Acquisition of a concession strategy in multi-issue negotiation

Yasumura, Yoshiaki
Kamiryo, Takahiko
Yoshikawa, Shohei
Uehara, Kuniaki

(Citation)
Web Intelligence and Agent Systems,7(2):161-171

(Issue Date)
2009

(Resource Type)
journal article

(Version)
Accepted Manuscript

(URL)
https://hdl.handle.net/20.500.14094/90001007

Acquisition of a Concession Strategy in Multi-Issue Negotiation

Yoshiaki YASUMURA [a,1], Takahiko KAMIRYO [a], Shohei YOSHIKAWA [a], and Kuniaki UEHARA [a]

[a] *Graduate School of Engineering, Kobe University, Japan*

Abstract. This paper presents a method for acquiring a concession strategy of an agent in multi-issue negotiation. This method learns how to make a concession to an opponent for realizing win-win negotiation. To learn the concession strategy, we adopt reinforcement learning. First, an agent receives a proposal from an opponent. The agent recognizes a negotiation state using the difference between their proposals and the difference between their concessions. According to the state, the agent makes a proposal by reinforcement learning. A reward of the learning is a profit of an agreement and a punishment of negotiation breakdown. The experimental results showed that the agents could acquire the negotiation strategy that avoids negotiation breakdown and increases profits of an agreement. As a result, agents can acquire the action policy that strikes a balance between cooperation and competition.

Keywords. Multi-Issue Negotiation, Concession Strategy, Reinforcement Learning, Win-Win Negotiation

1. Introduction

As the Internet spreads throughout the world, the use of e-commerce is growing. For e-commerce, we need transaction system using the network and technologies for effectively supporting users[10,2]. Especially, a negotiation support system is required for decreasing the difference of users' negotiating skills. For example, negotiation agents can reduce the users' load by negotiating automatically instead of users. For supporting negotiation, the negotiation agents should have good strategies to deal with various opponents.

So far, two types of negotiation have been researched. The first type is single-issue negotiation [7,16,17,18,19] that deal with only a price. However, in an actual negotiation, we should consider multiple issues such as the number of items, delivery date, and services in addition to a price. The second type is multi-issue negotiation that considers multiple issues in addition to a price [1,3,4,5,6,13,15]. In this type of negotiation, as the number of issues increases, the negotiation becomes more complex. In order to lead a good agreement, negotiation strategies for agents are important in multi-issue negotiation. In multi-issue negotiation, a good negotiation strategy is required to increase benefits of both sides. This strategy is called win-win strategy.

[1] Yoshiaki Yasumura, 1-1 Rokkodai Nada Kobe, Japan; E-mail: yasumura@ai.cs.kobe-u.ac.jp

In order to realize a win-win negotiation strategy, we should consider two points. The first point is to take account of each other's preferences. Considering each other's preferences enables to increase mutual benefits. The second point is a concession strategy to opponents. The difference of each other's concession strategy causes imbalance of their benefits. For example, when an agent makes concession largely and an opponent makes little concession at the beginning of the negotiation, the agent might make a loss. A solution of this problem is that agents acquire the concession strategy that can deal with various opponents.

In this paper, to adapt to opponents that have various preferences and strategies, we propose a method for learning negotiation strategy with reinforcement learning. Our goal is to realize win-win negotiation for various negotiation opponents. First, for realizing win-win negotiation, we propose a method for making a proposal to reach win-win agreement by considering opponent's preference. Second, for adapting to various opponents, we propose a method for learning negotiation strategy with reinforcement learning. In order to learn the action policy that leads better agreements, we should acquire the action policy that strikes a balance between cooperation and competition. Although reinforcement learning requires many experiences for acquiring the optimal action policy, a negotiation agent can efficiently learn from simulation experiences with various types of opponents. After the agent acquires the better strategy by the simulation, it can conduct good negotiation with various opponents.

2. Related Work

So far, automated negotiation has been studied by many researchers based on many fields such as game theory, operational research, decision theory and multi-agent.

Some of the researches are based on game theory, which mathematically analyzes negotiation strategies. However, it is difficult to apply the game theoretical strategies to real negotiation situation as mentioned in [9], because game theory basically assumes complete information. In real negotiation, negotiation agents have incomplete information about the negotiation opponents.

Most of negotiation strategy researches are based on simulation that uses negotiation agents. The negotiation settings of these researches are classified into two types. The first type is single-issue negotiation that deals with only a price [7,16,17,18,19] . In this type of negotiation, sellers and buyers present bidding prices alternately to lead to an agreement. In an actual negotiation we should consider multiple issues such as the number of items, delivery date and services in addition to a price. Single-issue negotiation cannot deal with multiple issues. The second type is multi-issue negotiation that considers multiple issues in addition to a price[1,3,4,5,6,13,15]. In this type of negotiation, as the number of issues increases, the negotiation becomes more complex and leading a good agreement becomes more difficult. In order to lead a good agreement, good negotiation strategies for agents are essential in multi-issue negotiation.

In multi-issue negotiation, it is desirable to increase mutual benefits for reaching a good agreement. This negotiation is called win-win negotiation. For realizing win-win negotiation, negotiators should make proposals by reflecting each other's preferences. However, most of the existing negotiation strategies are simple methods that bring own proposal close to opponent's proposal.

