2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

E-Negotiation of Dependent Multiple Issues by Using a Joint Search Strategy

Ta-Chiun Chou, Li-Chen Fu, and Kuang-Ping Liu, *National Taiwan University*

***Abstract*—Negotiations have been a widespread research topic in politics, economics, and management for decades. Recently, with the rapid growth of on-line bargains, automatic negotiations have become more and more important. Although many automatic negotiation strategies have been presented, most of them are focused on simple negotiations composed of independent multiple issues. These strategies can not be applied to realistic complicated negotiations made up of dependent multiple issues. Therefore, we propose a mechanism named Joint Genetic Algorithm (JGA) to deal with E-Negotiations of dependent multiple issues. In JGA, a joint search strategy is applied to find the satisfactory contract accepted by both parties, by means of the genetic algorithm to predict and learn opponent's preference. Experimental results show that JGA can facilitate to make a deal efficiently under different circumstances of conflict scenarios.**

I. Introduction

NEGOTIATIONS have been a widespread research topic in politics, economics, and management for decades. In recent years, with the rapid growth of on-line transactions, automatic negotiations have become more and more important. Above all, when there are a lot of trading issues in one offer, automatic negotiations will be more effective than manual negotiations.

Although many automatic negotiation strategies have been presented, most of them were focused on simple negotiations composed of independent multiple issues. Nevertheless, these strategies can not be applied to realistic complicated negotiations made up of dependent multiple issues.

Therefore, in this paper, we will propose a negotiation strategy to reach an agreement on dependent multiple issues. We use the genetic algorithm (GA) to develop our joint search algorithm by using our joint operations: joint selection operation, joint elitism operation, and joint fitness operation. The joint selection operation is mainly focused on diversification in own offer space. The joint elitism operation and joint fitness operation are implemented to predict the opponent's preference. Hence, our algorithm can find the satisfactory solution both in own and opponent's solution spaces. Finally, our experimental result will demonstrate that our strategy can make a deal more efficiently.

Manuscript received January 7, 2007. This research is sponsored by National Science Council under Research Grant with No. NSC 94-2752-E-002-007- PAE.

Ta-Chiun Chou and Kuang-Ping Liu, are with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan (e-mail: {tcchoub, amyliu}@sgi.csie.ntu.edu.tw).

Li-Chen Fu, is a Professor with the Department of Electrical Engineering and the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan (e-mail:lichen@ntu.edu.tw).

The rest of this paper is organized as follows. In Section II, we will give the related works. In Section III, we explain the mechanism of negotiation with dependent multiple issues. In Section IV, we will describe our joint search strategy. Experiments will be demonstrated in Section V. Finally, we make a conclusion in Section VI.

II. Related Work

This paper is built based on the past research in several important ways. First, with the popularity of internet, e-commerce framework has been established [1]. However, e-commerce market is not like traditional market, where the former is virtual and possible to make a trading at anytime and in anywhere. To fulfill this need, software agent technology was applied to e-commerce and at the same time, some of the automatic negotiation mechanisms were also proposed [2][3]. Furthermore, in order to make a more efficient and complicated negotiation, intelligent agents and distributed agents [4][5] are now widely used in most of the automatic negotiation.

Traditionally, negotiations were discussed by game theory [6]. However, there are some bottlenecks such as unavailable complete information and intractable full rationality of the players when we use the classic game theory to make negotiations. Nevertheless, by using Artificial Intelligent (A.I.) technique to resolve the negotiation problems, these unreachable assumptions seem to be unnecessary. Moreover, although we don't know the opponent's preference, we still have the capability to predict and learn the opponent's behaviors by using some learning skills such as the genetic algorithm and Bayesian rule [7][8]. Thus, we can find the best strategy to reach the more satisfactory agreement. Other A.I. approaches [9]-[13] of utilizing heuristic search and evaluation under the limited computation resource were discussed. The evaluation mechanism of inter-dependent negotiation issues was proposed in [14]. In this way, we will be able to negotiate for more complex transactions existing in our real world bargaining.

