http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (6): 801-810, 2009
ISSN 1812-5638

Machine Learning Approach in Optimizing Negotiation Agents for E-Commerce

S.C. Ng, M.N. Sulaiman and M.H. Selamat
Faculty of Computer Science and Information Technology, Universiti Putra Malaysia,
43400 UPM Serdang, Selangor, Malaysia

Abstract: This study discusses the implementation of machine learning approach in negotiation agents that can learn their opponent's preferences and constraints during one-to-many negotiations. A novel mechanism in learning negotiation is introduced. The genetic-based model of multi-attribute one-to-many negotiation, namely GA Improved-ITA is proposed. The GA Improved-ITA agents first utilize Genetic-Based Machine Learning (GBML) to identify their opponent's preferable negotiation issues. It is then followed by branch and bound search to search for the best value for each of the issues. The performance of GA Improved-ITA is promising when it is compared with the results of one-to-many negotiations obtained by Bayesian learning model and heuristic search algorithm.

Key words: E-commerce, one-to-many negotiation, intelligent agents, machine learning approach, genetic-based machine learning

INTRODUCTION

The immergence of Internet and WWW revolutionizes the conduct of business and commerce. Internet links thousands of organizations worldwide into a single network and creates a vast global electronic market place. Through computers and networks, buyers and sellers can complete purchase and sale transactions digitally regardless of their location. Both suppliers and consumers are benefited as transactions such as establishing price, paying bills and ordering goods can be accomplished through the network with lower cost. These phenomenon has brought to the concept of electronic commerce (e-commerce) in which the process of buying and selling goods and services electronically, involving transactions using the Internet, networks and other digital technologies (Laudon and Laudon, 2002).

Based on the nature of the participants in business transaction, e-commerce can be categorized as business-to-consumer, business-to-business and consumer-to-consumer. Each category of the e-commerce involves buying and selling. Media Lab's Consumer Buying Behavior (CBB) model for e-commerce (Moukas *et al*., 2000) has defined buying process into six stages, namely need identification, product brokering, merchant brokering, negotiation, purchase and delivery as well as product service and evaluation. Negotiation is the key component of e-commerce (Sandholm, 1999) as business deals are often made through negotiation. Negotiation is a process in which two or more parties with different criteria, constraints and preferences, jointly reach an agreement on the terms of a transaction (Rahwan *et al*., 2002). Generally, a negotiation involves one or more potential business partners while each of which has different business goals. These potential business partners exchange their goals in the form of offers and counter offers to see if they can agree to mutually acceptable terms of a transaction. Human-based negotiations are lengthy, error prone and costly as it attempts to settle various terms in a transaction for all parties, while the parties may have opposite goals. If some parties do not concede, the negotiation may take forever to reach consensus. The autonomous, social ability, reactivitiness and pro-activeness nature of software agents make them suitable to take over human's role in negotiation. Software agents support and provide automation including the decision making to the negotiation stage in online trading. Among the literature, prominent negotiation software agents that have been implemented are the study by Jonker *et al.* (2007), Sim (2007) and Manisterski *et al.* (2008). Nevertheless, these negotiation agents support one-to-one negotiation. To support fully autonomous multi-attribute one-to-many negotiation, Intelligent Trading Agency (ITA) (Rahwan *et al*., 2002) practices bilateral one-to-many negotiation by means of conducting a number of coordinated simultaneous one-to-one multi-attribute negotiations. This approach has many advantages over existing negotiation systems in terms of customizability, scalability, reusability and robustness as one party can

Corresponding Author: M.N. Sulaiman, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia Tel: +60389466507

negotiate with several parties concurrently and finally deal with the one that can provide the best offer. Nonetheless, for most of the automated negotiation systems, agents do not obtain quality negotiations. Since, the strategies are normally programmed prior to the start of a negotiation, the decision of agents to select the best course of action do not take dynamics of negotiation into consideration. A buyer or seller may change his decision during a negotiation due to environmental factors or individual basis. The self-interested nature of the agents makes them spending more time to search for feasible solutions while the final outcomes obtained are normally sub-optimal. Thus, a hypothesis can be made: in the dynamic electronic marketplace environment, agents with the ability to learn the opponent's sophisticated preferences will produce more optimal outcomes. The objective of this study is to implement machine learning approach that allows agents to reuse their negotiation experience to improve the final outcomes in a one-to-many negotiation. The proposed agents are able to learn opponent's preferences and constraints during the negotiation. Other significant preliminary research on intelligent negotiation agents are Choi *et al.* (2001), Lau (2005), Lin *et al.* (2006), Praca *et al.* (2008) and Hindriks and Tykhonov (2008). In this study, the proposed method combines Genetic Algorithms (GAs) (Goldberg, 1989) and constraint satisfaction approach (Yokoo and Hirayama, 2000; Rahwan *et al.*, 2002) to optimize the negotiation outcomes of one-to-many negotiation in terms of negotiation time and joint utility.

