventional method such as frequency-response methods [1-10] and disturbance rejection. Recently, the artificial intelligence approach is available for general linear control system [11, 27-30], bacterial foraging tuning [21]. On the other hand, the emotional function of the human mind can give an impact on decision-making, memory, action, and so on [26, 28, 29]. So, recently many articles and researchers have been interesting in implement such as modeling, control, express, and robot, and so on [26-30]. Since natural selection of bacterial foraging tends to eliminate animals with poor foraging strategies for locating, handling, and ingesting food, optimization models can be provided for social foraging where groups of parameters communicate to cooperatively forage in engineering.

In this paper, we deal with emotion control by disturbance acceptance and an intelligent tuning method of PID controller by bacterial foraging based optimal algorithm is suggested.

II. EMOTION PID CONTROLLER BY DISTURBANCE ACCEPTANCE

A. Condition for Disturbance Acceptance
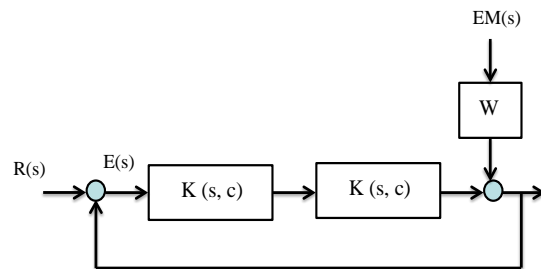
Fig. 1. Emotion control system with disturbance acceptance.

In Fig. 1, the disturbance acceptance constraint can be given by [7,8]

$$\max_{d(t)\in D}\frac{\|Y\|}{\|d\|}=\left\|\frac{w(s)}{1+K(s,c)G(s)}\right\|_{\infty}\langle\delta. \qquad (1)$$

Here, $\delta\langle 1$ is constant defining by the desired acceptance level and $\|\bullet\|_{\infty}$ denotes the H_{∞}-norm, which is defined as

$$\|G(s)\|_{\infty}=\max_{\omega\in[0,\infty)}|G(j\omega)|. \qquad (2)$$

The disturbance acceptance constraint becomes

$$\left\| \frac{w(s)}{1 + K(s,c)G(s)} \right\|_\infty =$$

$$\max_{\omega \in [0,\infty)} \left(\frac{w(j\omega)w(-j\omega)}{1 + K(j\omega,c)G(j\omega,c)K(-j\omega,c)G(-j\omega,c)} \right)^{0.5}$$

$$= \max_{\omega \in [0,\infty)} \left(\sigma(\omega,c) \right)^{0.5} \tag{3}$$

The controller K (s, c) is written as

$$K(s,\, c) = c_1 + \frac{c_2}{s} + c_3 s \tag{4}$$

The vector c of the controller parameter is given by

$$c = \begin{bmatrix} c_1, & c_2, & c_3 \end{bmatrix}^T \tag{5}$$

Hence, the condition for disturbance acceptance is given as

$$\max_{\omega \in [0,\infty)} \left(\sigma(\omega,c) \right)^{0.5} \langle \delta. \tag{6}$$

B. Performance Index for Disturbance Acceptance emotion Controller Tuning

The performance index defined as ITSE (Integral of the Time-Weighted Square of the Error) is written by While to perform social foraging an animal needs communication capabilities,

$$E(s) = \frac{B(s)}{A(s)} = \frac{\sum\limits_{j=0}^{m} b_j s^m}{\sum\limits_{i=0}^{n} a_i s^n}. \tag{7}$$

Because *E(s)* contains the parameters of the controller *(c)* and plant, the value of performance index, *PI* for a system of n*th* order can be minimized by adjusting the vector c as follows [7]:

$$\min_{c} PI(c) \tag{8}$$

The optimal tuning proposed in this paper is to find the vector c, such that the ITSE performance index, *PI (c)* is a minimum using bacterial algorithm and the constraint $\max_{\omega \in [0,\infty)} \left(\sigma(\omega,c) \right)^{0.5} \langle \delta$ is satisfied through real coded bacterial algorithms.

C. Optimal Computing of Bacteria Foraging

The characteristics of poor foraging can be applied to have an optimal solution through methods for locating, handling, and ingesting food. After many generations, a foraging animal takes actions to maximize the energy obtained per unit time spent foraging. That is, poor foraging strategies are

either eliminated or shaped into good ones. So, this computing has capabilities optimal solution against environment. Foraging theory is described [11, 12, 15, 21-24]. Selection behavior as bacteria forage performs selection for emotion function as individuals sad, anger, happy. While to perform emotional foraging to user demand, each foraging steps needs communication and individual emotion functions can obtain satisfactory results.