III. Multi-Issue Negotiations

A multi-issue negotiation consists of a lot of negotiation issues. In a multi-issue negotiation, each issue plays a specific role in the evaluation of an offer. Generally, we could divide negotiation issues into two groups: independent issues and dependent issues.

A. Independent Multi-Issue Negotiations

An independent issue means that the value of this issue will not affect the importance of other issues and the importance of this issue will not be affected by the values of other issues when we evaluate an offer. Therefore, we can view the weight value of an independent issue as a fixed value. Thus, the utility value contributed by this issue will be determined by the product of the corresponding weight value and the value of this issue. For example as Table I, one buyer would like to purchase a car, including 10 independent multiple issues, from one seller.

TABLE I
WEIGHT VALUE IN A CAR PURCHASE NEGOTIATION
(INDEPENDENT ISSUES)

Issue Weight in Independent Negotiation									
w1	w2	w3	w4	w5	w6	w7	w8	w9	w10
0.15	0.05	0.15	0.05	0.02	0.18	0.07	0.13	0.12	0.08

TABLE II
WEIGHTED SUM METHOD IN A CAR PURCHASE NEGOTIATION
(INDEPENDENT ISSUES)

$$\text{Utility(Offer)}= W_1{*}I_1+W_2{*}I_2+W_3{*}I_3+W_4{*}I_4+W_5{*}I_5+ W_6{*}I_6+W_7{*}I_7+W_8{*}I_8+W_9{*}I_9+W_{10}{*}I_{10}$$

- I_i: the value of ith issue
- W_i: the weight value of ith issue

By using the linear weighted sum method to calculate the utility of an offer, we have the formulation in Table II. From this formulation, we can understand that the utility value contributed by <ith> issue is only affected by the fixed weight value of <ith> issue.

B. Dependent Multi-Issue Negotiations

A dependent issue means that the value of this issue will affect the importance of other issues or the importance of this issue will be affected by the values of other issues when we evaluate an offer. That is to say, the corresponding weight value of this issue is not a fixed value. The utility value contributed by this issue will be determined by the product of the corresponding weight value and the value of this issue where the weight is determined by some values involving more issues.

TABLE III
WEIGHT VALUE IN A CAR PURCHASE NEGOTIATION
(DEPENDENT ISSUES)

Issue Weight in Dependent Negotiation									
Master Issue	Slave Issue								
Value / w1	w2	w3	w4	w5	w6	w7	w8	w9	w10
20K~30K / 0.15	0.05	0.15	0.05	0.02	0.18	0.07	0.13	0.12	0.08
30K~40K / 0.25	0.05	0.05	0.10	0.08	0.07	0.16	0.04	0.10	0.10
40K~50K / 0.18	0.02	0.17	0.03	0.11	0.09	0.01	0.19	0.08	0.12

Let's look an example as Table III. One buyer would like to purchase a car including 10 dependent issues, from one seller. We define the following dependent relationship within issues. In this case, the first issue is a master issue. The other 9 issues are slave issues. The weight of corresponding slave issues will be decided by the value of a master issue.

TABLE IV
WEIGHTED SUM METHOD IN A CAR PURCHASE NEGOTIATION
(INDEPENDENT ISSUES)

$$U(O)=\begin{cases}\sum_{i=1}^{i=10} w'_i{*}I_i\ ;\ \text{if } 20K < I_1 < 30K\\ \sum_{i=1}^{i=10} w''_i{*}I_i\ ;\ \text{if } 30K < I_2 < 40K\\ \sum_{i=1}^{i=10} w'''_i{*}I_i\ ;\ \text{if } 40K < I_3 < 50K\end{cases}$$

PS: w'i, w''i, w'''i: the corresponding weight of level-1, level-2, level-3

According to the value of a master issue, we will be able to decide which part of an offer is belonged to. (A "part" refers to a specific range of the value of a master issue in one offer.) Then, we can also get the relative weight value of other 9 slave issues. Table IV depicts the relation of master and slave issues on an offer when we use the linear weight-sum method to compute the utility of an offer.