GA IMPROVED-ITA

The GA Improved-ITA is made up of a group of intelligent agents that learn the negotiation by capturing opponent agent's preferences during the negotiation. These learning agents first use GAs as the learning method to identify the significant variables that dominate the negotiation. These intelligent agents then implement branch and bound search to generate the best offer during the negotiation process.

System operation: The GA Improved-ITA's framework is adapted from ITA one-to-many negotiation framework (Rahwan *et al.*, 2002) and with the addition of agent's learning capability to optimize the negotiation outcomes. The agent's learning mechanism is made up of two levels: variable identification and value estimation level. At the variable identification level, the agents recognize opponent's preferable negotiation issues. After having this information, the agents then estimate the best value for each variable in order to generate a feasible offer that is beneficial to both parties. Negotiation can be viewed as a sequential decision making process while learning of the agents occurs whenever the agents receive an offer from their opponents. Figure 1 shows the system operation of GA Improved-ITA. A buyer agent starts a negotiation by initializing a coordinating agent and a set of sub-negotiating agents or sub-buyers, bargaining concurrently with several seller agents. Each of the seller agents has his own database of the product sold. The sub-negotiating agents start a negotiation by sending the buyer's requirements to the seller agents. After receiving the requirements from the sub-negotiating agents, the seller agents search through their database to find packages that match the requirements. There are two types of requirements: non-negotiable and negotiable requirements. The non-negotiable requirements have to be fulfilled first before the agents can proceed to the negotiable requirements. The seller agents then return the

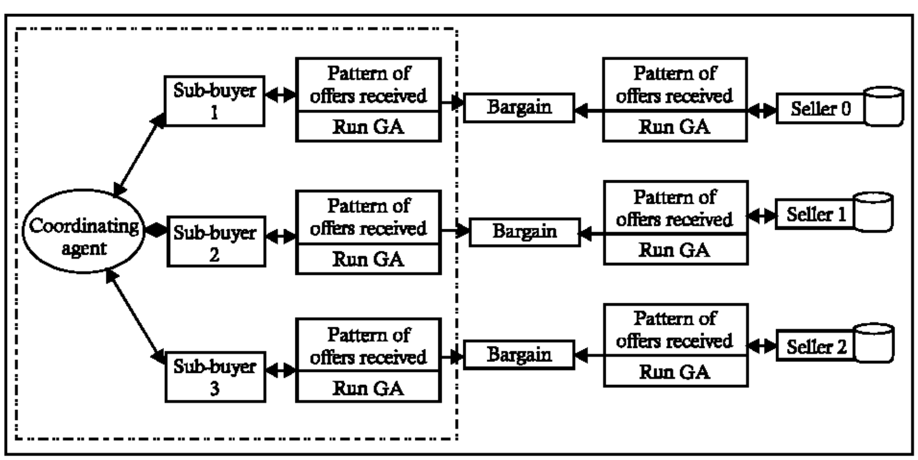

Fig. 1: System operation of GA Improved-ITA

best package of their own interest as an offer to each of the sub-negotiating agents. Mean while, the sub-negotiating and seller agents start running GAs by initializing a population of negotiation issues.

The learning mechanism is activated whenever an agent receives an offer from its opponent. For each generation in the learning process, the selection, crossover and mutation are applied to the population of negotiation issues. When an agent receives an offer from the opponent agent, it evaluates the offer. At the same time, the agent runs the learning process to capture the opponent's preferable issues by recognizing the pattern of the offer received. After that, the agent generates a counter offer to its opponent based on its utility information and the information obtained from the learning process. This process continues until an agreement is reached by one of the sub-negotiating agents with a seller agent or an agent of one side has made decision to terminate the negotiation.

System architecture: Learning agents are agents that take part in a direct negotiation with opponents. They are sub-negotiating and seller agents in GA Improved-ITA. Figure 2 shows the architecture of an individual learning agent in GA Improved-ITA. The individual learning agent's main architecture consists of a detector, the proposed GAs learning environment (GBMLE), a negotiation engine and an effector. The negotiation engine is equipped with knowledge of a set of constraint variables with different levels of satisfactions. It mainly performs two functions: evaluating offers and generating new offers.

The detector detects two types of data from the negotiation environment: (1) the counter of the offers received and (2) the value of each negotiation issue from the opponent's offers. The counter is incremented by one each time when the learning agent receives an offer from its opponent agent. The message passed within the offer from opponent agent is perceived as raw data. The detector then extracts the value of each negotiation issue from the offer received. Combining these two types of data yields the opponent's preference learning data.