Lau et.al. [13] proposed the strategy that learns a preferable proposal for the opponent and makes a proposal considering the balance of their preferences. However, this method cannot adapt to the difference of products and opponents because the agent need to learn strategy for the different preference or products again. Jonker et.al [3] proposed an effective negotiation model in multi-issue negotiation. However, this model is not flexible for the various opponents' strategy because the concession rate is fixed.

As mentioned above, the existing methods assumed that some conditions (ex. the products, opponent's strategy) are fixed. Therefore, they cannot deal with various opponents or products. In this paper, for dealing with various opponents and products, we propose a method for learning a strategy by negotiating in variety of conditions such as opponents and products. For acquiring a good negotiation strategy, a negotiation agent needs a supervisor. In negotiation, however, it is difficult to get supervised signals for two reasons. First, it is costly to obtain the best proposal as the supervised signal from a supervisor through many negotiations. Second, it is hard for supervisors to decide what the best proposal is.

In negotiation, an agent can learn from negotiation results instead of a supervisor. For learning from results, a negotiation agent learns with reinforcement learning. However, there are no methods that learn a negotiation strategy with reinforcement learning. Therefore, we propose a method for acquiring a negotiation strategy with reinforcement learning. By this method, we can acquire the strategy that can deal with various opponents through a lot of negotiation with various opponents.

Recently, multi-issue negotiation with nonlinear utility functions has been studied [8]. In this setting, an agent has a nonlinear utility function for considering various constraints. Although this setting is close to the real world negotiation, we do not deal with nonlinear utility functions for multi-issue negotiation in this paper. However, the method for acquiring negotiation strategy for linear utility functions is still required, because the existing methods for linear utility functions are not good enough for win-win negotiation with various opponents as mentioned above. Moreover, the negotiation strategy for linear utility functions can be the basis of the negotiation with nonlinear utility functions.

3. Negotiation Agent

In this section, we define a setting of negotiation. Under this setting we propose a method to make a proposal for win-win negotiation.

3.1. Negotiation Setting

We define a setting of multi-issue negotiation. In this negotiation, two agents make proposals alternately. The number of proposals in a negotiation is limited by deadline. A product is represented as n values $(x_1, ..., x_n)$. Each value is evaluated by an evaluation function $v_i(x_i)$. User's preference of the attributes is denoted as a weight vector $W = \{w_1, w_2, ..., w_n\}$. These weights are normalized as $\sum_{i=1}^{n} w_i = 1$. A utility function U is defined as

$$U = \sum_i w_i v_i(x_i). \tag{1}$$

As preferences of users are various, the users' utilities are different for the same proposal. This difference enables us to realize win-win negotiation.

3.2. Negotiation Agent

We describe actions of an agent such as a proposal, agreement, and breakdown in negotiation. A win-win strategy needs an opponent's preference to reflect preferences of both sides. In an actual negotiation, we do not know the opponent's preference. In this research, however, for focusing on acquisition a win-win strategy, we assume that an agent can estimate the opponent's preference.

3.2.1. Proposal

First, we present how to make a proposal. To realize win-win negotiation, agents require a function for making the better proposal to approach win-win agreement. Therefore, we propose a method for making the proposal by considering opponent's preference. To reflect preferences of both sides, an agent makes the proposal that maximizes the own benefit in the range that the opponent's utility is more than that of the previous own proposal.

The procedure for making a proposal is as follows. Note that we number the proposals of both sides together in a negotiation. That is, if the tth proposal is opponent's, the $(t+1)$th proposal is own proposal. First, an agent finds the range that the opponent's utility is more than that of the previous own proposal with concession α. When the agent makes the $(t+1)$th proposal, the range of the proposal is denoted as

$$U_{t+1}^{opp} - U_{t-1}^{opp} - \alpha^{own} > 0 \tag{2}$$

,where U_{t+1}^{opp} denotes the opponent's utility on the $(t+1)$th own proposal, and $\alpha^{\,own}$ denotes the amount of concession to the opponent. Second, the agent makes the proposal that maximizes the own utility in the range of the equation(2). The agent finds the maximized proposal by the simplex method. Using this method for making a proposal, agents can reach win-win agreement.

3.2.2. Agreement

Next, we describe a condition of an agreement. An agent makes an agreement when the own utility at the $(t-1)$th proposal is more than that at the tth proposal. The own utility of the opponent's last proposal is denoted as U_t^{own}, and the own utility of the own last proposal is represented as U_{t-1}^{own}. Using the utility functions, the condition of an agreement is expressed as

$$U_t^{own} - U_{t-1}^{own} > 0 \tag{3}$$

3.2.3. Breakdown

We define two kinds of breakdown : breakdown by deadline and breakdown by rejection. In breakdown by deadline, negotiation is finished when the number of proposals is more than the deadline. With the tth proposal, the condition of breakdown by deadline is expressed as the equation (4).

$$t > deadline \tag{4}$$

In breakdown by rejection, negotiation is finished when an agent decides to break down the negotiation. The decision is based on the prediction value at the deadline. This prediction uses a linear regression. The negotiation is broken down before deadline when the prediction value is less than the reservation utility of the agent. A reservation utility is a minimum utility to get a benefit. If the utility at the agreement is lower than the reservation utility, the agent loses in this negotiation. With the utility function U, the condition of breakdown by rejection is denoted as

$$U^{own}_{deadline} < RU^{own} \tag{5}$$

,where $U^{own}_{deadline}$ represents the predicted own utility at the deadline. RU^{own} expresses an own reservation utility.