III. EMOTION CONTROL BY CHEMOTACTIC BEHAVIOR OF E. COLI.

This paper considers the foraging behavior of E. coli, which is a common type of bacteria as in reference 4-5. Its behavior to move comes from a set of up to six rigid 100–200 rps spinning flagella, each driven as a biological motor. An E. coli bacterium alternates between running and tumbling. Running speed is 10–20 $\mu m/\sec$, but they cannot swim straight. When we can summarize the chemotactic actions of bacteria as the following description:

- If in neutral medium, alternate tumbles and runs, its action is having search.

- If swimming up a nutrient gradient (or out of noxious substances), swim longer (climb up nutrient gradient or down noxious gradient), its behavior seeks increasingly favorable environments.

- If swimming down a nutrient gradient (or up noxious substance gradient), then search action is avoiding unfavorable environments.

So, it can climb up nutrient hills and at the same time avoid noxious substances. The sensors it needs for optimal resolution are receptor proteins which are very sensitive and high gain. That is, a small change in the concentration of nutrients can cause a significant change in behavior. This is probably the best-understood sensory and decision-making system in biology.

Mutations in E. coli affect the reproductive efficiency at different temperatures, and occur at a rate of about 10^{-7} per gene and per generation. E. coli occasionally engages in a conjugation that affects the characteristics of a population of bacteria. Since there are many types of taxes that are used by bacteria such as, aerotaxis (it are attracted to oxygen), light (phototaxis), temperature (thermotaxis), magnetotaxis (it it can be affected by magnetic lines of flux. Some bacteria can change their shape and number of flagella which is based on the medium to reconfigure in order to ensure efficient foraging in a variety of media. Bacteria can form intricate stable spatio-temporal patterns in certain semisolid nutrient substances. They can eat radially their way through a medium if placed together initially at its center. Moreover, under certain conditions, they will secrete cell-to-cell attractant signals so that they will group and

protect each other. These bacteria can swarm.

A. Optimization Function of Bacterial Swarm Foraging

The main goal based on bacterial foraging is to apply in order to find the minimum of $P(\phi), \phi \in R^n$, not in the gradient $\nabla P(\phi)$. Here, when ϕ is the position of a bacterium, and $J(\phi)$ is an attractant-repellant profile. That is, it means where nutrients and noxious substances are located, so *P<0, P=0, P>0* represent the presence of nutrients. A neutral medium, and the presence of noxious substances, respectively can showed by

$$H(j,k,l) = \{\phi^i(j,k,l) \big| i = 1,2,...,N\}. \quad (9)$$

Equation represents the positions of each member in the population of the *N* bacteria at the *j*th chemotactic step, *k*th reproduction step, and *l*th elimination-dispersal event. Let *P(i, j, k, l)* denote the cost at the location of the *i*th bacterium $\phi^i(j,k,l) \in R^n$. Reference [20, 21] let

$$\phi^i = (j+1,k,l) = \phi^i(j,k,l) + C((i)\varphi(j), \quad (10)$$

so that *C(i)>0* is the size of the step taken in the random direction specified by the tumble. If at $\phi^i(j+1,k,l)$ the cost *J(i, j+1, k, l)* is better (lower) than at $\phi^i(j,k,l)$, then another chemotactic step of size *C(i)* in this direction will be taken and repeated up to a maximum number of steps N_s. N_s is the length of the lifetime of the bacteria measured by the number of chemotactic steps. Functions $P_c^i(\phi)$, *i*=1, 2, . . . , *S*, to model the cell-to-cell signaling via an attractant and a repellant is represented by [20, 21]

$$P_c(\phi) = \sum_{i=1}^{N} P_{cc}^i = \sum_{i=1}^{N} \left[-L_{attract} \exp\left(-\delta_{attract} \sum_{j=1}^{n} (\phi_j - \phi_j^i)^2 \right) \right]$$
$$+ \sum_{i=1}^{N} \left[-K_{repellant} \exp\left(-\delta_{attract} \sum_{j=1}^{n} (\phi_j - \phi_j^i)^2 \right) \right], \quad (11)$$

When we where $\phi = [\phi_{1,...,}\phi_p]^T$ is a point on the optimization domain, $L_{attract}$ is the depth of the attractant released by the cell and $\delta_{attract}$ is a measure of the width of the attractant signal. $K_{repellant} = L_{attract}$ is the height of the repellant effect magnitude), and $\delta_{attract}$ is a measure of the width of the repellant. The expression of $P_c(\phi)$ means that its value does not depend on the nutrient concentration at position ϕ. Model use the function $P_{ar}(\phi)$ to represent the environment-dependent cell-to-cell signaling as

$$P_{ar}(\phi) = \exp(T - P(\phi))P_c(\phi) \quad (12)$$

where *T* is a tunable parameter. Model consider

minimization of $P(i, j, k, l) + P_{ar}(\phi^i(j,k,l))$, so that the cells will try to find nutrients, avoid noxious substances, and at the same time try to move toward other cells, but not too close to them. The function $P_{ar}(\phi^i(j,k,l))$ implies that, with *M* being constant, the smaller $P(\phi)$, the larger $P_{ar}(\phi)$ and thus the stronger attraction, which is intuitively reasonable. In tuning the parameter *M*, it is normally found that, when *M* is very large, $P_{ar}(\phi)$ is much larger than $J(\phi)$, and thus the profile of the search space is dominated by the chemical attractant secreted by E. coli. On the other hand, if *T* is very small, then $P_{ar}(\phi)$ is much smaller than $P(\phi)$, and it is the effect of the nutrients that dominates. In $P_{ar}(\phi)$, the scaling factor of $P_c(\phi)$ is given as in exponential form.

