Automated Negotiation And Behavior Prediction

Mohammad Irfan Bala, Sheetal Vij
Department of Computer Engineering
Maharashtra Institute of Technology, Pune

Abstract

Negotiation is a joint decision making process between two or more parties with limited common knowledge and conflicting preferences. With the proliferation of web technologies it has becomes increasingly important to make the traditional negotiation pricing mechanism automated and intelligent. In electronic commerce, the task of negotiation can be delegated to a software agent in order to save time of human users on activities which are either demanding or repetitive. Various tactics have been given till date which determines the behavior of the software agents in the negotiation process. Negotiation is although very complex activity to automate without human intervention but the software agents when enhanced with learning techniques can better simulate the human intelligence and increase the profits of their owners. Prediction of partners behavior in negotiation has been an active research direction in recent years as it will not only improve the utility gain for the adaptive negotiation agent and also achieve the agreement much quicker. The basic concept is that the information about negotiators, their individual actions and dynamics can be used by software agents which are equipped with adaptive capabilities so that they can learn from past negotiations and provide assistance for selection of appropriate negotiation tactics.

1. Introduction

Negotiation is a form of interaction in which a group of agents with conflicting interests and a desire to cooperate try to come to a mutually acceptable agreement on the division of scarce resources. These resources may not only refer to money but can also include other parameters like product quality features, warranty period, way of payment, etc. Electronic negotiations have gained heightened importance due to the advance of the web and e-commerce. The tremendous success of online auctions clearly indicates that e-negotiation will gradually become the core of e-commerce. Whether it is a case of B to B purchase or a case of online shopping, it is important to make the traditional negotiation mechanism automated and intelligent. The automation saves human negotiation time and computational agents are better at finding deals in strategically complex settings.

Traditionally negotiation processes have been carried out by humans registering at certain web pages, placing bids, making offers and receiving counter offers from other participants. One major disadvantage with this way of e-negotiation is that the knowledge and experience is kept within the human minds. Agent mediated negotiations have received considerable attention in the field of automated trading. However various problems are faced by the negotiation agents such as limited and uncertain knowledge and conflicting preferences. Also agents may have inconsistent deadline and partial overlaps of zones of acceptance. Moreover, multilateral negotiations are more complicated and time consuming than bilateral negotiations. These factors make it difficult to reach consensus. The agent should be able to learn and adapt according to the behavior of the opponent in order to be successful.

The need is that the agents should be equipped with a decision making mechanism which allows them to adapt to the behavior of the negotiation partner. Intelligent systems for negotiation aim at increasing the negotiators abilities to understand the opponent's needs and limitations. This ability helps to predict the opponent's moves which can be a valuable tool to be used in negotiation tasks. Various approaches have been proposed which are capable of predicting the opponent's behavior. The research presented here focuses on the online prediction of the other agent's tactic in order to reach better deals in negotiation. While the extensive coverage of all the prediction methods employed in negotiation is beyond the scope of the current work, it is useful to mention several key studies.

2. Related work

Predicting the agent's behavior and using those prediction results to maximize agents own benefits is one of the crucial issues in the negotiation process. It is necessary for an agent to produce offers based on his own criteria because an agent has limited computational power and incomplete knowledge about opponents. Various approaches [1,2,10,15,16,18] have been proposed in the past for predicting the opponent's negotiation behavior. We reviewed some of the approaches to come up with certain conclusions regarding the efficiency of each approach and their short comings.

Initially game theory was used in the negotiation process. It treats negotiation as a game and the negotiation agents are treated as players of the game. Zeng and Sycara [9] used game theoretic approach with Bayesian belief revision to model a negotiation process. However game theory has two main drawbacks which make it unsuitable for use in the negotiation process. First is that it assumes the agent has infinite computational power and secondly it assumes all the agents have common knowledge. These limitations of the game theory were overcome by the decision functions.