TABLE V
AN ILLUSTRATION OF OFFERS AND DISTINGUISHED PARTS
FOR SELLER AND BUYER

Seller

Offer \ Issue	I_1	I_2	I_3
1st part	$20K \le I_1 < 30K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$
2nd part	$30K \le I_1 < 40K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$
3rd part	$40K \le I_1 < 50K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$

Buyer

Offer \ Issue	I_1	I_2	I_3
1st part	$20K \le I_1 < 28K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$
2nd part	$28K \le I_1 < 36K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$
3rd part	$36K \le I_1 < 44K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$
4th part	$44K \le I_1 < 50K$	$0 \le I_2 \le 1$	$0 \le I_3 \le 1$

Let's see an example in Table V. There are 3 parts in the seller's offer and 4 parts in the buyer's offer. Hence, if we figure out the chart by offers and utility values, we can get Fig. 1. The goal of our negotiation strategy is to find the optimal solution (or offer) in a lot of peaks and valleys.

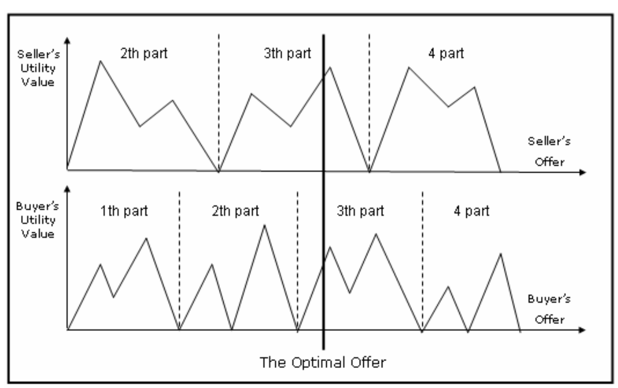

Fig. 1. Dependent Multi-Issue Negotiation.

IV. E-NEGOTIATION BY USING A JOINT SEARCH STRATEGY

A. Joint Search

From the viewpoint of search algorithm, our negotiation on dependent multiple issues is to find the best deal in multiple parts of the offer and both solution spaces in Fig. 1. Therefore, there are two goals that we need to achieve. One is to search good offers existing in different parts according to the master issue. The other is to find good offers from seller's and buyer's offers.

In this paper, we develop a joint search algorithm, named "Joint Genetic Algorithm (JGA)", to find the best offer by using our new joint operations in the genetic algorithm

B. Terms of Joint Genetic Algorithm

In the following, we will show the three features of our JGA. One is the joint selection operation, another is the joint fitness operation, and the other is joint elitism operation. Fig. 2 describes the flow chart of JGA.

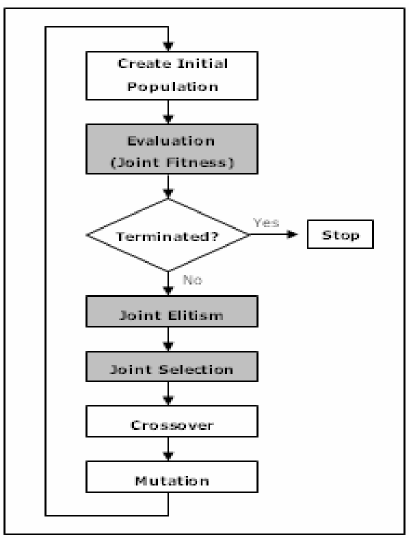

Fig. 2. Joint Genetic Algorithm

1) Joint Elitism Operation

In this research we assume that the opponent's preference is unknown. The only message about the opponent's information is the offers we received during negotiations. From the viewpoint of searching algorithm, an opponent's offer is a point for which the opponent wants to search in his offer space. The latest opponent's offer is better than his old offers for the opponent.

In a GA, due to the unknown opponent's offer space, it is impossible to pick up the opponent's desired offers by the selection operation. Hence, we need to consider another method to involve the opponent's desired offers. The elitism operation plays an important role to reserve good chromosomes to the next generation. Here we develop a joint elitism operation, which is to store the latest offers received from the opponent and to put them into the next generation directly, following the spirit of elitism.