The GBMLE constructs a preference learning model based on opponent's preference learning data. This model consists of chromosomes that represent different combinations of negotiation issues. The function of GBMLE is to predict the opponent agent's preferable negotiation issues. Next, this information coupled with the user's utility information, the negotiation engine then generates a counter offer and sends it to the opponent agent through the effector. The value of negotiation

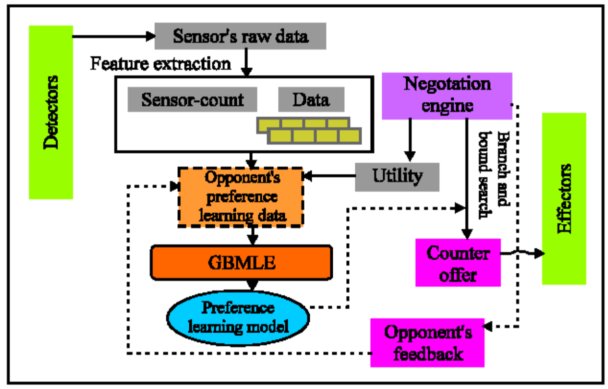

Fig. 2: GA Improved-ITA individual agent's architecture

issues in the next offer received by the learning agent provide the feedback indicating the type of negotiation issues the opponent agent would prefer. The detector appends this feedback of information to the previous opponent's preference learning data in order to update the preference learning model.

Offer evaluation and generation: The two important tasks of a negotiation agent are evaluating offers received and generating new offers. Constraints satisfaction techniques (Yokoo and Hirayama, 2000) and GAs approach are applied in GA Improved-ITA to carry out these tasks. Evaluating an offer is to reason the acceptability of an offer sent by the opponent agent. The GA Improved-ITA agents use the utility theory and constraint propagation techniques (Kowalczyk and Bui, 2000) to evaluate an offer. Prior to a negotiation, both sides of agents, i.e., buyer and seller represent their preferences and constraints in a utility function. To specify more complex preferences, there are different levels of satisfaction for the values within the domain of a single variable. Figure 3 shows the activity diagram for evaluating offers and generating new offers of GA Improved-ITA sub-negotiating and seller agents. When an agent receives an offer from its opponent, the proposed value of each negotiation issue must belong to its domain as specified by the agent. It is noted that the offer consists of a number of variables $X = \{x_1,\ldots,x_n\}$, while there is a value for each variable in X. The agent then finds the utilities of these variables by referring to its utility function. An offer is acceptable if it at least satisfies the minimum utility of both parties. If both agents' areas of acceptability overlap, a solution can be found within the zone of agreement (Zeng and Sycara, 1998), otherwise, the negotiation fails.

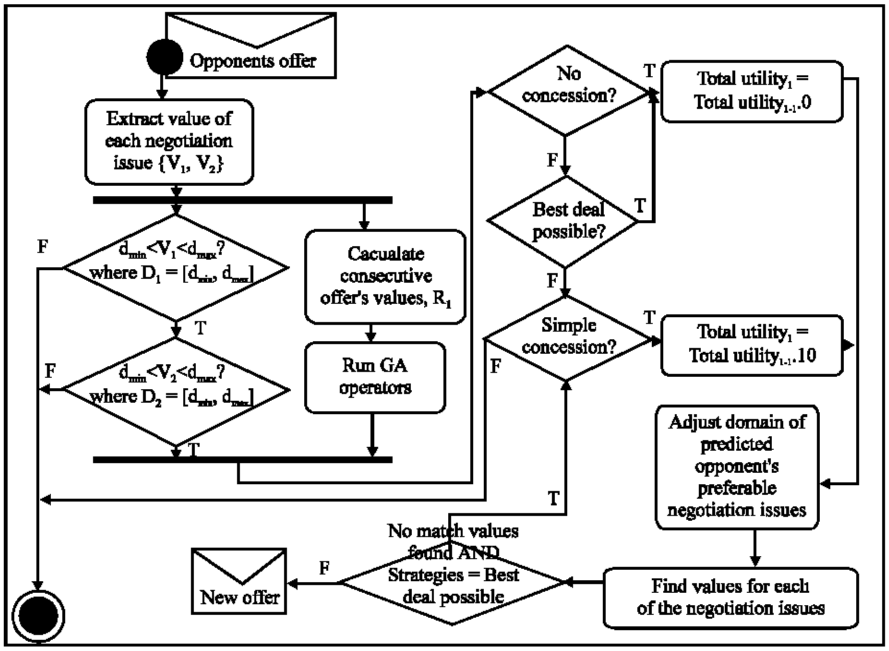

Fig. 3: GA Improved-ITA evaluating offers and generating new offers activity diagram

Generating a new offer means searching for prospective solutions that lie within the zone of agreement. The GA Improved-ITA agents identify the set of significant variables first before they can assign values to each variable in the new offer. Hence, the agents can look for better values for the significant variables by the way maintaining the values of the less important variables when generating a new offer. This method saves not only the searching time, but also the agent's effort in finding better solutions. The task of identifying significant variables is carried out in the GBMLE. If the zone of agreement exists between both agents after the evaluation state, the values of the variables from opponent agent's offer are passed to the GBMLE as opponent's preference learning data. A preference learning model is then constructed. This model helps GA Improved-ITA agents to predict the preferable negotiation issues of opponent agents. This set of negotiation issues is the outcomes of the learning mechanism. To generate a new offer, an agent first calculates the total utility of the offer received by summing the utilities of the variables in the offer from opponent agents. It is noted that the strategies determine the amount of concession the party is willing to make during an episode of negotiation. Based on the negotiation strategies that the agent is bound to, the agent makes more concession on the values of the significant variables. The constraint consistency maintenance and branch and bound search are then utilized to search for better values of the offer.