B. Computing Procedure for Optimization Acceptance of Emotion Function

This paper describes the method in the form of an algorithm to search optimal value of PID parameter [22-24].

[step 1] Initialize parameters *n, N, N_C, N_S, N_{re}, N_{ed}, P_{ed}, C(i)(i=1,2,...,N), ϕ^i*, and random values of PID parameter. Where,

n: Dimension of the search space (Each Parameter of PID controller),

N: The number of bacteria in the population,

N_C: chemotactic steps,

N_{re}: The number of reproduction steps,

N_{ed}: the number of elimination-dispersal events,

P_{ed}: elimination-dispersal with probability,

C(i): the size of the step taken in the random direction specified by the tumble. The controller parameter is searched in the range of Kp=[0 30], Ti=[0 30], and Td=[0 30].

[step 2] Elimination-dispersal loop: *l=l+1*

[step 3] Reproduction loop: *k=k+1*

[step 4]Chemotaxis loop: *j=j+1*

[substep a] For *i =1,2,…,N*, take a chemotactic step for bacterium *i* as follows.

[substep b] Compute ITSE (*i ,j, k, l*).

[substep c] Let ITSE$_{last}$=ITSE (*i,j,k,l*) to save this value since we may find a better cost via a run.

[substep d] Tumble: generate a random vector $\Delta(i) \in R^n$ with each element $\Delta_m(i), m = 1,2,..., p,$ a randonumber on [-1, 1].

[substep e] Move: Let
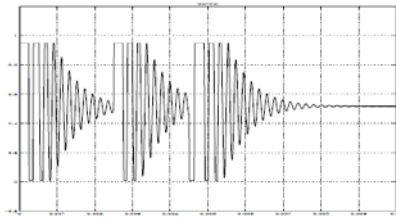
$$\phi^i(j+1,k,l) = \phi^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$$

This results in a step of size *C(i)* in the direction of the tumble for bacterium *i*.

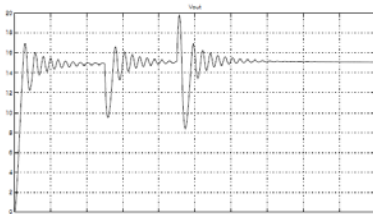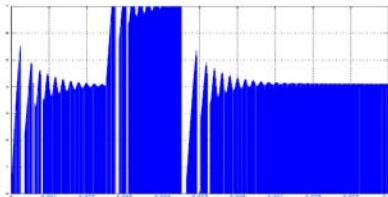[substep f] Compute ITSE $(i, j+1, k, l)$.
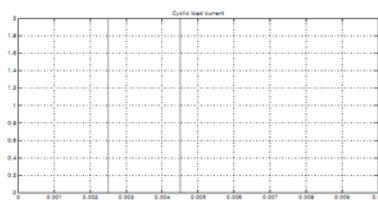
[substep g] Swim

Fig. 2. Traditional PID tuning with disturbance.

Fig. 3. Bacterial foraging based PID tuning with disturbance acceptance.

Fig. 4. PID based tuning for emotion signal.

Fig. 5. Source emotional signal.

IV. CONCLUSION

In this paper, we introduce the characteristics of bacterial foraging into emotion computing with disturbance rejection. Basically, we have been thing that disturbance was not useful in control system. However, in this paper, we introduce that un-useful disturbance into our emotion control system positively by adding traditional control structure. Of course, to use effectively, we need to research in real experimental system and industrial in the future.

REFERENCES

[1] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control system engineering: A survey," Control Eng. Practice, vol. 10, pp. 1223–1241, 2002.

[2] R. A. Krohling and J. P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithms," IEEE Trans. Evol. Comput., vol. 5, pp. 78–82, Feb. 2001.

[3] PASSINO, K. M., "Biomimicry of Bacterial foraging for Distributed Optimization," University Press, Princeton, New Jersey, 2001.

[4] PASSINO, K. M., "Biomimicry of Bacterial Foraging for distributed Optimization and Control," IEEE Control Systems Magazine, 2002.

[5] Farmer, J. D., Packard, N. H and Perelson, A. S, "The immune system, adaptation, and machine learning," Vol. Physica. D, No. 22, (1986) 187 – 204.