Faratin [5] proposed a bilateral negotiation model in which the two parties negotiate on an issue like price, delivery time, quality etc. The two parties adopt opposite roles (buyer and seller) and use one of the three families of negotiation tactics namely: Time dependant tactics, Resource dependant tactics and behavior dependant tactics. The offers exchanged between the agents are represented as $X^{t}_{a\to b}$. This is the offer generated by agent 'a' for agent 'b' at time 't'. All the offers are restricted in between min_a and max_a which specifies the range of all possible offers of 'a'. Each agent has a scoring function V_a which assigns a score to each offer produced. A sequence of alternating offers and counter offers by the agents is called negotiation thread. An agent may respond to the offer by any of the three ways: withdraw, accept or offer

$$R^a(t^n, x^{t_{n-1}}_{b\to a}) = \begin{cases} withdraw(a,b) & if t^n > t^a_{max} \\ accept(a,b,x^{t_{n-1}}_{b\to a}) & if V^a(x^{t_{n-1}}_{b\to a}) > V^a(x^{t_n}_{a\to b}) \\ offer(a,b,x^{t_n}_{a\to b} & otherwise \end{cases}$$

$X^{t_n}_{a\to b}$ is the counter offer generated by agent 'a' in response to the offer $X^{t_{n-1}}_{b\to a}$ of agent 'b'. t^a_{max} is the deadline for agent 'a' by which the negotiation should be complete.

Chongming Hou [1] proposed to use non linear regression approach for the prediction of the opponent's tactics. It could predict the approximate value of opponent's deadline and reservation values. The performance of the agent improved by using this approach as it reduced the number of negotiation breakdowns and caused early termination of unprofitable negotiations. But this approach is restricted for bilateral negotiations only and can be used only when the agent is sure that the opponent is using one of the above mentioned families of tactics for negotiation.

E-negotiation can be classified into three types: one-to-one negotiation, one-to-many negotiation and many-to-many negotiation. Hsin Rau, Wei-Jung Shiang, Chao-Wen Chen and Chiuhsiang Joe Lin [6] focused on one-to-many negotiation architecture and integrated two commonly used coordination strategies i.e. Desperate Strategy and Patient Strategy, to develop a new coordination strategy. In desperate strategy agents want to complete negotiation process as early as possible. Negotiation process is terminated as soon as any of the sub-buyer agents is successful in negotiation. In case several proposals are found at the same time, the proposal with highest utility gain is accepted. However in patient strategy all the sub-buyer agents are allowed to complete their negotiation process. The sub-buyer agents finishing early are made to wait for other sub-buyer agents till all complete their negotiation. After all the sub-buyer agents finish negotiation, coordination agent selects the best proposal to make the contract.

Many other prediction approaches have been proposed which are based on machine learning mechanism. Most of the work devoted to the learning approach is focused on learning from previous offers i.e. offline learning. They require training data and such agents need to be trained in advance. However this approach may not always work well for the agents whose behavior has been excluded from the training data. Also such data may not be always available. This issue was overcome by Fenghui Ren and Minjie Zhang [18] who proposed three regression functions namely linear, power and quadratic to predict agent's behavior. These regression functions use only data about historical offers in the current negotiation thread instead of the using training data which may not always be available.

Brzostowski and Kowalczyk [10] presented a way to estimate partner's behaviors by employing a classification method. They used a decision making mechanism in which the agents are allowed to mix time-dependant tactics with behavior dependant tactics using weights which can result in quite complex negotiation behavior. However this approach only works for the time dependent agent and the behavior-dependent agent, which limits its application domains. Gal and Pfeffer presented a machine learning approach based on a statistical method [14,17]. One major limitation of this approach is the difficulty to train the system perfectly. Therefore, for some unknown kind of agents whose behaviors are excluded in the training data, the prediction result may not reach the acceptable accuracy requirements.

I. Roussaki, I. Papaioannou, M. Anagnostou [13] proposed an approach based on learning technique which has been employed by Client Agents and uses a feed-forward back-propagation neural network. It consists of a single output linear neuron and three hidden layer's neurons. These neural networks require minimal computational and storage resources making it ideal for mobile agents. The agents use a fair relative tit-for-tat negotiation strategy and the results obtained were evaluated via numerous experiments under various conditions. The experiments indicated an average increase of 34% in reaching agreements. This approach has excellent performance when the acceptable interval of the negotiation issue overlaps irrespective of the concession rate. On the other hand if the acceptable intervals overlap is limited and the deadline is quite high, this approach is likely to fail.