3.3. Preliminary Experiment

We simulated negotiation using the proposed agents for investigating the basic characteristics of the agents. The purpose of this experiment is to verify that the proposed agents can reach win-win agreement. The setting of the simulation is as follows. The number of issues is three, and each issue takes the value from 0 to 5. The evaluation function of an issue for sellers denotes $v^{seller}_i(x_i) = x_i$, and that for buyers denotes $v^{buyer}_i(x_i) = 5 - x_i$. In this function, the evaluation of the product for sellers increases as x_i gets close to {5, 5, 5}, and decreases as x_i gets close to {0, 0, 0}. We call three issues as issue A, issue B and issue C. Each agent makes a concession at a constant rate in this simulation. The first proposal and the preference of sellers and buyers are defined as follows.

- The preference of seller = {0.5, 0.2, 0.3}
- The preference of buyer = {0.3, 0.2, 0.5}
- The first proposal of seller = {4, 4, 4}
- The first proposal of buyer = {1, 1, 1}

Table 1 shows the result of this simulation. In table 1, concession rates of a seller and a buyer are represented as α^{seller} and α^{buyer}, respectively. The utilities of the seller and buyer at agreements are described as U^{seller} and U^{buyer}, respectively.

First, we focus on the negotiation that both agents have the same concession rate 0.2. In this simulation, both agents obtained the utility 3.0. If the agents reach the agreement whose values of all issues are the middle of the first proposals, both of them obtained the utility 2.5. This result shows that both of the agents achieved bigger profits than the middle. That is, the agents could reach win-win agreement. The win-win agreement is achieved by making the proposal considering opponent's preference.

Next, we focus on the negotiation that agents have the different concession rates. Table 1 shows that the larger the difference of concession rates between both parties is, the larger the difference of their utilities at the agreement is. This means that the difference of concession rates causes imbalance of agents' utilities at the agreement. For correcting the imbalance, agents require good concession strategies.

From these results, the proposed agents can reach a win-win agreement if the concession rates of both sides are the same. However, they reach an imbalance agreement if the concession rates of them are different.

Table 1. The result of preliminary experiment

α^{seller}	α^{buyer}	A	B	C	U^{seller}	U^{buyer}
0.2	0.2	5.0	2.5	0.0	3.0	3.0
0.2	0.19	5.0	1.5	0.0	2.8	3.19
0.2	0.1	3.99	0.0	0.0	1.99	3.88

4. Negotiation Strategy Acquisition

Though we can realize win-win negotiation by the agents that have the same concession rate, the benefit of the agent changes largely depending on a concession strategy. In this section, we discuss a method for acquiring concession strategies to deal with various opponents. Human learns negotiation strategies from a lot of experiences. In the same way, an agent can learn from experiments with reinforcement learning.

4.1. Reinforcement Learning

Reinforcement learning refers to a class of problems in machine learning. Many researches for multiagent adopt reinforcement learning[11,12].

Reinforcement learning algorithms attempt to find a policy for maximizing cumulative reward for the agent over the course of the problem. Q-learning is famous reinforcement learning method. In Q-learning, we use a tuple (s, a, s', r), which is a condition s, action a, next condition s', and reward r. An agent learns by updating the action evaluation $Q_t(s, a)$ denoted as

$$Q_{t+1}(s, a) = (1 - \alpha)Q_t(s, a) + \alpha r + \gamma\, Q_t(s', a') \tag{6}$$

,where $\alpha \in (0; 1)$ is a learning rate parameter, and $\gamma \in (0; 1)$ is the discount factor. $Q_t(s, a)$ is denoted as the expected value of the reward given by taking action a in a state s. An agent chooses an action according to $Q_t(s, a)$. Updating $Q_t(s, a)$, the agent can find the optimal action policy.

4.2. Learning Concession Strategy

A negotiation agent learns an action based on a reward of an agreement proposal. For choosing the next actions, we adopt ϵ-greedy strategy. The goal of agents is to learn an optimal concession strategy for leading a good agreement. For this learning, states, action, and reward should be defined appropriately.

State A current state in a negotiation is represented as a negotiation stage and concession balance. The negotiation stage means how close both of proposals are. This is expressed as the difference of the utilities in a current proposal. With an utility function U, the difference of the utilities is represented as

$$NegotiationStage = U_t^{own} - U_{t-1}^{own}. \tag{7}$$

The concession balance is expressed as the difference of the total concession of both sides. Considering the concession balance, the agent can recognize which agent makes more concession. With the amount of concession α_i, the concession balance is denoted as

$$ConcessionBalance = \sum_{i=1}^{t} \alpha_i^{own} - \sum_{i=1}^{t} \alpha_i^{opp}. \quad (8)$$

In equations (7),(8), t is the last opponent's proposal , and $t-1$ is the last own proposal. The total own concession is $\sum_{i=1}^{t-1} \alpha_i^{own}$. The total opponent's concession is $\sum_{i=1}^{t-1} \alpha_i^{opp}$. With the two-dimensional state space, the agent can recognize a negotiation state.

Action An action of an agent is to decide a concession rate : how much the agent makes a concession to the opponent in the next proposal. The amount of concession is defined as

$$\alpha = \beta(U_{MAX}^{opp} - U_t^{opp}) \quad (9)$$

,where U_{MAX}^{opp} denotes maximum utility of opponents, and U_t^{opp} denotes the opponent's utility on the last own proposal. β indicates how much an agent bring the own proposal close to the opponent's proposal. This rate is expressed as a discrete value from one percent to ten percent. Learning the optimal concession leads to acquire a good action policy.