Fig. 3. Joint Elitism Operation

In the proposed joint elitism operation, there is a FIFO elitism queue (see Fig. 3) with a fixed size. When a new offer arrives from the opponent, the oldest one will be removed from this queue if it is full. This is because the latest offer is also the most desired point that the opponent wants to reach in the opponent's space.

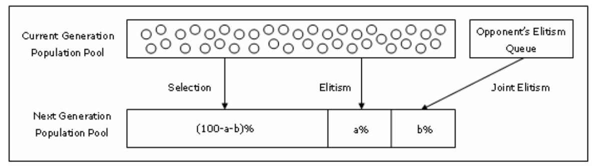

Fig. 4. Joint Elitism Operation

Consequently, the population of the next generation consists of three parts (See Fig. 4): population by selection operation, population by elitism operation, and population by joint elitism operation.

2) Joint Fitness Operation

After we use the joint elitism operation to put the opponent's offers into our population pool, it is possible that the opponent's offers can not be selected to be the parents in mating pool by the fitness operation. Because it is likely that the opponent's offer is not preferred by own fitness function, we will not select the opponent's offer. Therefore, we develop a joint fitness mechanism to pick up the opponent's offers as shown in Fig. 5.

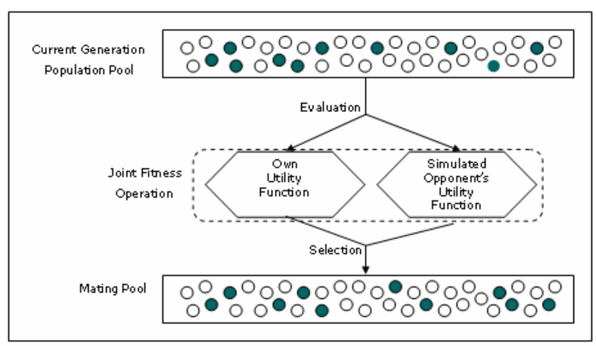

Fig. 5. Joint Fitness Operation

However, we do not have any information about the opponent's preference, and thus we can only simulate the opponent's utility function. We will use the only message, the opponent's offers, to implement a simulated utility function. We use Euclidean Distance, which examines the root of squared differences between coordinates of a pair of objects, to calculate the difference between the latest opponent's offer and the offer which we need to evaluate. The distance represents the degree of similarity between two offers. Hence, we can view the distance between the current offer and the latest opponent offer as the simulated utility function value. Besides, we use a linear weighted sum to combine own utility function and simulated utility function as in Fig. 6.

$$Fitness(O) = (1-c) \times \frac{U(O)}{U(O_{max})} + c \times \left\{1 - \frac{EuclideanDist(O, O_{latest\_opponent})}{EuclideanDist(Negotiation\ Space)}\right\}$$

- c: Joint fitness rate
- O: Offer which we need to evaluate
- $U(O)$: Utility function of offer O
- O_{max}: Offer with maximum utility value
- $U(O_{max})$: Utility function of offer O_{max}
- $O_{latest\_opponent}$: The latest offer receiving from opponent
- $EuclideanDist(O, O_{latest\_opponent})$: Euclidean distance between O and $O_{latest\_opponent}$
- $EuclideanDist(Negotiation\_Space)$: The maximum distance in the negotiation space

Fig. 6. Joint Fitness Function

3) *Joint Selection Operation*

In this operation, we will focus on the joint search to search in each part of an offer simultaneously under the limited computational resource. In our joint selection operation, we select the offers by using the round-robin rule based on the roulette wheel selection. It is described as in Fig.7. After we get an offer by roulette wheel selection, we will follow the sequence (Part-A → Part-B→ Part-C → Part-A→ Part-B→ Part-C) to make a selection. For example, we get the offer of Part-A by roulette wheel selection. Meanwhile, Part-A is also the part in which we want to gain an offer according to the sequence of round-robin. As a result, we will pick up this offer from the population of current generation to next generation. However, if we get the offer of Part-B which is not the part we want according to the sequence of round-robin, we will give up this offer and then make a roulette wheel selection again until we get the offer of Part-A.