Experimental Setup: A comprehensive negotiation model is required to clearly define the different phases of a negotiation process and to show: 1] What information and knowledge should be defined at each phase; 2] How the information and knowledge can be used by an automated negotiation system to conduct its negotiations, and 3] How the results of negotiations provide feedback to other phases of a negotiation process. We consider bilateral negotiations, i.e. negotiation between only two parties. The interaction between two parties is regulated by a negotiation protocol which defines the rules for the exchange of proposals or offers. Specifically, we will be using the alternating-offers protocol for negotiation, in which the negotiating parties exchange offers in turns. We choose this protocol due to its simplicity and more importantly because it is widely studied and used in the literature.

Now, the parties negotiate over a set of issues, where an issue can take any value from the associated range of alternatives. The outcome of negotiation consists of a mapping of every issue to a value, and the set of all possible outcomes is called the negotiation domain or outcome space or scenario. The domain knowledge is common to the negotiating parties and remains same during a single negotiation encounter. In addition to the domain, both parties have privately-known preferences which are used by the utility function to calculate the utility of each offer. Each utility function (U) maps all possible outcomes to a real-valued number in the range [0; 1]. The overall utility consists of a sum of the utility for each individual issue. While the outcome space of each domain is known to all, the utility function of each player is private information. The player does not have any information about the preferences of the opponent. However the player can attempt to learn about preferences from previous negotiation encounters. The scenario also contains a deadline with it and may also have some discount factor.

3. Programmers design

E-Negotiation design is currently more of a trial-and-error game because of lack of a coherent resource that indicates which negotiation technique is best suited to a given type of domain. Also due to wide variety of possibilities, it should be clear that universally there is not any best technique or approach for automated negotiation. The automated negotiation process is still in its infant stage, because there are still some difficulties in this field. The first is the ontology issue; the second is agent's strategies and third is Communication protocol. In short, the most prominent issues that must be addressed in a negotiation mechanism are:

- How to represent the preferences and offers of each party;
- How to compute concession and generate a counteroffer;
- How to evaluate an incoming offer;
- How to learn the opponent's preferences;

All these issues will be addressed in my project. The input will be an xml file which will contain information about the negotiation scenario and the predefined issues for that scenario. Each issue can take only some discrete values and the importance of each value are assigned by assigning weights to them. Each agent will get separate xml file as input. the objective is to find any mutually acceptable offer before the deadline. Any of the agents may decide to end the negotiation prematurely without specifying reasons which will give zero utility to both the agents.

3.1 Mathematical model

The given mathematical model is for bilateral negotiations where an agent can negotiate about multiple issues. It also supports learning from the previous negotiation rounds. The mathematical model for the proposed work is as follows:

M={ A,S,T,D,P}

A : The set of agents that will participate in the negotiation. A={*a*1, *a*2, ...*an*} represents n agents. Each agent will follow a distinct strategy during negotiation.

S : The set of scenarios in which the agents can negotiate. Each scenario will have some predefined issues associated with it. Each issue is assigned some weight to indicate its importance.

For all $S_i \in S$, $S_i = \{I_1, I_2 ... I_n\}$ where I represents the issue in the given scenario and is given by:

$$I = \{R_1W_1, R_2W_2, ... R_nW_n\}$$

R is the range of values an issue can take and W is the weight assigned to each issue.

T : The time limit for the negotiation. Whole process of negotiation should complete before the time limit.

D : The database containing the offers of previous negotiation rounds. It can be used to predict the strategy of the opponent.

P : The vector set of the forecasted values

The output may be successful or unsuccessful negotiation. The output will be some offer which is acceptable to both the agents when the negotiation is successful. Utility function will be used to evaluate the utility of each offer.