Reward In reinforcement learning, we should set a reward of an agent properly to acquire better actions. In this research, an agent acquires positive rewards when they reach an agreement, and acquire negative reward when the negotiation breaks down.

An agent obtains a reward defined as the equation (10) when the agents reach an agreement,

$$Reward = U_t^{own} - RU^{own} \quad (10)$$

,where U_t^{own} is the own utility on the agreement proposal. When the own utility is higher than the reservation utility RU^{own}, the agent gets a positive reward. On the other hand, when an own utility is lower than the reservation utility RU^{own}, the agent gets a negative reward. A negotiation breakdown gives an agent a negative reward.

4.3. Flow of Learning

The flow of learning in negotiation is as follows. In this learning, $s1, s2$ represent a negotiation stage and a concession balance, respectively.

step1. An agent receives opponent's proposal and observes the state $(s1, s2)$.

step2. According to ϵ-greedy selection, the agent chooses an action a to make a next proposal to the opponent.

step3. When the agents reach an agreement, the agent receives a reward according to the equation (10). A negotiation breakdown gives the agent the reward $r = -1$.

step4. The agent observes a next state $(s1', s2')$.
step5. According to the equation (6), $Q(s1, s2, a)$ is updated.
step6. If the negotiation reaches an agreement or break down, the agent resets the state and returns to step 1, else the agent returns to step 2.

5. Experiment

In this section, we report experiments of negotiation simulation using agents. We conducted four types of experiments with various setting on a negotiation environment and opponents.

5.1. Experimental Setting

The basic setting is as follows.

- Bilateral negotiation
- Agents make a proposal alternately.
- The number of issues is three in experiment 1, 2-1, 2-2, 3, 4. The number of issues in experiment 2-3 is ten.
- The first proposal of an own agent is {4, 4, 4}.
- The first proposal of an opponent is {1, 1, 1 }.
- The evaluation function of oneself is denoted as $v_i^{own}(x_i) = x_i$, and that for the opponent is denoted as $v_i^{opponent}(x_i) = 5 - x_i$.
- The reservation utility of both agents is 3.0.
- The learning rate α is 0.9, and the discounting rate γ is 0.9.

Since most of negotiations can be transformed into this setting by normalizing the values of the issues, the results of this setting can be applied to most of negotiations. Though the number of issues is three in most of the experiments, agents deal with ten issues in experiment 2-3 for verifying that the learning agent can acquire negotiation strategy in more complex environments. However, this setting has the limitation that utility functions of agents are linear.

A learning agent makes a next proposal based on the concession rate acquired by learning. The concession rates of agents and the preference of both agents are different in each experiment. In the experiments, an opponent agent makes a proposal to maximize only its utility. So we call this the selfish agent. The opponent agents have three types of strategies. Figure 1 shows the outline of concession strategies of agents. The vertical axis indicates the change of the total concession, the horizontal axis indicates the number of proposals.

First, we describe a High-Low strategy. The feature of this strategy is that an agent makes a large concession in the early stage of the negotiation, and gradually decreases the amount of concession. The agent decides the amount of concession α by equation (11).

$$\alpha = \beta(U_{MAX} - U_t^{opp}) \tag{11}$$

,where U_{MAX} is the maximum of utilities. U_t^{opp} is opponent's utility on the own proposal. That is, U_t^{opp} indicates the total concession that an agent makes to an opponent.

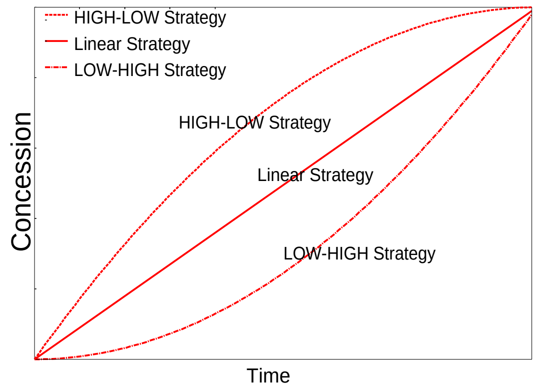

Figure 1. three kinds of strategies of an opponent

In the early stage of negotiation, an opponent's utility U_t^{opp} is small. As the negotiation progresses, U_t^{opp} gradually increases. β denotes the speed of negotiation. The feature of this strategy is that the agent can avoid breakdown because of large concession in early stage. However, due to the large concession in early stage, the agent often makes a loss at the agreement.

Next, we describe the linear strategy. This is the simple strategy that makes a constant concession to an opponent. In this strategy, the amount of concession α is determined as following equation.

$$\alpha = \beta \tag{12}$$

, where β denotes the speed of negotiation. The amount of concession α depends on the constant β.

Finally, we describe the Low-High strategy that is contrary of the High-Low strategy. In this strategy, an agent makes a little concession in the early stage of negotiation, and gradually increases the amount of concession. An agent makes a concession using concession rate α.

$$\alpha = \beta U_t^{opp} \tag{13}$$

, where β denotes the speed of negotiation. The feature of this strategy is that opponents tend to make breakdown by rejection because of a little amount of concession at early stage of negotiation. However, a little amount of concession at early stage tends to give an agent higher utility among three strategies.