In this way, it is guaranteed that we can pick up the offers coming from each part and there is the same amount of offers from each part. Thus, in any generation, there will always be offers coming from each part. Consequently, we will be able to avoid falling in locally optimal offers and reach a great diversity. When there are a lot of locally optimal offers existing in our negotiation, we can jump out of these local peaks and get our globally optimal offer easily.

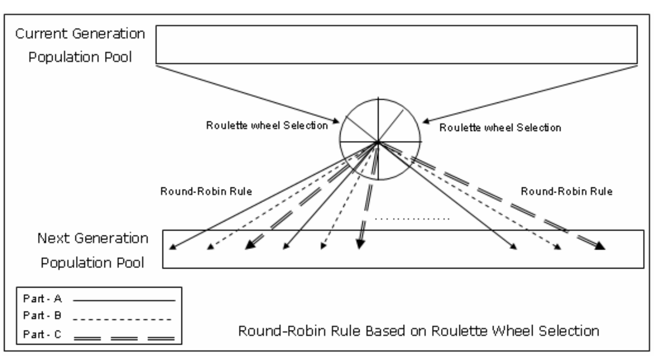

Fig. 7. Joint Selection Operation

C. *Negotiation Procedure*

We take the protocol of alternative offer. It means that a seller and a buyer send the offer in turn. We suppose that the seller will send an offer to the buyer first. After the buyer receives an offer, the buyer will send a counter offer to the seller by adjusting his offer and predicting the opponent's preferences. After repeating this procedure several times, the joint payoff, the product of seller's utility value and buyer's utility value, will increase step by step. Finally, we will get the satisfactory offer accepted by both parties. The detailed procedure of E-Negotiation by using a joint search strategy is described as in Fig. 8.

```
/* Joint Genetic Algorithm */
Set (generation_number); /* Set the generation number */
Set (the_first_player); /* Set the first player */
While (generation_number !=0){

    /* Player A */
    playerA_recieve_offer_from_playerB();
    playerA_generate_counter_offer();
    playerA_send_counter_offer_to_playerB();

    /* Player B */
    playerB_recieve_offer_from_playerA();
    playerB_generate_counter_offer();
    playerB_send_counter_offer_to_playerA();

    /* Enter the next generation */
    generation_number--;
}
```

Fig. 8. Negotiation Procedure

V. EXPERIMENT

A. *Experiment Design*

Our experiment scenario is that one buyer wants to buy a computer from one seller. They begin a negotiation with 10 dependent issues in Table VI.

TABLE VI
10 NEGOTIATION ISSUES IN THE EXPERIMENT

Issue	Weight	Name	Range
I_1	W_1	Price	0.0 ~ 1.0
I_2	W_2	CPU	0.0 ~ 1.0
I_3	W_3	Memory	0.0 ~ 1.0
I_4	W_4	Hard Disk	0.0 ~ 1.0
I_5	W_5	LCD	0.0 ~ 1.0
I_6	W_6	CD/DVD ROM	0.0 ~ 1.0
I_7	W_7	Network	0.0 ~ 1.0
I_8	W_8	Bluetooth	0.0 ~ 1.0
I_9	W_9	Web Camera	0.0 ~ 1.0
I_{10}	W_{10}	Mouse	0.0 ~ 1.0

The detailed weight value list is in Table VII and Table VIII. The first issue, price, is the master issue. The other issues are slave issues. We get the weight values of slave issues according to the value of the master issue.

TABLE VII
WEIGHT LIST OF SELLER'S ISSUE

I_1 (Price)	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	W_9	W_{10}
0~20000	0.12	0.08	0.15	0.05	0.02	0.18	0.07	0.13	0.12	0.08
20001~35000	0.12	0.09	0.06	0.13	0.12	0.08	0.16	0.04	0.10	0.10
35001~50000	0.12	0.05	0.17	0.03	0.11	0.09	0.04	0.19	0.08	0.12
50001~65000	0.12	0.11	0.02	0.15	0.10	0.10	0.06	0.14	0.13	0.07
65001~80000	0.12	0.05	0.17	0.06	0.11	0.09	0.01	0.19	0.08	0.12