[6] Kazuyuki Mori and Makoto Tsukiyama, "Immune algorithm with searching diversity and its application to resource allocation problem," Trans. JIEE, Vol. 113 - C, No. 10, (1993).

[7] Dong Hwa Kim, "Robust tuning of PID controller with disturbance rejection using bacterial foraging based optimization," International symposium on computational intelligent and industrial application (ISCIIA2004), Dec. 20-22, 2004, Hikou, China.

[8] Dong Hwa Kim, "Robust PID controller tuning using multiobjective optimization based on clonal selection of immune algorithm," Proc. Int. Conf. Knowledge-based intelligent information and engineering systems. Springer-Verlag. pp. 50-56, 2004.

[9] Z. Michalewicz, "Genetic Algorithms + Data Structures=Evolution Programs," Springer-Verlag, Berlin Heideberg, 1996.

[10] Dong Hwa Kim, "Robust Tuning of PID Controller With Disturbance Rejection Using Bacterial Foraging Based Optimization", ACC2005, Portland, 2005.

[11] Dong Hwa Kim , "Design and Tuning Approach of 3-DOF Emotion Intelligent PID(3-DOF-PID) Controller", IEEE-EMS2013 conference Nov.11-12, 2012. Malter, 2012.

Ontology-based Heuristic Algorithm for Text Classification in Oil Industry Application

Nayat Sanchez-Pi[1], Luis Marti[2], and Ana Cristina Garcia Bicharra[3]
[1] Instituto de Lógica, Filosofia e Teoria da Ciência, Niterói, RJ, Brazil
[2] Electrical Engeenering Department, Pontificia Universidade Catolica, RJ, Brazil
[2] Computer Science Institute, Fluminense Federal University, RJ, Brazil

Abstract - *Today, with the advances of new technologies, accidents, incidents and occupational health records are stored in diverse repositories. The amount of information of HSE that is daily generated has become increasingly huge. Most of this information is stored as unstructured data. The challenge of unstructured data is a top priority for industries that are looking for ways to search, sort, analyze and extract knowledge from masses of data. In this paper we provide with a solution to oil and gas industry for accident investigation using information extraction techniques. The main objective is to propose and evaluate information extraction techniques in occupational health control process, particularly, for automatic detection of accidents from unstructured texts. Our proposal divides the problem in subtasks such as text analysis, recognition and classification of failed occupational health control, resolving accidents.*

Keywords: Information retrieval, ontology, text classification, text recognition, text analysis, intelligent systems

1 Introduction

There is an important effort of oil and gas industry to reduce the number of accidents and incidents. There are standards to identify and record workplace accidents and incidents to provide guiding means on prevention efforts, indicating specific failures or reference, means of correction of conditions or circumstances that culminated in accident. Besides, oil and gas industry is increasingly concerned with achieving and demonstrating good performance of occupational health and safety (OHS), through the control of its OHS risks, consistent with its policy and objectives of OHS.

Health, Safety and Environment (HSE) continues to be a priority issue for the offshore oil and gas industry and a determining factor in its overall success. Years passed since community takes into account the implications of oil industry to Health, Safety and the Environment. Oil and gas industries are frequently in the news. Much of the time this news is related to changes in prices of oil and gas. Another less frequent subject of media attention is when disasters strike, as in the offshore oil drilling platform explosion and fire.

The development of automatic methods to produce structured information from unstructured text sources would be extremely valuable to the oil industry. A structured resource would allow researches and industry professionals to write a single query to retrieve all the transcription interactions of any accident. Instead of the thousands of abstract provided by querying the unstructured corpus, the query on the structured corpus might result in a few hundred well-formed results; this would obviously save a tremendous amount of time and energy.

The main objective is to propose and evaluate information extraction techniques in occupational health control process, particularly, for automatic detection of accidents from unstructured texts. Our proposal divides the problem in subtasks such as text analysis, recognition and classification of failed occupational health control, resolving accidents. We present an ontology-based approach to the automatic text categorization. An important and novel aspect of this approach is that our categorization method does not require a training set, which is in contrast to the traditional statistical and probabilistic methods that require a set of pre- classified documents in order to train the classifier.

2 State of the art

Automatic text categorization is a task of assigning one or more pre-specified categories to an electronic document, based on its content. Nowadays, text classification is extensively used in many contexts. One of the examples is the automatic classification of incoming electronic news into categories, such as entertainment, politics, business, sports, etc. Standard categorization approaches utilize statistical or machine learning methods to perform the task. Such methods include Naïve Bayes [1], Support Vector Machines [2], Latent Semantic Analysis [3] and many others. A good overview of the traditional text categorization methods is presented in [4]. All of these methods require a training set of pre-classified documents that is used for classifier training; later, the classifier can correctly assign categories to other, previously unseen documents.