5.2. Experiment 1

In this experiment, we focus on how an agent learns behavior without breakdown. An opponent agent has the linear strategy that chooses a random concession at the proposal. An agent learns 50000 times. In this experiment, the preference of the agent is $\{0.5, 0.2, 0.3\}$. The preference of the opponent is $\{0.3, 0.2, 0.5\}$.

Figure 2 shows the result of this experiment. The vertical axis indicates the difference of the both agents' utility at the agreement ((the utility of the agent) - (the utility of

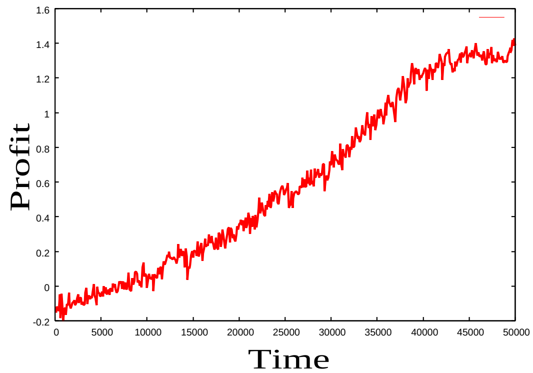

Figure 2. Difference of both agents' utilities (Experiment 1-1)

the opponent)). The difference of the utilities is averaged in every one hundred learning. The horizontal axis indicates the number of learning.

In figure 2, at the early stage of learning the agent made a loss. However, as learning advanced, the agent gradually acquired the action that can increase the own utility. Finally, the agent learned the action that makes the own utility higher than the utility of the opponent. As a result, though the agent can learn to increase the own utility, the opponent makes a loss largely at the agreement. This means that the agent simply reinforced the action that makes little concession to increase the own utility because the opponent does not make the break down.

This experimental result showed that our agent can increase its profit in the no breakdown setting. However, the negotiation strategy acquired by learning is not good because it is simply little concession strategy.

5.3. Experiment 2

This experiment demonstrates the effect of breakdown to learning. In the experiment 2-1, agents use only breakdown by deadline. In the experiment 2-2, agents use breakdown by rejection. In the experiment 2-1 and experiment 2-2, we use the linear strategy agent that makes random concessions. The preferences of an agent and an opponent are {0.5, 0.2, 0.3} and {0.3, 0.2, 0.5}, respectively. In the experiment 2-3, we use the opponent agent that has three strategies, linear strategy, High-Low strategy, and Low-High strategy. The opponent chooses one of three strategies randomly, and the agents decide their preferences randomly when negotiation starts.

5.3.1. Experiment 2-1

Here we show the result in the case of applying breakdown by deadline. Figure 3 shows the learning result without breakdown and that with breakdown by deadline. The vertical axis indicates the difference of the both agents' utilities at the agreement. The difference of the utilities is averaged in every one hundred learning. The horizontal axis indicates the number of learning. Figure 4 shows the change of the total number of breakdown in one hundred learning.

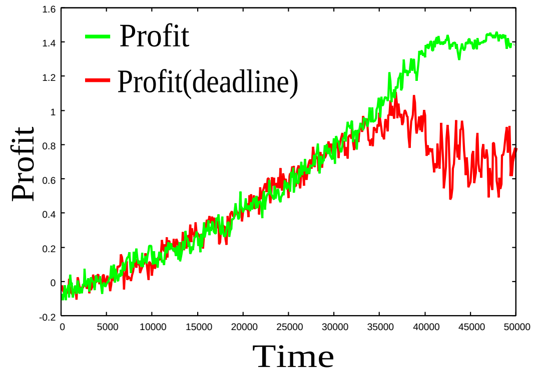

Figure 3. Difference of both agents' utilities , (Experiment 2-1)

At the early stage of learning, the agent reinforces the action making a little concession to increase the own utility due to small number of breakdown. The number of breakdown increases from the middle stage to the late stage of learning. This means that little concession causes breakdown by deadline. From the middle stage to the late stage of learning, the profit of the agent did not increase as the number of the breakdown increased. The negative rewards at breakdown revise the actions of the agent. As a result, the agent reinforces the action making a large concession to avoid breakdown. Compared to a learning agent without breakdown, the learning agent with breakdown by deadline makes the difference of both agents' utilities small.

However, a problem in this result is that the number of breakdown does not decrease after learning. Receiving the negative reward due to the breakdown by deadline, the agent revises the action that avoids the breakdown. However, it is difficult to revise the action for avoiding breakdown at the early stage of negotiation. Since the number of breakdown does not decrease, learning with breakdown by deadline is not sufficient to realize the win-win negotiation.

From this result, our agent can acquire the strategy that increases its profit in the setting with breakdown by deadline. The strategy is not simply little concession in order to avoid breakdown. The fact proved that our agent could acquire the strategy that strikes a balance between cooperation and competition. For acquiring the strategy, breakdown plays important roles. However, breakdown by deadline cannot effectively revise the strategy, because the agent could not decrease the number of breakdown.

5.3.2. Experiment 2-2: Breakdown by Rejection

The result of using breakdown by rejection is presented here. In figure 5, Profit(deadline) represents the result of using breakdown by deadline and Profit(breakdown) represents the result of using breakdown by rejection. The vertical axis indicates the difference of the both agents' utilities at the agreement. The difference of the utilities is averaged in every one hundred learning. The horizontal axis indicates the number of learning. Figure 6 shows the change of the total number of breakdown in every one hundred learning.