TABLE VIII
WEIGHT LIST OF BUYER'S ISSUE

I_1 (Price)	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	W_9	W_{10}
0~20000	0.11	0.12	0.08	0.09	0.15	0.05	0.13	0.17	0.03	0.07
20001~35000	0.11	0.10	0.13	0.11	0.08	0.07	0.10	0.10	0.16	0.04
35001~50000	0.11	0.02	0.19	0.08	0.08	0.10	0.07	0.10	0.13	0.12
50001~65000	0.11	0.07	0.10	0.06	0.12	0.14	0.08	0.10	0.13	0.09
65001~80000	0.11	0.04	0.16	0.09	0.06	0.10	0.19	0.10	0.01	0.14

During the negotiation, two GAs are executed in turn. One is for the seller and another is for the buyer. The experiment is terminated after 1000 generations. We use joint utility value (the product of seller's utility value and buyer's utility value) to evaluate which offer is better for us. Three negotiation methods are taken in our experiments as follows:

- Method-I: GA (Traditional)
- Method-II: GA + Joint Elitism + Joint Fitness
- Method-III: GA + Joint Elitism + Joint Fitness + Joint Selection

Besides, we use two different negotiation scenarios (See Table IX and Table X) in our experiments:

- Scenario-I: the moderate conflict between seller's and buyer's preferences
- Scenario-II: the high conflict between seller's and buyer's preferences

TABLE IX
SCENARIO-I: THE UTILITY FUNCTION

	Seller's Utility Function	Buyer's Utility Function
I_1	$(I_1) * (W_1)$	$(1-I_1) * (W_1)$
I_2	$(I_2) * (W_2)$	$(I_2) * (W_2)$
I_3	$(I_3 * I_3) * (W_3)$	$(1-I_3 * I_3) * (W_3)$
I_4	$(I_4 * I_4)^{1/2} * (W_4)$	$(I_4 * I_4)^{1/2} * (W_4)$
I_5	$(I_5) * (W_5)$	$(1-I_5) * (W_5)$
I_6	$(I_6 * I_6) * (W_6)$	$(I_6 * I_6) * (W_6)$
I_7	$(1-I_7) * (W_7)$	$(I_7) * (W_7)$
I_8	$(\sin(Pi/2*I_8)) * (W_8)$	$(\sin(Pi/2*I_8)) * (W_8)$
I_9	$(1-I_9 * I_9) * (W_9)$	$(I_9 * I_9) * (W_9)$
I_{10}	$(\tan(Pi*/4*I_{10})) * (W_{10})$	$(\tan(Pi*/4*I_{10})) * (W_{10})$

- I_i: the value of <I_i>
- W_i: the weight value of <I_i>

TABLE X
SCENARIO-II: THE UTILITY FUNCTION

	Seller's Utility Function	Buyer's Utility Function
I_1	$(I_1) * (W_1)$	$(1-I_1) * (W_1)$
I_2	$(1-I_2) * (W_2)$	$(I_2) * (W_2)$
I_3	$(I_3 * I_3) * (W_3)$	$(1-I_3 * I_3) * (W_3)$
I_4	$(1-(I_4 * I_4)^{1/2}) * (W_4)$	$((I_4 * I_4)^{1/2}) * (W_4)$
I_5	$(I_5) * (W_5)$	$(1-I_5) * (W_5)$
I_6	$(1-I_6 * I_6) * (W_6)$	$(I_6 * I_6) * (W_6)$
I_7	$(1-I_7) * (W_7)$	$(I_7) * (W_7)$
I_8	$(\sin(Pi/2*I_8)) * (W_8)$	$(1- \sin(Pi/2*I_8)) * (W_8)$
I_9	$(1-I_9 * I_9) * (W_9)$	$(I_9 * I_9) * (W_9)$
I_{10}	$(1-\tan(Pi*/4*I_{10})) * (W_{10})$	$(\tan(Pi*/4*I_{10})) * (W_{10})$

- I_i: the value of <I_i>
- W_i: the weight value of <I_i>

B. Experiment Parameters

In our experiment, one chromosome represents one offer and one gene represents one issue. One chromosome includes 10 genes. There are 100 chromosomes in the population pool. The GA parameters are listed in Table XI.