However, it is often the case that a suitable set of well categorized (typically by humans) training documents is not available. Even if one is available, the set may be too small,

or a significant portion of the documents in the training set may not have been classified properly. This creates a serious limitation for the usefulness of the traditional text categorization methods.

As described by the World Wide Web Consortium (W3C), ontology defines the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (a domain is just a specific subject area of knowledge, such as medicine, real estate, automobile repair, or financial management). More specifically, ontology is a data model that represents a set of concepts (entities) within a given domain and the relationships between those concepts. It is used to reason about the concepts within that domain.

In this paper, we introduce a novel text categorization method based on leveraging the existing knowledge represented in a domain ontology. The novelty of this approach is that it is not dependent on the existence of a training set, as it relies solely on the entities, their relationships, and the taxonomy of categories represented in the ontology.

In the proposed approach, the ontology effectively becomes the classifier. Consequently, classifier training with a set of pre-classified documents is not needed, as the ontology already includes all important facts. The proposed approach requires a transformation of the document text into a graph structure, which employs entity matching and relationship identification. The categorization is based on measuring the semantic similarity between the created graph and categories defined in the ontology require a training set of pre-classified documents that is used for classifier training; later, the classifier can correctly assign categories to other, previously unseen documents.

However, it is often the case that a suitable set of well-categorized (typically by humans) training documents is not available. Even if one is available, the set may be too small, or a significant portion of the documents in the training set may not have been classified properly. This creates a serious limitation for the usefulness of the traditional text categorization methods.

In this paper, we introduce a novel text categorization method based on leveraging the existing knowledge represented in domain ontology. The novelty of this approach is that it is not dependent on the existence of a training set, as it relies solely on the entities, their relationships, and the taxonomy of categories represented in the ontology. In the proposed approach, the ontology effectively becomes the classifier. Consequently, classifier training with a set of pre-classified text is not needed, as the ontology already includes all relevant facts. The proposed approach requires a transformation of the unstructured text into a graph structure, which employs entity matching and relationship identification. The categorization is based on measuring the

semantic similarity between the created graph and categories defined in the ontology.

3   Proposal

Our proposal strategy is to use an ontology as the key component of our text classification heuristic algorithm. Besides the ontology itself, the algorithm is composed of a set of modules:

1.   A lemmatization, stemming and stop-word removing preprocessing. In this work we applied for this task the functionality provided by the Apache Lucence framework [5].

2.   A thesaurus for locating words appearing in the text in the ontology. In our case we used a customized version of OpenOffice Brazilian Portuguese thesaurus [6].

3.   Set of ontology elements tagged with its corresponding classification label.

4.   A thesaurus crawling algorithm that takes care of determining the matching degree of text words with a corresponding ontology term.

3.1   Ontology

Ontologies offer knowledge that is organized in a more structural and semantic way. Their use in text categorization and topic identification has lately become an intensive research topic. As ontologies provide named entities and relationship between them, an intermediate categorization step requires matching terms to ontological entities. Afterwards, an ontology can be successfully used for term disambiguating and vocabulary unification, as presented in [7]. Another approach, presented in [8], reinforces co-occurrence of certain pairs of words or entities in the term vector that are related in the ontology. The use of descriptions of neighboring entities to enrich the information about a classified document is described in [9]. Interesting approach, although very different, is presented in [10]; where authors automatically build partial ontology from the training set to improve keyword-based categorization method. Other categorization approaches based on using recognized named entities are described in [11] and [12].

An ontology is defined as "an explicit specification of a conceptualization" [13]. An ontology created for a given domain includes a set of concepts as well as relationships connecting them within the domain. Collectively, the concepts and the relationships form a foundation for reasoning about the domain.

A comprehensive, well-populated ontology with classes and relationships closely modeling a specific domain represents a vast compendium of knowledge in the domain. It is only

natural to expect that having such a comprehensive knowledge about the domain, one should be well equipped to create software systems implementing a variety of tasks concerning the domain of the ontology. Recently, ontologies have been used in various semantic applications, ranging from business analytics to semantic data integration [14].

We believe that the knowledge represented in such a comprehensive ontology can be used to identify topics (concepts) in a text document, provided the document thematically belongs to the domain represented in the ontology. Furthermore, if the concepts in the ontology are organized into hierarchies of higher-level categories, it should be possible to identify the category (or a few categories) that best classify the content of the document.

Within the area of computing, the ontological concepts are frequently regarded as classes that are organized into hierarchies. The classes define the types of attributes, or properties common to individual objects within the class. Moreover, classes are interconnected by relationships, indicating their semantic interdependence (relationships are also regarded as attributes) [15]. Class hierarchies and class relationships form the schema level of the ontology, while the individuals (object instances or just instances) and links among them (relationship instances) form the so-called ground level of the ontology.