Constraints: The fact that many good algorithms, models and theories cannot be verified without a practical application platform constrains the development of automated negotiation research. Such a situation is widespread in the entire e-commerce applications related to automated trading. The proposed solution has a few more constraints; most important being that the system will not be able to negotiate in all possible situations. It will be able to negotiate only in some specific and predefined situations. Also it may not be able to identify all possible strategies of the opponent. Human intervention in not required throughout the negotiation process but they have to specify the requirements at the start of the negotiation. Also we have a time limit for the negotiation process to complete and we assume that there is no network delay which would otherwise reduce the number of offers exchanged.

3.2 Negotiation algorithm

The description of the proposed bilateral negotiation algorithm is given as following.

1. Buyer and seller register themselves with well known registration center making themselves visible to all.
2. Registration center will match the two parties with common objective assigning one as buyer and other as seller.
3. Both parties will be represented by agents of their choice. Issues of conflict may be mutually decided or pre-decided. All the preferences of two parties should be known to their corresponding agents only. Initialization of negotiation begins with nTime=0 & n=0. Suppose agent A starts the negotiation with agent B.
4. $n = n + 1$, Agent A will generate the proposal for the *nth* round and then wait for the response
5. If the agent A receives the refusal proposal or nTime>deadline(T) goto step 8. If nTime < T and agent receives accepted proposal goto step 7. Else it will receive a counter proposal and will evaluate its utility.
6. If agent A refuses B's counter proposal or nTime > T then negotiation failed, goto step 8. Else if it accepts agents counter proposal and nTime < Time goto step 7. Else agent A will go to next round, goto step 4
7. Negotiation is successful, goto step 9
8. Negotiation failed. Reason of failure may be time out or one of the agents may refuse the opponents offer.
9. The negotiation is over.

All the offers and counter offers generated during negotiation are stored in a database which is accessible to the agent only. This database can be used by the agent to learn about the behavior of opponent and also predict the next offers of the opponent.

The utility function used for evaluating the offers is given as:

$$U(I_1, I_2, ... I_n) = \sum_{i=1}^{n} W_i \times \frac{eval(I_i)}{max(eval(I_i))}$$

where U is the utility function, I represents issues and W is the weight assigned to the issue. Each discrete value that an issue can take is assigned some evaluation value represented by eval(I_i)

We have added a discount factor(d) in [0,1] to each scenario to make negotiation more competitive and interesting. Discounted utility(U^t_D) at time t is given by

$$U^t_D = U^t * d^t$$

If d = 1, such a scenario is considered to be undiscounted and the utility is not affected by time, while if d is very small the agents are under high pressure to reach an agreement.

3.3 Sample negotiation scenario

Consider a simple negotiation set up with the objective of trying to secure a contract between two companies named Casa Ltd and Rosa Inc. Rosa wants to sell an aircraft which Casa is considering purchasing. Misty and Smiley are two software agents where ”Misty” negotiates on behalf of Rosa Inc. and ”Smiley” represents Casa Ltd.

We have considered only two issues in this simple negotiation: the price of the aircraft and the warranty period. It has been observed that the normal price of this aircraft is in the range of $300,000 to $320,000. The sensible increase is of $10,000. Thus, the price options are $300,000, $310,000, and $320,000. In this industry there are four types of warranty periods available. The options are: 2 year, one year, 6 month or no warranty.

For preparation of negotiation Misty and Smiley along with Rosa Inc and Casa Ltd each rated the two issues. The pre-negotiation steps are conducted separately and the opponent is not aware about the ratings.

Misty's Issue Ratings

Negotiation Issue	Rating
Price	70
Warranty	30

Smiley's Issue Ratings

(Misty does not see this)

Negotiation Issue	Rating
Price	50
Warranty	50

Issue Ratings

Each issue can have one or more options, say, price has three options: 300,000 $, 310,000 $, 320,000 $. After rating the issues, the options in each issue must also be rated similarly. For each issue at least one of the options must be assigned maximum rating and at least one of the options must be assigned a zero rating.