Due to the small number of breakdown at the early stage of learning, the agent reinforced the action that makes a little concession to increase the own utility. At the middle stage of learning, the number of breakdown increases because of reinforcing the

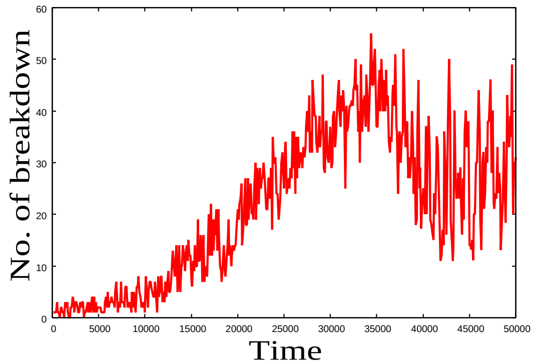

Figure 4. No. of breakdown (Experiment 2-1)

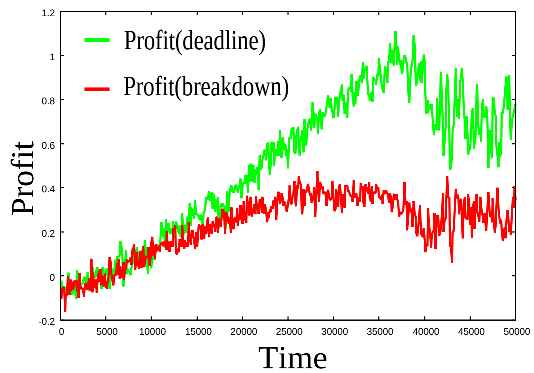

Figure 5. Difference of both agents' utilities (Experiment 2-2)

little concession actions. From the middle to the late stage of learning, the number of breakdown decreases, because breakdown revises the action of the agent more effectively than breakdown by deadline.

Figure 5 shows that breakdown by rejection reduces the difference of agents' utilities. Negotiation generally needs a balance between competition and cooperation. Competition is the action that increases the own utility and cooperation is the action that makes concessions for an agreement. In this experiment, giving the positive reward at agreement and the negative reward at breakdown, the agent can obtain the policy for deciding the concession rate according to the opponent. As a result, the agent brought a balance between competition and cooperation, and learned the action policy that makes a win-win agreement.

This result showed that our agent could also acquire the strategy that increases its profit in the setting with breakdown by rejection. Our agent can acquire the strategy that strikes a balance between cooperation and competition. Compared to the setting with breakdown by deadline, the agent can decrease the number of breakdown. This is because breakdown before deadline can revise the competitive behavior in the early stage of the negotiation.

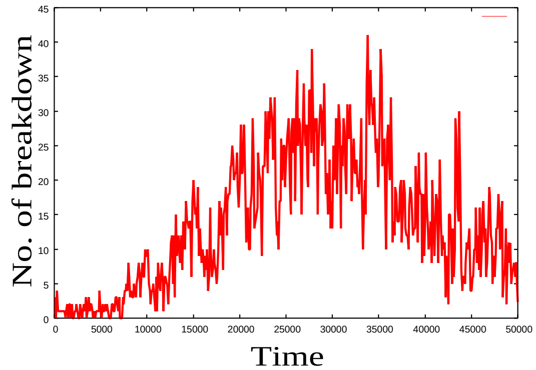

Figure 6. No. of breakdown (Experiment 2-2)

5.3.3. *Experiment 2-3*

Here we perform an experiment in the setting that opponents have various strategies and preferences. Moreover, the number of issues is ten in this experiment. The reason for increasing the number of issues is to prove that our agent can realize win-win negotiation even if the number of issues is comparative large. An opponent randomly chooses one of three strategies (linear strategy, High-Low strategy, Low-High strategy) in each negotiation. An opponent agent randomize a value of parameter β in the equation (12), (13) and (14). To make negotiations more complex, an agent and an opponent have random preferences in each negotiation.

Figure 7 shows the change of the profit difference between the agents. In figure 7, the vertical axis indicates the difference of the both agents' utilities at the agreement. The difference of utilities is averaged in every one hundred learning. The horizontal axis indicates the number of learning. Figure 8 shows the change of the total number of breakdown in every one hundred learning.

In Figure 8, the number of breakdown increases from the early stage to the middle stage, and decreases from the middle stage to the late stage in learning as well as in the case of figure 6. In Figure 7, by increasing the number of breakdown, an agent revises the action to reduce the difference of both agents' utilities.

This result shows that the agent can deal with the opponent that has various preferences and strategies. In addition, the agent can acquire good negotiation strategy even if the number of issues in the negotiation is comparative large. This is because the proposed method for making a proposal adequately deals with multi-issue.

5.4. *Experiment 3*

In this experiment, we show how an agent learns actions against a learning agent. Figure 9 and Figure 10 show the results of this experiment. Figure 9 shows the difference of utilities at the agreement. In Figure 9, the vertical axis indicates the difference of the both agents' utilities at the agreement. The difference of utilities is averaged in every one hundred learning. The horizontal axis indicates the number of learning. Figure 10 shows the change of the total number of breakdown in every one hundred learning.