We built a domain ontology for the Health, Safety and Environment (HSE) of oil and gas domain, see Figure 1. We also obtain the inferences that describe the dynamic side and finally we group the inferences sequentially to form tasks. Principal concepts of the ontology, see Figure 2, are the following:

- **Anomaly:** Undesirable event or situation which results or may result in damage or faults that affect people, the environment, equity (own or third party), the image of the Petrobras System, products or production processes. This concept includes accidents, illnesses, incidents, deviations and non-conformances.

  o **Neglect**: Any action or condition that has the potential to lead to, directly or indirectly, damage to people, to property (own or third party) or environmental impact, which is inconsistent with labor standards, procedures, legal or regulatory requirements, requirements management system or practice.

    ▪ **Behavioral neglect:** Act or omission which, contrary provision of security, may cause or contribute to the occurrence of accidents.

    ▪ **Non-behavioral neglect:** Environmental condition that can cause an accident or contribute to its occurrence. The environment includes adjective here, everything that relates to the environment, from the atmosphere of the workplace to the

facilities, equipment, materials used and methods of working employees who is inconsistent with labor standards, procedures, legal requirements or normative requirements of the management system or practice.

o **Incident:** Any evidence, personal occurrence or condition that relates to the environment and / or working conditions, can lead to damage to physical and / or mental.

o **Accident:** Occurrence of unexpected and unwelcome, instant or otherwise, related to the exercise of the job, which results or may result in personal injury. The accident includes both events that may be identified in relation to a particular time or occurrences as continuous or intermittent exposure, which can only be identified in terms of time period probable. A personal injury includes both traumatic injuries and illnesses, as damaging effects mental, neurological or systemic, resulting from exposures or circumstances prevailing at the year's work force. In the period for meal or rest, or upon satisfaction of other physiological needs at the workplace or during this, the employee is considered in carrying out the work.

  ▪ **Accident with injury:** It's all an accident in which the employee suffers some kind of injury.

    • **Injury**: Any damage suffered by a part of the human organism as a consequence of an accident at work.

      o **With leave:** Personal injury that prevents the injured from returning to work the day after the accident or resulting in permanent disability. This injury can cause total permanent disability, permanent partial disability, total temporary disability or death.

      o **Without leave:** Personal injury that does not prevent the injured to return to work the day after the accident, since there is no permanent disability. This injury, not resulting in death, permanent total or partial disability or total temporary disability, requires, however, first aid or emergency medical aid. Expressions should be avoided "lost-time accident" and "accident without leave", used improperly to mean, respectively, "with leave injury" and "injury without leave."

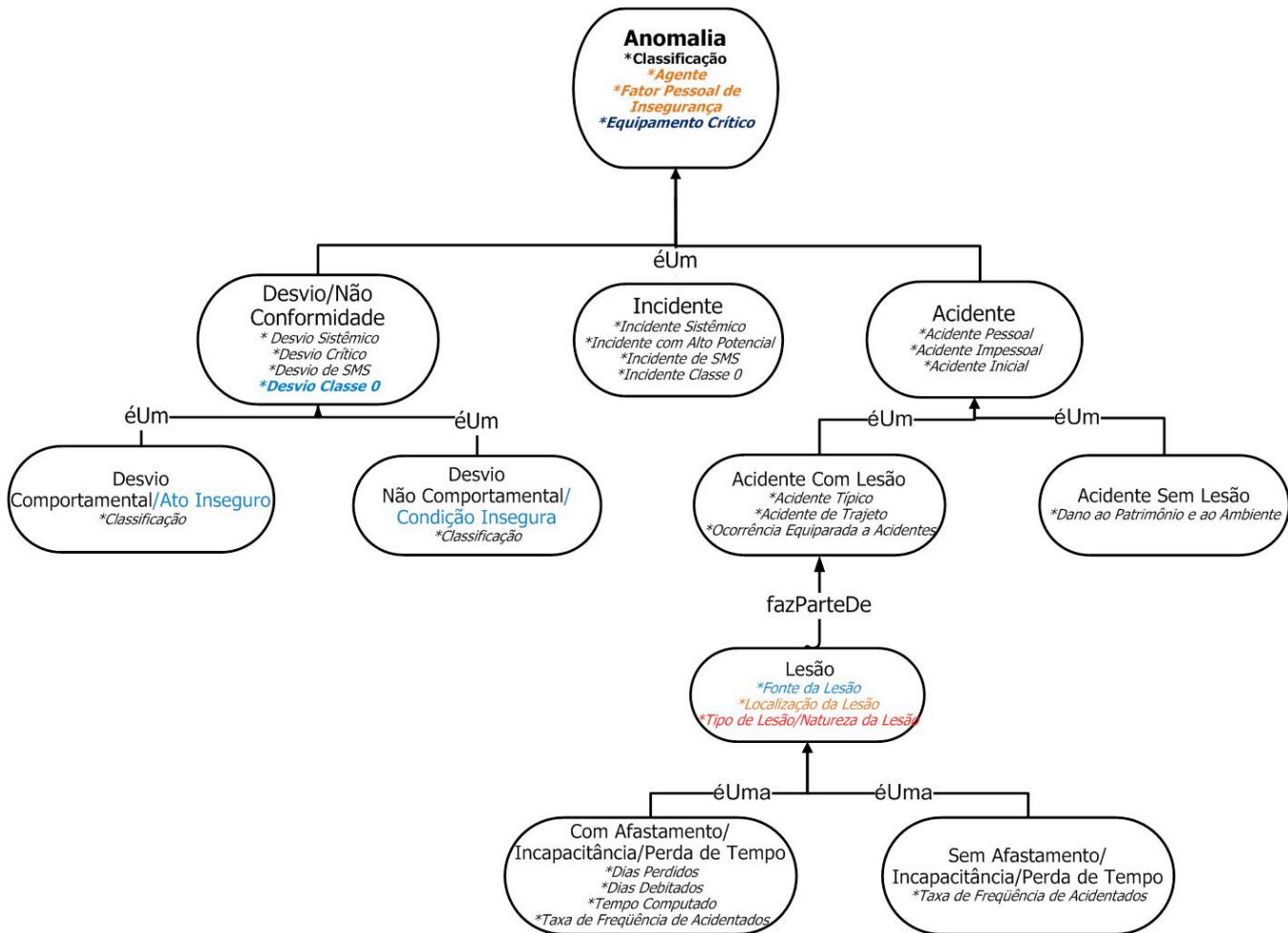- **Without Injury:** Accident causes no personal injury.