NEGOTIATION LEARNING MODEL IN GA IMPROVED-ITA

The negotiation learning method that employs GAs driven implementation of rule-based machine learning (GBML) (Sette *et al*., 2004) is proposed. The goal of the proposed negotiation learning method is to generate a set of rules for recognizing opponent's preferences by exploiting the GAs operators. Learning Classifier System (LCS) (Goldberg, 1989) is a machine learning system that applies genetic learning to rules in a production system. XCS (Wilson, 1995) was then resembles traditional LCS but differs in several aspects. In XCS, each classifier maintains a prediction of expected payoff but the classifier's fitness is given by a measure of the prediction's accuracy. The proposed agents use XCS to generate the rules for identifying the set of negotiation issues that dominate the negotiation. This negotiation learning model is called NEXCS.

The NEXCS resides in GBMLE for an individual negotiation agent. Outside GBMLE, an environment that represents the negotiation problem is created. This environment mainly consists of three components. They are the state of problem, execution of action and the payoff function for the action executed. The state of problem is the offer received from the opponent agent. It is encoded to binary string representation. Each of the genes within the string represents the changes of consecutive offer's values of a negotiation issue. The execution of action means the running of classifier's action to the environment. In this context, the action denotes the set of significant negotiation issues. The payoff function determines the reward of the predicted and actual action. The scheme defined is to give maximum reward, 1,000 to a correct prediction and a minimum reward, 0 to an incorrect prediction based on the feedback from the environment, which refers to the next offer received by the agent.

The environment interacts with NEXCS by sending the current state of problem to NEXCS as input message and providing a reward back to NEXCS in response to the predicted action from NEXCS. The regime of learning opponent agent's preferences by an agent engages in explore/exploit pairs (Wilson, 1995). In the exploration mode, NEXCS attempts to discover hypotheses which are the possible opponent agent's preferable issues. The NEXCS chooses negotiation issues randomly from a set of negotiation issues (action set). Learning occurs in exploration mode. After the exploration, the environment returns the payoff in response to the action to NEXCS. Exploitation determines how well NEXCS is doing to obtain the right prediction based on the payoff from environment. Thus, NEXCS seeks to predict the opponent agent's preferable issues by choosing the set of negotiation issues with the highest prediction from the action set in the exploitation mode.

The process of NEXCS starts with a population of classifiers, which denotes the rules for identifying opponent agent's preferable issues. Each classifier, as defined in XCS, consists of a condition, the resultant action, numerosity and various predictive accuracy measures, such as p, ε and F. Numerosity stands for the number of multiple occurrences of a classifier. p is an estimate of the proportion of examples that the classifier classifies correctly. ε is calculated from the absolute difference between the predicted and the actual reward that a classifier receives. F is the accuracy of a classifier relative to those classifiers with the same action that are active at the same time.

The population is empty at the start of the run. When the detectors receive the current state of problem from environment, NEXCS compares the state of problem with the classifiers in population, P. The classifiers with the condition that matches the state are put into match set (M). Covering (Wilson, 1995) occurs if the state does not match the condition of any classifiers or when the population is empty. Within the match set, NEXCS checks to see if there is an existing classifier with the same condition and action with the new created classifier. If it happens, the new classifier is discarded and the existing classifier's numerosity is incremented by one. The numerosity value affects a classifier's fitness that constitutes the probability of a classifier to be selected for reproduction in genetic algorithms.

```
Procedure doSingleStepProblemExplore;
   begin
         generate match-set from population;

         //execute covering
         if no classifier in population matches the problem state
               create new classifier in population;

         for each classifier in match-set
               calculate predicted fitness;

         end for

         // action denotes set of opponent preferable negotiation
         issues
         generate action-set from match-set;
         genarate a random action;
         execute a random action;
         get reward from environment;
         update fitness of actions in action-set;

         if counter of received offer is greater than θga
               run GA operations;
         end if
   end
```

Fig. 4: Exploration mode algorithms in NEXCS

Figure 4 shows the algorithm of exploration mode in the GA Improved-ITA negotiation learning process. Adapted from XCS, NEXCS forms a system prediction $P(a_i)$ for each action a_i in M. $P(a_i)$ is computed using a fitness-weighted average of the prediction of classifier advocating a_i. $P(a_i)$ for each a_i is stored in a prediction array. NEXCS then selects an action from the prediction array and forms an action set (A) of classifiers. In exploration mode, NEXCS selects the negotiation issues randomly. The selected negotiation issues are wrapped as the argument of action from NEXCS and then posted to the negotiation engine, which is equipped with the utility information of an agent. Prescribed by the concession rules being used (Kowalczyk and Bui, 2000), the agent generates a new offer by constraint satisfaction approach by conceding more on the values of the negotiation issues predicted by NEXCS.