After the user assigns the ratings for each issue and each option, agents calculate ratings for complete packages that are the subject of negotiations. Each package consists of some value for price and warranty, for example,”320,000$ and No warranty” is one complete package. Each company is presented a few packages and their ratings so that they can evaluate if the ratings are correctly assigned.

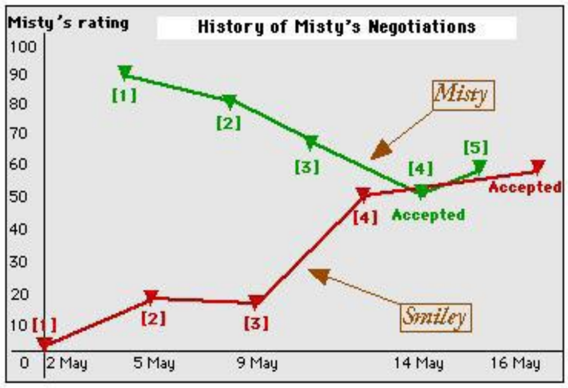
Price	Rating (Max = 70)
300 000 $	0
310 000 $	45
320 000 $	70

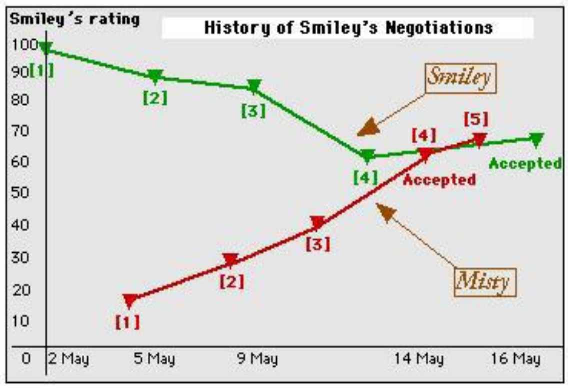
Warranty	Rating (Max = 30)
No warranty	30
One year	30
6 months	10
Two years	0

Option ratings

Then the exchange of offers and counter offers begins where each agent offers some concession over his own previous offer. Concessions offered in each offer depend on the strategy followed and the prediction results obtained from the previous offers. Misty and Smiley continue to exchange offers and messages till one of them accepts the offer of other or the time period is elapsed. The graphical representation of offers and counter offers can help the users to understand the offer generation process.

The graph that Misty and Smiley sees are:





4. Results and discussions

4.1 Negotiation algorithm

We have created two scenarios in which the agents will negotiate namely 'Laptop' and 'Camera'. Figure 1 and 2 shows the graphical representation of the outcome space of both scenarios.

Laptop: There are three issues: brand of laptop, hard disk size, and external monitor size. Each issue has only three options, making a total of 27 possible outcomes.

Camera: This scenario represents the negotiation between a buyer and a seller of a camera. It has six issues: maker, body, lens, tripod, bags, and accessories and each issue has multiple options. The size of the outcome space for this scenario is 3600.

Figure 3 shows the results for actual negotiation process between two agents for laptop domain.