Figure 1. Ontology for Health, Safety and Environment in Oil Industry. *Anomaly* Concept.

3.2 Classification algorithm

As mentioned before, the classification algorithm proposed in this work relies on the previous ontology, a thesaurus to establish the degree of matching between a given sentence and some ontology terms of interest. The algorithm is presented as pseudo-code in Figure 1 as function ClassifyText(). It proceeds by first filtering and rearranging the input sentence in order to render it in a format suitable for processing (PreprocessText() method in step 1 of Figure 3). We have employed Apache Lucene text processing tools for stemming, lemmatization and stop-word removal.

Having the filtered text represented as a set of words, the algorithm proceeds to identify which terms of the ontology are most closely related to that set. It carries that out by invoking for each word the function ComputeSimilarityLevels(). This function —which is described in Figure 4— returns the set of ontology terms that are related with a given word by recursively traversing a thesaurus up to a given number of levels. If a connection between a word and a term is established that term is

included, along with its level of similarity in the set of related terms Θ. The level of similarity is defined as the number of jumps needed to get from to word to the term using the thesaurus. A lower level implies higher similarity.

The result of the classification is one or more ontology terms that are most closely related to the text, or, posed in other words, the terms with minimal level of similarity. It should be beard in mind that the two functions presented here have been simplified for didactical reasons, and in practice some a harder to read but more efficient option is used.

4 Conclusions

In this paper, we introduce a novel text categorization method based on leveraging the existing knowledge represented in domain ontology. The novelty of this approach is that it is not dependent on the existence of a training set, as it relies solely on the entities, their relationships, and the taxonomy of categories represented in the ontology.

```
function:  ClassifyText(t): ⟨Γ, l⟩
   input:  t: string – the text to be classified.
  output:  Γ := {w_0, ..., w_n} – set of ontology terms that
           can classify the current text.
           l: integer – level of similarity of the terms.

      1    Ω = PreprocessText(t) – yields a set of
           words after lemmatization, stemming and
           stop-word removal.
      2    l = ∞  – best similarity level.
      3    Γ = ∅.
      4    for each w_k ∈ Ω do
      5        Θ_k = ComputeSimilarityLevels(w_k, 0).
      6        for each ⟨w_j, l_j⟩ ∈ Θ_k
      7            if l_j < l then
      8                Γ = {w_j}.
      9                l = l_j.
     10            else if l_j = l then
     11                Γ = Γ ∪ {w_j}.
     12            end-if
     13        end-for each
     14    end-for each
     15    return ⟨Γ, l⟩.
```

Figure 2. Pseudo-code description of the algorithm used to compute the levels of similarity between a given word found in text and ontology terms.

The classification algorithm proposed herein has an adequate computational performance. However it has some clear drawbacks when confronted to complex and contradictory texts. This is not an issue for our application domain. In spite of the texts are written in a natural language, for this particular domain, unstructured texts are written in a very direct discourse and there was no a large variation in the amount of information in each text, issues that were good for the step 1. See Figure 4.