Figure 5 shows the algorithm of exploitation mode in the negotiation learning process. When the agent receives the counter offer from its opponent agent, the counter offer becomes payoff information from environment to NEXCS. The payoff function defines the reward to the earlier predicted opponent's preferable negotiation issues and actual counter offer received by the negotiation agent. This information is used to update the classifier's attributes in terms of its p, ε and F-values. In the next cycle of negotiation, NEXCS carries out the

```
Procedure doSingleStepProblemExploit;
    begin
            generate match-set from population;

            //execute covering
            if no classifier in population matches the problem state
                    create new classifier in population;

            for each classifier in match-set
                    calculate predicted fitness;

            end for

            // action denotes set of opponent preferable negotiation
            issues
            generate action-set from match-set;
            genarate best action with the heghest prediction array;
            execute a best action;
            get reward from environment;

            if prediction is correct
                    set correct [counter] = 1;
            else
                    set correct [counter] = 0;
            else if

            set systemError [counter] = reward-best prediction array;

    end
```

Fig. 5: Exploitation mode algorithm in NEXCS

prediction by selecting the negotiation issues based on the values of p, ε and F of classifiers within A. Thus, the action with the highest prediction is selected. The attempt of testing the agent's learning performance is conducted in exploitation mode.

The explore/exploit regime is carried out iteratively in every cycle of negotiation as reinforcement learning. In the exploration mode, GAs may occur among classifiers in A as rule discovery operation. Classifiers with high fitness values are chosen to undergo genetic operation: crossover and mutation. It is noted that if the population size L exceeds the limit N, classifiers are deleted to return L to N. In the exploitation mode, the agent tests its learning performance without running the GAs in NEXCS. The explore/exploit regime is conducted in every cycle of negotiation between each pair of negotiation agents in GA Improved-ITA until an agreement is reached or one of the parties terminates the negotiation.

EXPERIMENTAL RESULTS

A computer trading scenario is used in the experiments. The trade starts when the buyer agent sends out a request for computer package to all participating seller agents, simultaneously. The seller agents search for available packages in their database and, in return, generate an offer to the buyer agent. The negotiation involves four issues: CPU name, CPU speed, hard disk manufacturer and total price of hard disk and CPU. The non-negotiable attributes are CPU name and hard disk manufacturer while the total price of hard disk and CPU and CPU speed are negotiable attributes. The buyer and seller agents' intention is to get the best deal possible from their opponent. A negotiation cycle consists of one exchange of offers and counter offers by each pair of negotiation agents. The composite buyer agent consists of three instances of sub-buyers and a coordinating agent, implemented as a multi-threaded system. GA Improved-ITA uses the XCSJava1.0 library (Butz, 2000) to implement GBML.

To evaluate the performance of GA Improved-ITA, two major experiments are conducted. The first experiment evaluates the performance of NEXCS in negotiation problems. The second experiment examines the performance of the learning agents in GA Improved-ITA by observing the final negotiation outcomes. Results obtained were compared with negotiation outcomes of ITA agents and the negotiation agents that utilizes Bayesian learning model (Ng *et al*., 2009) in terms of the negotiation payoff, the joint utility of negotiation agents and the negotiation cycle.

Performance of machine learning approach for learning negotiation: Figure 6 shows the performance graph of the proposed negotiation learning model (NEXCS) in randomly generated negotiation problems. The performance of NEXCS is evaluated by using JXCS developed by the Faculty of Computer Studies and Mathematics of University of the West England, United Kingdom. There are four kinds of data in the graph: prediction accuracy, system error, population of classifiers (chromosomes) and optimality. The accuracy curve (blue) shows the model's prediction performance after 5,000 iterations. The model's accuracy of predicting the opponent's preferable negotiation issues correctly remains around 89% level. The corresponding error curve (red) is around the 2% level over the 5,000 iterations.

The light blue curve represents the percentage of the number of classifiers contained within the classifier population. Optimality shows the percentage of the optimal set of classifiers presented in the population on any given sampled iteration. The optimality is approaching 100% level after the 5,000 iterations.