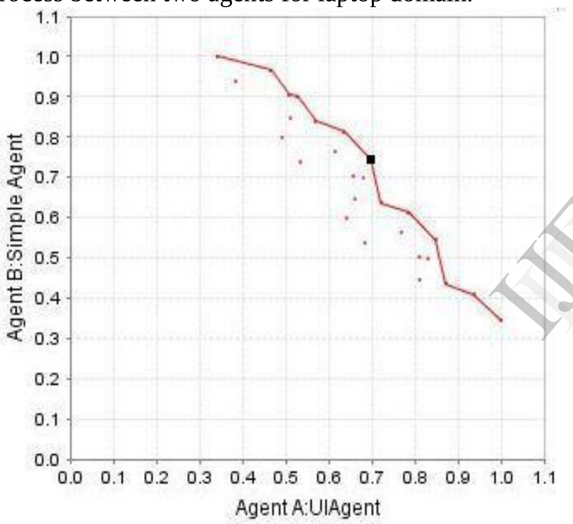


Figure 1: Outcome space for laptop scenario

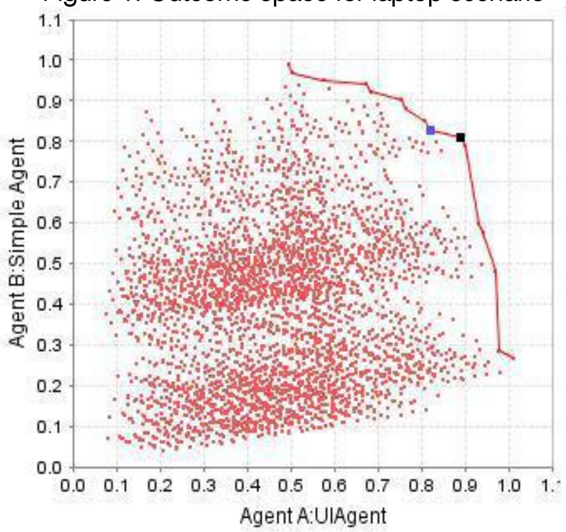


Figure 2: Outcome space for camera scenario

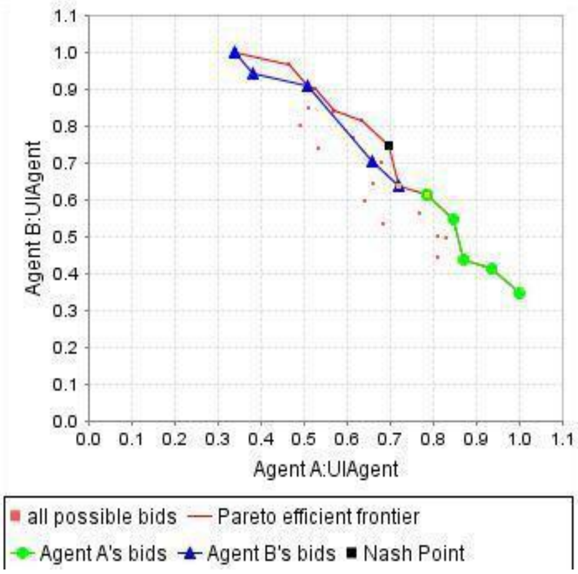


Figure 3: Negotiation between two agents for laptop domain

4.2 Behaviour prediction

We have identified 9 strategies which an agent can follow during negotiation. Faratin had proposed three families of strategies and we have further divided each family into three strategies. Three families proposed by Faratin [5] are:

- Time dependant
- Behavior dependant
- Resource dependant

Each of these families may be hardheaded, linear or conceder. So we have total of 9 combinations of these strategies. A graph is drawn for each of these 9 strategies. The graph generated by the opponent's offers is matched with these graphs to predict the opponent's strategy. Association and co-relation rules will be used to determine the similarity between the graphs.

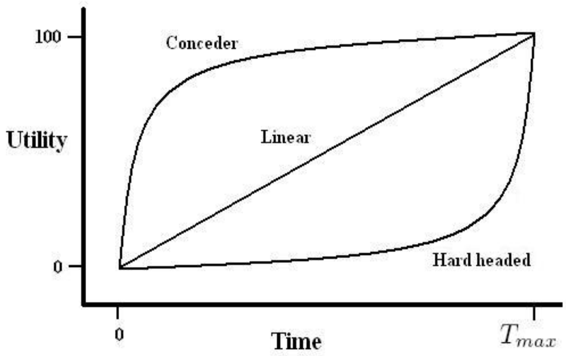


Figure 4: Types of strategies

5. Conclusion

The system will be capable of negotiating without human intervention. A user is only required at the beginning of the negotiation process to specify its requirements which is given as input to the agents. Introduction of prediction mechanism will further improve the efficiency of the negotiation system. However both the fields of automated negotiation and behavior prediction are in their initial stages of development and no such software or application is available for the simulation of these processes. The system can be extended to multi-lateral negotiation in future. Also the agents do not necessarily scale when matched with people. Emotions, cultural differences and computability need to be taken into consideration when developing such agents. Another direction of further work could be around mediated negotiation scenarios, in which a mediator agent has the task to find the mutually acceptable offer given the requirements of both the parties.