TABLE XI
PARAMETERS IN GA

Generation Number	Population Size	Crossover Rate	Mutation Rate	Elitism Rate	Joint Elitism Rate (b)	Joint Fitness Rate (c)
1000	100	0.5	0.1	0.1	0.25	0.5

C. Experimental Results

1) Scenario-I:

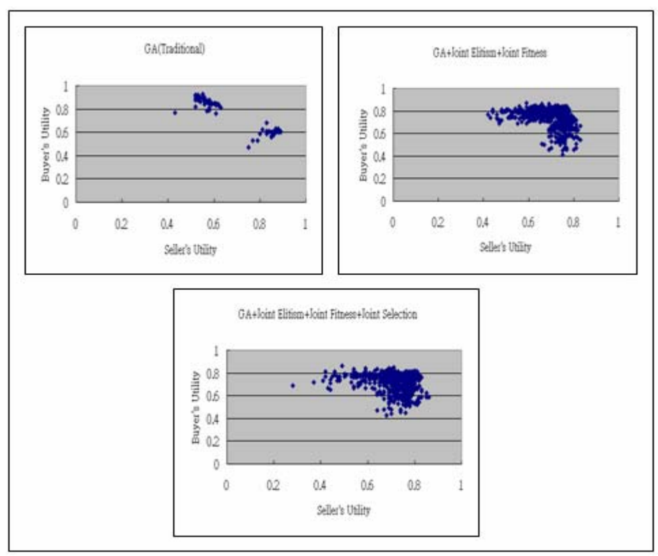

Fig. 9. Scenario-I: Experimental Result

TABLE XII
SCENARIO-I: COMPARISON OF JOINT UTILITY

Method	Joint Utility (Seller's Utility*Buyer's Utility)
GA	0.5644
GA+ Joint Elitism+ Joint Fitness	0.6468
GA+ Joint Elitism+ Joint Fitness+ Joint Selection	0.6561

2) Scenario-II:

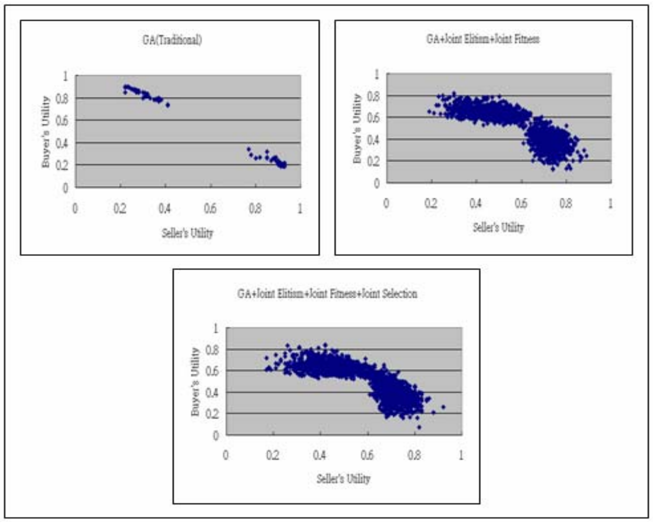

Fig. 10. Scenario-II: Experimental Result

TABLE XIII
SCENARIO-II: COMPARISON OF JOINT UTILITY

Method	Joint Utility (Seller's Utility*Buyer's Utility)
GA	0.3034
GA+ Joint Elitism+ Joint Fitness	0.4160
GA+ Joint Elitism+ Joint Fitness+ Joint Selection	0.4355

D. Summary

The analysis of experimental results in Fig. 9, Fig. 10, Table XII, and Table XIII is summarized as the following:

- The traditional GA gets the worst searching result, the lowest joint utility. The results are almost located in the both ends. Each party tries to get its own profit, not taking any account of the other party.
- The GA with joint elitism and joint fitness triggers the searching direction near to the center, the opponent's preference. That is to say, it can find the better offers with the higher joint utility accepted by the opponent.
- The GA with the joint selection improves the searching result to get the solutions with the higher joint utility. The trend of experimental result towards to the center more concentrative. It is because GA with the joint selection can increase the diversity in the solution space.
- The JGA improves the performance (the joint utility) much in Scenario-II, but little in Scenario-I. So, when there is a high conflict negotiation, the JGA can make a deal with the higher payoffs.