Comparison of negotiation payoff: The main claim of this study is to develop negotiation agents that have the learning ability to improve the negotiation outcomes in a one-to-many negotiation. The outcomes are measured in

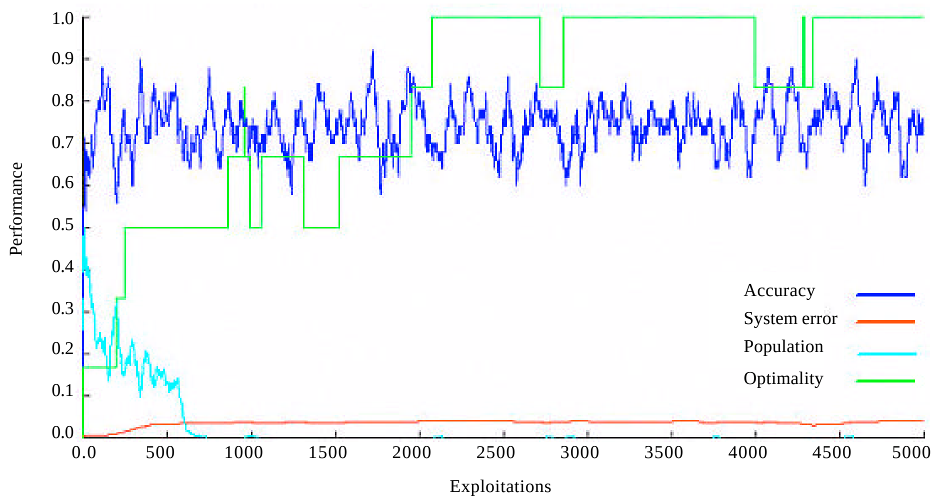

Fig. 6: Performance graph of NEXCS

terms of negotiation payoff of both buyer and seller agents, the number of negotiation cycle and the joint utility between buyer and seller agents. To evaluate the effectiveness of learning in one-to-many negotiation, the negotiation outcomes of GA Improved-ITA are compared with the negotiation outcomes of ITA and Bayes Improved-ITA (Ng *et al*., 2009).

Figure 7 shows the negotiation payoff of ITA, GA Improved-ITA and Bayes Improved-ITA (Ng *et al*., 2009). In terms of buyer, ITA and GA Improved-ITA achieve the highest total utility. They gain the same value for the CPU speed. Bayes Improved-ITA agents have the lowest total utility. They gain a better price by sacrificing the speed value. Although, the total utility values of the GA Improved-ITA and ITA are identical, the GA Improved-ITA agents can achieve better price from opponent at the end of the negotiation.

In a negotiation, the joint utility indicates the quality of a particular negotiation process (Zeng and Sycara, 1998). Table 1 shows the joint utility of ITA, GA Improved-ITA and Bayes Improved-ITA (Ng *et al*., 2009). It can be observed that GA Improved-ITA has the highest joint utility value. It indicates that GA Improved-ITA achieved more optimal outcomes in comparison with ITA and Bayes Improved-ITA (Ng *et al*., 2009). Thus, GA Improved-ITA has presented better quality of negotiation than ITA and Bayes Improved-ITA (Ng *et al*., 2009).

Comparison of negotiation cost: Figure 8 shows the negotiation cost of ITA, GA Improved-ITA and Bayes Improved-ITA (Ng *et al*., 2009). GA Improved-ITA spends the least negotiation cycle to achieve the agreement. It is 28.6% less than ITA. Although, Bayes Improved-ITA behaved consistently throughout the negotiation (Ng *et al*., 2009), the Bayesian learning approach needs more time to learn the negotiation in comparison with the GAs implemented in GA Improved-ITA.

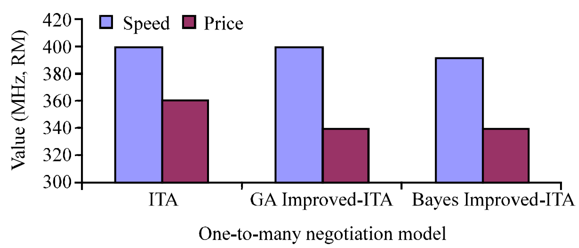

Fig. 7: Comparison of negotiation payoff of ITA, GA Improved-ITA and Bayes Improved-ITA

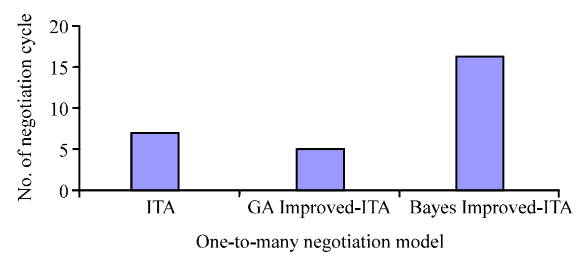

Fig. 8: Comparison of negotiation cost of ITA, GA Improved-ITA and Baye Improved-ITA

Table 1: Joint utility of ITA, GA Improved-ITA and Bayes Improved-ITA

One-to-many negotiation model	Joint utility
ITA	0.08
Bayes Improved-ITA	0.18
GA Improved-ITA	0.19