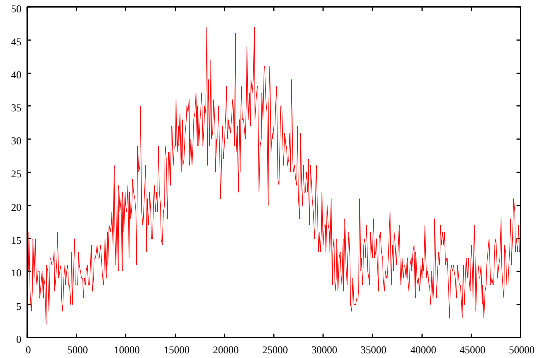

Figure 7. Difference of both agents' utilities (Experiment 2-3)

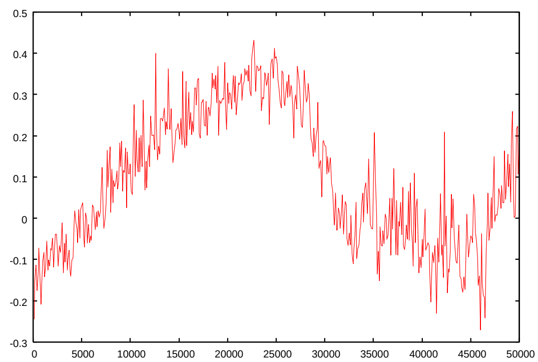

Figure 8. No. of breakdown (Experiment 2-3)

In the early stage of learning, both of agents learn to make a little concession because of small number of breakdown. From the early stage to the middle stage of learning, the number of breakdown increases, because each agent makes the breakdown by rejection to the other agent that makes a little concession. Receiving the negative rewards by breakdown, each agent revises the action. As a result, from the middle stage to late stage of learning, the number of breakdown decreases. Finally the agent learns an action policy to get a good agreement. The difference of both agents' utilities at the agreement was less than 0.1. From this result, using our method for learning a concession policy, we can make negotiations more effective.

5.5. Experiment 4

We compare our method with the existing methods. In this experiment, a learning agent estimates an opponent's preference based on the k-Nearest Neighbor algorithm. This algorithm estimates the preference as the average of k nearest instances. An instance consists of the first three proposal of the opponent. Using this estimation, the agent negotiates with an opponent after learning.

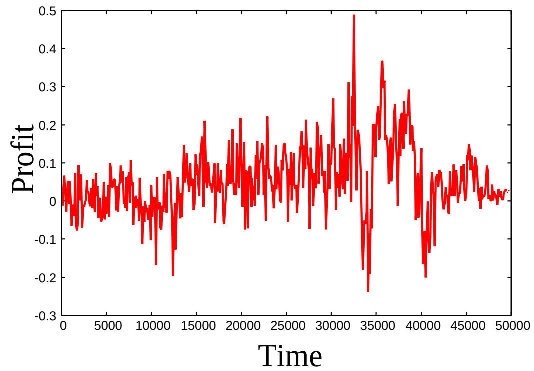

Figure 9. Difference of both agents' utilities (Experiment 3)

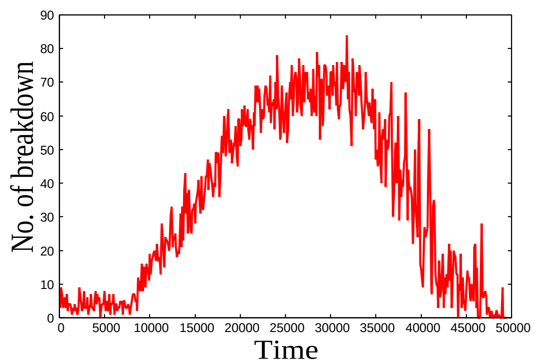

Figure 10. No. of breakdown (Experiment 3)

In this experiment, we compare our method with Coehoorn's method[4]. We call this method KDE. Since the original KDE method make no concession in its utility, most of negotiations in the setting of this paper cause breakdown. So we add a concession function to KDE. We use two type of the KDE method. KDE(1) makes a little concession 0.1 to the opponent, and KDE(2) makes a comparative large concession 0.5.

Table 2 shows the results of this experiment. The value in this table showed the utility at agreement averaged in 100 negotiations in various preference setting. The former value denotes the utility of the left side, and the latter value represents the utility of the above. The value in the parenthesis represents the number of breakdown.

The proposed agent achieved more utility than the opponent except KDE(1). The number of the breakdown of the proposed agent is comparative small. Since KDE(1) makes a little concession to the opponent, it can obtain more utility than the opponent if the negotiation reaches agreement. However, most of the negotiation of this agent cause breakdown. So the strategy of the KDE(1) is not suitable for win-win negotiation. From these results, we can verify that the proposed agent achieved better negotiation than the other agents.

Table 2. The compared result of the experiment

	Proposed agent	Selfish	KDE(1)	KDE(2)
Proposed agent	-	3.42:2.88 (1)	2.76:3.80 (87)	3.71:2.36 (43)
Selfish	-	-	2.78:3.82 (93)	2.86:2.44 (0)
KDE (1)	-	-	-	3.86:2.55 (36)

6. Conclusion

In this paper, we presented a learning method of concession strategy for adapting to various opponents for win-win negotiation. For realizing win-win negotiation, we propose a method for making the proposal to reach win-win agreement by considering opponent's preference. The simulation results using the proposed agents showed that this method enables win-win agreement when both of the agents have the same concession rate. However, the difference of the concession rates causes imbalance results.