VI. CONCLUSION

E-Negotiation will be the core technology in the next generation of E-Commerce. Traditional negotiation mechanisms are only to make the simple negotiations consisting of independent multiple issues. Hence, in this paper, we propose the joint search strategy to resolve the realistic complex negotiations made up of dependent multiple issues.

First of all, we introduce the mechanism of dependent multi-issue negotiation. It is more suitable than any others proposed before and can actually be applied to real world complicated negotiations.

Furthermore, we develop the joint search strategy, named as JGA, by using our new joint operations: joint elitism operation, joint fitness operation, and joint selection operation. In the JGA, we search not only in each part of own offer space, but also in the seller's and the buyer's offer spaces simultaneously. Consequently, we can avoid falling into locally optimal solutions by increasing more diversity in our solution space.

Finally, experiments are executed in two different negotiation scenarios: one with moderate conflict and the other with high conflict between seller's and buyer's preferences. The experiment result shows that the JGA can improve the performance and get the maximum joint payoffs in both scenarios.

REFERENCES

[1] J.H. Wu, and T.L. Hisa, *"Analysis of E-commerce innovation and impact: A hypercube model", Electronic Commerce Research and Applications,* vol 3, pp. 389-404, 2004

[2] N.R. Jennings, *"Automated Negotiation - Prospects, Methods and Challenges", International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199-215, 2000.

[3] C. Li, J.A. Giampapa, and K. Sycara, *"A Review of Research Literature on Bilateral Negotiations", tech. report CMU-RI-TR-03-41*, Robotics Institute, Carnegie Mellon University, November, 2003.

[4] K. Sycara, K. Decker, A. Pannu, M. Williamson, and D. Zeng, *"Distributed Intelligent Agents," IEEE Transactions on Intelligent Systems*, vol. 11, no. 6, pp. 36-46, Dec 1996.

[5] R. Krovi, *"Agent Behaviors in Virtual Negotiation Environments", IEEE Transactions on Systems, Man and Cybernetics,* Part C, vol. 29, no.1, pp. 15-25, Feb. 1999.

[6] S.H. Wu, *"Agent coordination by Negotiation in Trusted-Third-Party Mediated Games"*, PhD Thesis, National Tsing Hua University, 1999

[7] D. Zeng, and K. Sycara, *"Bayesian Learning in Negotiation," International Journal of Human Computer Systems*, vol. 48, pp.125-141, 1998.

[8] Y. Ishihara, R. Huang, K. M. Sim, *"Learning opponent's eagerness with Bayesian updating rule in a market-driven negotiation model", IEEE International Conference on Advanced Information Networking and Applications*, vol 1, pp. 903 - 908, 2005

[9] J.R. Oliver, "A Machine-Learning Approach to Automated Negotiation and Prospects for Electronic Commerce", *Journal of Management Information Systems*, vol 13, no. 3, Winter 1997, pp. 83-112, 1997

[10] B. Rubenstein, *"A Weighted Sum Genetic Algorithm to Support Multiple Party Multiple-Objective Negotiations", IEEE Transactions on Evolutionary Computation*, vol 6, no. 4, pp. 366-377, 2002

[11] J.R. Lin, *"Automated Multi-Issue Negotiations between Computational Agents in a Dynamic Environment"*, PhD Thesis, National Taiwan University, 2003

[12] R.Y.K Lau, *"Towards Genetically Optimised Multi-Agent Multi-Issue Negotiations", Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pp 29- 35, 2005

[13] E. H. Gerding and H. L. Poutre, *"Bilateral Bargaining with Multiple Opportunities: Knowing Your Opponent's Bargaining Position," IEEE Transactions on Systems, Man and Cybernetics*, Part C, vol. 36, no.1, pp. 45-55, Jan 2006.

[14] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam, *"Protocols for Negotiating Complex Contracts," IEEE Transactions on Intelligent Systems*, vol. 18, no.6, pp. 32-38, Nov/Dec 2003.