DISCUSSION

The proposed negotiation learning model, namely NEXCS is tested with simulated negotiation problems in explore/exploit regime in JXCS (Geenyer, 2000) over 5,000 iterations. The NEXCS predicts the preferences and constraints of opponent agent during a negotiation by modelling the rules for identifying opponent agent's preferable issues as a set of classifiers. The accuracy value obtained in the first experiment shows that it is promising to apply machine learning approach for learning multi-attribute negotiation. From the data obtained in the first experiment, it can be observed that the number of classifiers decreases over time although the accuracy remains level. The model is initialized with 27 rules at the beginning of the run. The number of classifiers finally settles to 10 rules which are significant in the prediction of the opponent's preferable negotiation issues. The achievement is due to discarding of less fit classifiers over the genetic operations in GBMLE. At the end of the run, only the set of fittest classifiers remains in the classifier population. Compare to other related research by Choi *et al.* (2001) that employs GAs to generate multiple potential offers by manipulating the value of negotiation issues, GA Improved-ITA saves extensive negotiation time by narrowing the search space of the agent when generating a new offer. The first experiment study has proven that machine learning approach is effective in learning multi-attribute negotiations. The main claim of the study is to optimize automated negotiations in terms of the negotiation payoff, joint utility of negotiation agents and the negotiation cost. Due to the promising performance of machine learning approach to learn multi-attribute negotiations in the first experiment, the proposed negotiation learning model (NEXCS) was then implemented in agent system that yields GA Improved-ITA to improve the negotiation outcomes in one-to-many negotiations. The second experiment investigates the negotiation outcomes of GA Improved-ITA in multi-attribute one-to-many negotiation. It can be observed that GA Improved-ITA outperformed ITA and Bayes Improved-ITA (Ng *et al.*, 2009) to achieve maximum payoff. GA Improved-ITA obtained the negotiation payoff that approximated the user's utility information. It is due to the ability of the negotiation agents in GA Improved-ITA to learn the user's utility structure so that it can identify the significant negotiation issues that are preferable by the user of the negotiation agent system. Besides obtaining the maximum payoff for the negotiation agent itself, achieving a win-win situation is always the ideal in negotiation. However, most of the negotiations end up with sub-optimal outcomes (Bosse and Jonker, 2005) as the parties involved focus on money and have ignored other terms that might be more important in a transaction. From the results of the joint utility values which measures the social welfare of buyer and seller agents (Cheng *et al.*, 2005), it can be observed that agents with the learning ability can achieve more optimal outcomes than the non-learning agents. Once the negotiation agent has the knowledge of the utility information of its opponent, the agent can combine this information to its utility function and decide on the most feasible solution to both sides of the parties during negotiations. The ITA obtained sub-optimal outcomes as the agents behaved self-interested that the welfare of its opponent was ignored during the negotiation. The significance of the results implies that learning opponent's preferences during the negotiation can optimize the negotiation payoff of both sides of the negotiation parties. On the other hand, optimizing negotiation outcomes includes minimizing the negotiation cost that is measured by the time spent on negotiations. GA Improved-ITA outperformed ITA and Bayes Improved-ITA (Ng *et al.*, 2009) to reach agreement in a negotiation with fewer cycles. It is due to GA Improved-ITA agent's learning mechanism to identify significant negotiation issues first while retaining the values of the rest of the issues when generating offers to opponents. This method can obviously save agent's effort to search for feasible solutions in comparison with Bayes Improved-ITA agents (Ng *et al.*, 2009) that take every negotiation issue as proportional each time an agent updates its belief. The experiment work showed that GA-Improved ITA agents have successfully served the objective of this study to optimize multi attribute one-to-many negotiation by maximizing the negotiation payoff and joint utility of both sides of the negotiation parties while minimizing the negotiation cost.

CONCLUSIONS

This study attempts to implement machine learning approaches that allow negotiation agents to reuse their experience to improve the final outcomes of one-to-many negotiation. An enhanced version of ITA, namely GA Improved-ITA one-to-many negotiation model was developed. Negotiation agents in GA Improved-ITAs are equipped with learning capability that can learn the preferences and constraints, which is the private information, of their opponents. In GA Improved-ITA, negotiation agents first identify their opponent's preferable negotiation issues from offers received by executing NEXCS. The agents then apply branch and bound search method to search for best values for each of the negotiation issues. By having the information of the predicted opponent's preferable negotiation issues, the agents can focus on the identified set of issues during while generating a new offer. The proposed negotiation agents successfully improve the negotiation outcomes by reducing the negotiation cost while optimizing the benefits of all parties involved in negotiation. The significance of the results from the experimental study has proven that the adaptive nature of agents can increase the fitness of these autonomous agents in the dynamic electronic market rather than solely practicing the sophisticated negotiation strategies for the negotiation agents. Besides, it is found out that promising negotiation outcomes are achieved when the opponent's preference and constraints are taken into consideration during the negotiation rather than being self-interested. As future study, the proposed GMBLE will be tested in negotiations that involve more and 20 issues to observe the performance of the learning model.