References

[1] C. Hou, ”Predicting agent’s tactics in automated negotiation,” Proc. IEEE/WIC/ACM Int’l Conf. Intelligent Agent Technology (IAT ’04), pp. 127-133, 2004.
[2] Beheshti, R.; Mozayani, N. ”Predicting opponents offers in multi-agent negotiations using ARTMAP neural network”, Second International Conference on Future Information Technology and Management Engineering, 2009. FITME ’09. Pp 600-603, 2009
[3] Real Carbonneau , Gregory E. Kersten , Rustam Vahidov,” Predicting opponent’s moves in electronic negotiations using neural networks”, Expert Systems with Applications: An International Journal , Volume 34 Issue 2 ,2008
[4] Debajyoti Mukhopadhyay, Sheetal Vij, Suyog Tasare, ”NAAS: Negotiation automation architecture with buyers behavior pattern prediction component”, Advances in Intelligent Systems and Computing Volume 176, pp 425-434,2012
[5] Faratin P,”Automated service negotiation between autonomous compositional agents”, PhD thesis, Queen Mary and Westfield college, University of London,UK, 2000
[6] Hsin Rau, Chao-Wen Chen, Wei-Jung Shiang, Chiuhsiang Joe Lin, ”Develop an adapted coordination strategy for negotiation in a buyer-driven Emarketplace”, Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, pp 3224-3229, 2008
[7] Farhana H. Zulkernine and Patrick Martin,”An Adaptive and Intelligent SLA Negotiation System for Web Services”,IEEE Transactions on service computing, VOL. 4,pp 31-43, 2011
[8] Galit Haim, Sarit Kraus and Yael Blumberg,” Learning Human Negotiation Behavior Across Cultures”,Second International Working Conference on Human Factors and Computational Models in Negotiation, 2010
[9] Zeng, D., Sycara, K. ”Bayesian learning in negotiation”. International Journal of Human-Computer Studies 48, 125-141.1998
[10] J. Brzostowski and R. Kowalczyk, ”Predicting partner’s behaviour in agent negotiation,” Proc. Int’l Joint Conf. Autonomous Agents and Multiagent Systems , pp. 355-361, 2006.
[11] Cao Mukun,” Multi-agent automated negotiation as a service”, 7th International Conference on Service Systems and Service Management (ICSSSM), pp 1-6,2010
[12] Raz Lin and Sarit Kraus,”Magazine communications of the ACM”,Volume 53 Issue 1, January 2010
[13] Roussaki, I.; Papaioannou, I.; Anagnostou, M.,” Employing neural networks to assist negotiating intelligent agents”, 2nd IET International Conference on Intelligent Environments, Volume: 1 pp: 101-110, 2006
[14] Sanghyun Park; Sung-Bong Yang,” An automated system based on Incremental learning with applicability toward multilateral negotiations ”SICEICASE International Joint Conference, pp: 6001-6006, 2006
[15] Ning Liu; DongXia Zheng; YaoHua Xiong,” Multiagent negotiation model based on RBF neural network learning mechanism”, International Symposium on Intelligent Information Technology Application Workshops, pp 133-136, 2008
[16] Hamid Jazayeriy, Masrah Azmi-Murad, Md. Nasir Sulaiman and Nur Izura Udzir,” A review on soft computing techniques in automated negotiation”, Academic journals for Scientific research and essays, Volme 6(24), pp 5100-5106, 2011
[17] Bangqing Li; Yulan Ma,” An auction-based negotiation model in intelligent multi-agent system”, International Conference on Neural Networks and Brain, volume 1, pp 178-182, 2005
[18] Fenghui Ren and Minjie Zhang,”Prediction of partners behaviors in agent negotiation under open and dynamic environments”, proceedings of International Conferences on Web Intelligence and Intelligent Agent Technology, pp 379-382, 2007