In order to redress the imbalance, we propose a method for learning concession strategy. In this method, we use reinforcement learning for acquiring concession strategy. This method decides the concession rate based on the negotiation state using the difference between their proposals and the difference between their concessions. The experimental results showed that by using reinforcement learning the agent could acquire the strategies that can manage opponents with various concession strategies and preferences. If an opponent agent can make negotiation breakdown, the learning agent can revise a competitive action and acquire a cooperative action. When both agents learn their concession strategies, they could finally get almost the same utilities. The proposed agent can learn negotiation strategy even if the number of issues is comparative large. Compared to the agent with no learning, the learning agent could realize the effective negotiation that strikes a balance between cooperation and competition. In the experiment, our agent competed with a selfish agent and the KDE agent. This result showed that our agent reached better agreement than the others.

For the future, additional work is needed on technique that can learn the strategy in multi-issue negotiation with nonlinear utility functions. Another future work is to estimate opponent's preference. In the experiment of this paper, the agent used the k-NN approach for estimating opponent's preference. We should refine the estimating method for acquiring good negotiation strategy. One way for estimation is to acquire opponent's preference and concession strategy simultaneously by reinforcement learning.

References

[1] Tibor Bosse, Catholijn M. Jonker, Jan Treur: *Experiments in Human Multi-Issue Negotiation: Analysis and Support*. Proc. of Third International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS-2004, pp671-678, (2004).

[2] Longbing Cao, Chengqi Zhang, Jiming Liu: *Ontology-based integration of business intelligence*, Web Intelligence and Agent Systems, Vol.4, No.3, pp.313-325 (2006).

[3] Catholijn M. Jonker, Valentin Robu: *Automated Multi-Attribute Negotiation with Efficient Use of Incomplete Preference Information.* Proc. of Third International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS-2004, pp1056-1063, (2004).

[4] R.M.Coehoorn and N.R. Jennings: *Learning an opponent's preferences to make effective multi-issue negotiation tradeoffs* Proc. of 6th International Conference on E-Commerce, pp.59-68, (2004).

[5] S.S.Fatima, M.J.Wooldridge and N.R.Jennings: *Multi-issue negotiation under time constraints*. Proc. of First International Joint Conference on Autonomous Agents and Multi-agent systems, AAMAS-2002,pp143-150,(2002).

[6] S.S.Fatima, M.J.Wooldridge and N.R.Jennings: *An agenda-based framework for multi-issue negotiation*. Artificial Intelligence, Vol. 152, Issue 1 ,pp1-45, (2004).

[7] Pu Huang and Katia Sycara: *A Computational Model For Online Agent Negotiation*. Proc. of 35th Hawaii International Conference on System Sciences, HICSS-2002,(2002).

[8] Takayuki Ito, Mark Klein, Hiromitsu Hattori: *Multi-issue Negotiation Protocol for Agents: Exploring Nonlinear Utility Spaces*. Proc. of the Twentieth International Joint Conference on Artificial Intelligence, pp.25-35 (2007).

[9] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C. and Wooldridge, M.: *Automated negotiation: prospects, methods and challenges*, International Journal of Group Decision and Negotiation, Vol.10, No.2, pp199-215 (2001).

[10] Junzhong Ji, Chunnian Liu, Jing Yan, Ning Zhong: *An improved Bayesian network structure learning algorithm and its application in an intelligent B2C portal*, Web Intelligence and Agent Systems, Vol.5, No.2, pp.127-138 (2007).

[11] Ville Kononen: *Asymmetric Multiagent Reinforcement Learning*, Web Intelligence and Agent Systems, Vol.2, No.2 pp.105-136 (2004).

[12] Dehu Qi and Ron Sun: *A Multi-Agent System Integrating Reinforcement Learning, Bidding and Genetic Algorithms*, Web Intelligence and Agent Systems, Vol.1, No.3, pp.187-202 (2003).

[13] Raymond Y. K. Lau, Eivind Valdal: *Discovering Negotiation Knowledge for a Probabilistic Negotiation Web Service in e-Business*. Proc. of Web Intelligence 2004, pp264-270, (2004).

[14] Raymond Y. K. Lau: *Towards Genetically Optimised Multi-Agent Multi-Issue Negotiations*. Proc. of Annual Hawaii International Conference on System Sciences, p.35c, (2005).

[15] G. Lai, K. Sycara, C. Li: *A decentralized model for multi-attribute negotiations*. Proc. of the 8th international conference on Electronic commerce, pp3-10, (2006).

[16] C. Li, J.A. Giampapa, and K. Sycara: *Bilateral Negotiation Decisions with Uncertain Dynamic Outside Options*. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Special Issue on Game-theoretic Analysis and Stochastic Simulation of Negotiation Agents, Vol. 36, No. 1, pp31-44, (2006).

[17] Shavit Talman, Meirav Hadad, Ya'akov Gal, Sarit Kraus: *Adapting to Agents' Personalities for negotiation*. Proc. of the 4th International Joint Conference on Automonous Agent and Multiagent systems, pp.383–389, (2006).

[18] Edward Tsang and Tim Gosling: *Simple Constrained Bargaining Game*. Proc. of First International Joint Conference on Autonomous Agents and Multi-Agent Systems, AAMAS-2002, (2002).

[19] Sheng Zhang, Song Ye, Fillia Makedon, James Ford: *A hybrid negotiation strategy mechanism in an automated negotiation system*. Proc. of the 5th ACM Conference on Electronic Commerce, pp.256-257 (2004).