ACKNOWLEDGMENT

The authors wish to thank Dr. Iyad Rahwan for providing helpful information and data sets of ITA.

REFERENCES

Bosse, T. and C.M. Jonker, 2005. Human vs computer behavior in multi-issue negotiation. Proceedings of the Rational, Robust and Secure Negotiation Mechanisms in Multi-Agent Systems, Jul. 25, Amsterdam, Netherland, pp: 11-24.

Butz, M.V., 2000. XCSJava 1.0: An implementation of the XCS classifier system in Java. IlliGAL (Technical Report No. 2000027). Illinois Genetic Algorithm Laboratory, Department of General Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, USA.

Cheng, C.B., C.C.H. Chan and C.C. Lin, 2005. Buyer-supplier negotiation by fuzzy logic based agents. Proceedings of the Information Technology and Applications, ICITA 2005. Third International Conference, Jul. 4-7, Taichung, Taiwan, pp: 137-142.

Choi, S.P.M., J. Liu and S.P. Chan, 2001. Evolutionary Negotiation in Agent-Mediated Commerce. In: Active Media Technology, Liu, J. et al. (Eds.). Springer-Verlag, Berlin, Heidelberg, pp: 224-234.

Geenyer, A., 2000. The Use of Learning Classifier Systems JXCS. In: CoIL Challenge 2000: The Insurance Company Case (Technical Report No. 2000-09), Van der Putten, P., M. Van Someren (Eds.). Leiden Institute of Advanced Computer Science, Leiden, Netherlands.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. 1st Edn., Addison-Wesley, New York, USA., ISBN: 0201157675.

Hindriks, K. and D. Tykhonov, 2008. Opponent modeling in automated multi-issue negotiation using Bayesian learning. Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, May 12-16, Estoril, Portugal, pp: 331-338.

Jonker, C., V. Robu and J. Treur, 2007. An agent architecture for multi-attribute negotiation using incomplete preference information. Autonomous Agents Multi-Agent Syst., 15: 221-252.

Kowalczyk, R. and V. Bui, 2000. On Constraint-Based Reasoning in E-Negotiation Agents. In: Agent-Mediated Electronic Commerce III, Dignum, F. and U. Cortés (Eds.). Springer-Verlag, Berlin, Heidelberg, pp: 31-46.

Lau, R.Y., 2005. Adaptive negotiation agents for e-business. Proceedings of the 7th International Conference on Electronic Commerce, Aug. 15-17, New York, USA., pp: 271-278.

Laudon, K.C. and J.P. Laudon, 2002. Management Information System: Managing the Digital Firm. 7th Edn., Prentice-Hall, Inc., New Jersey, ISBN: 0130330663, pp: 23.

Lin, R., S. Kraus, J. Wilkenfeld and J. Barry, 2006. An automated agent for bilateral negotiation with bounded rational agents with incomplete information. http://www.umiacs.umd.edu/~sarit/Articles/linetalEcai06.pdf.

Manisterski, E., R. Lin and S. Kraus, 2008. Understanding how people design trading agents over time. Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, May 12-16, Estoril, Portugal, pp: 1593-1596.

Moukas, A., G. Zacharia, R. Guttman and P. Maes, 2000. Agent-mediated electronic commerce: An MIT media laboratory perspective. Int. J. Electron. Commerce, 4: 5-21.

Ng, S.C., M.N. Sulaiman and M.H. Selamat, 2009. Intelligent negotiation agents in electronic commerce applications. J. Artif. Intell., 2: 1-11.

Praca, I., M.J. Viamonte, Z. Vale and C. Ramos, 2008. Agent-based simulation of electronic market places with decision support. Proceedings of the 2008 ACM Symposium on Applied Computing, Mar. 16-20, Fortaleza, Ceara, Brazil, pp: 3-7.

Rahwan, I., R. Kowalczyk and H.H. Pham, 2002. Intelligent agents for automated one-to-many e-commerce negotiation. Proceedings of the 25th Australasian Conference on Computer Science, 2002, Melbourne, Victoria, Australia, pp: 197-204.

Sandholm, T., 1999. Automated negotiation. Commun. ACM, 42: 84-85.

Sette, S., B. Wyns and L. Boullart, 2004. Comparing learning classifier systems and genetic programming: A case study. Eng. Appl. Artif. Intell., 17: 199-204.

Sim, K.M., 2007. Relaxed-criteria G-negotiation for grid resource co-allocation. SIGecom Exch., 6: 37-46.

Wilson, S.W., 1995. Classifier fitness based on accuracy. Evol. Comput., 3: 149-175.

Yokoo, M. and K. Hirayama, 2000. Algorithms for distributed constraint satisfaction: A review. Autonomous Agents Multi-Agent Syst., 3: 185-207.

Zeng, D. and K. Sycara, 1998. Bayesian learning in negotiation. Int. J. Hum. Comput. Syst., 48: 125-141.