In the proposed approach, the ontology effectively becomes the classifier. Consequently, classifier training with a set of pre-classified documents is not needed, as the ontology already includes all important facts. The proposed approach requires a transformation of the document text into a graph structure, which employs entity matching and relationship identification. The categorization is based on measuring the semantic similarity between the created graph and categories defined in the ontology require a training set of pre-classified documents that is used for classifier training; later, the classifier can correctly assign categories to other, previously unseen texts.

```
function:  ComputeSimilarityLevels(w_0, l_0): Θ
   input:  w_0: string – the word to be processed.
           l_0: integer – initial level.
  global:  l_max: integer – maximum number of levels to
           be reached.
  output:  Θ := {⟨w_1, l_1⟩, ..., ⟨w_n, l_n⟩} – set of pairs of
           ontology term and level of similarity. Each
           pair represents how "close" is word w_n to w.

      1    if l_0 = l_max then
      2        return Θ = ∅ – max. number of levels
           reached.
      3    end-if
      4    if OntologyContains(w_0) then
      5        return Θ = {⟨w_0, l_0⟩} – the word is a
           term of the ontology, no further search is
           necessary.
      6    end-if
      7    Υ = Thesaurus(w_0) – determine the
           synonyms of w_0.
      8    Θ = ∅.
      9    for each w_k ∈ Υ do
     10
           Θ_k = ComputeSimilarityLevels(w_k, l_0 + 1).
     11        Θ = Θ ∪ Θ_k.
     12    end-for each
     13    return Θ.
```

Figure 3. Pseudo-code description of the algorithm used to compute the levels of similarity between a given word found in text and ontology terms.

In subsequent steps we intend to combine this algorithm with other machine learning approaches like Naïve Bayes, Support Vector Machines, etc. This combination will be capable of yielding a more general solution that is able to cope with situations that fall outside the scope covered by the ontology while preserving the accuracy of the ontology based classification

Acknowledgements

Figure 4. Preprocessing and algorithm results.

5   References

Number in square brackets ("[ ]") should cite references to the literature in the main text. List the cited references in numerical order at the very end of your paper (under the heading `References'). Start each referenced paper on a new line (by its number in square brackets).

[1] Lewis, D.D.: Naive (Bayes) at forty: The independence assumption in information retrieval. ECML-98, 10th European Conference on Machine Learning, Chemnitz, DE (1998)

[2] Vapnik, V.: The nature of statistical learning theory. Springer Verlag (1995)

[3] Deerwester, S., Dumais, S., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. Journal of the Society for Information Science (1990) 41 (1990) 391-407

[4] Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys (CSUR) 34 (2002) 1 – 47

[5] Gospodnetic, O.; Hatcher, E., McCandless M.: Lucene in Action (2nd ed.). Manning Publications. ISBN 1-9339-8817-7 (2009).

[6] -- DicSin: Dicionário de Sinônimos Protuguês Brasil. Apache   OpenOffice.org http://extensions.openoffice.org/en/project/DicSin-Brasil (2013)

[7] Bloehdorn, S., Hotho,A.: Text Classification by Boosting Weak Learners based on Terms and Concepts. 4th IEEE International Conference on Data Mining (ICDM'04) (2004)

[8] Nagarajan, M., Sheth, A.P., Aguilera, M., Keeton, K., Merchant, A., Uysal, M.: Altering Document Term Vectors for Classification - Ontologies as Expectations of Co-occurrence. LSDIS Technical Report (November, 2006)

[9] Gabrilovich, E., Markovitch, S.: Overcomingthe Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. 21th National Conference on Artificial Intelligence, Boston, MA, USA (2006)

[10] Wu, S.-H., Tsai, T.-H., Hsu, W.-L.: Text categorization using automatically acquired domain ontology. 6th international workshop on Information retrieval with Asian languages - Volume 11, Sapporo, Japan (2003)

[11] Sheth, A.P., Bertram, C., Avant, D., Hammond, B., Kochut, K.J., Warke, Y.: Semantic Content Management for Enterprises and the Web. IEEE Internet Computing July/August 2002 (2002)

[12] Hammond, B., Sheth, A.P., Kochut, K.J.: Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content. Real World Semantic Web Applications, IOS Press, 2002 (2002)

[13] Gruber, T.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5 (1993) 199-220, 1993

[14] Buccella, A., Cechich, A., Brisaboa, N. R.: Ontology-Based Data Integration. In: Rivero, L.C., Doorn, J.H., Ferraggine, V.E. (eds.): Encyclopedia of Database Technologies and Applications. Information Science Reference (2005)

[15] Sheth, A.P., Arpinar, I.B., Kashyap, V.: Relationships at the Heart of Semantic Web: Modeling, Discovering, and Exploiting Complex Semantic Relationships. In: Nikravesh, M., Azvin, B., Yager, R., Zadeh, L. (eds.): Enhancing the Power of the Internet: Studies in Fuzziness and Soft Computing. Springer Verlag (2003)