Meta-Strategy Based on Multi-Armed Bandit Approach for Multi-Time Negotiation*

Ryohei KAWATA^{†a)}, Nonmember and Katsuhide FUJITA^{†b)}, Member

SUMMARY Multi-time negotiation which repeats negotiations many times under the same conditions is an important class of automated negotiation. We propose a meta-strategy that selects an agent's individual negotiation strategy for multi-time negotiation. Because the performance of the negotiating agents depends on situational parameters, such as the negotiation domains and the opponents, a suitable and effective individual strategy should be selected according to the negotiation situation. However, most existing agents negotiate based on only one negotiation policy: one bidding strategy, one acceptance strategy, and one opponent modeling method. Although the existing agents effectively negotiate in most situations, they do not work well in particular situations and their utilities are decreased. The proposed meta-strategy provides an effective negotiation strategy for the situation at the beginning of the negotiation. We model the meta-strategy as a multi-armed bandit problem that regards an individual negotiation strategy as a slot machine and utility of the agent as a reward. We implement the meta-strategy as the negotiating agents that use existing effective agents as the individual strategies. The experimental results demonstrate the effectiveness of our meta-strategy under various negotiation conditions. Additionally, the results indicate that the individual utilities of negotiating agents are influenced by the opponents' strategies, the profiles of the opponent and its own profiles.

key words: automated negotiation, meta-strategy, multi-time negotiation, multi-armed bandit problem, multi-agent system

1. Introduction

Automated negotiation is an important interaction for autonomous agents ([2], [3] etc.). Negotiating agents resolve conflicts, which enables them to cooperate with each other. Motivated by the challenges of automated negotiation, the Automated Negotiating Agents Competition (ANAC) was first organized in 2010 to facilitate research in automated multi-issue closed negotiation ([4] etc.). The ANAC setup is based on realistic models that include time discounting, closed negotiation, and alternative offering protocols. By analyzing the results of ANAC, the trends of the automated negotiating agents' strategies and critical factors for developing competition have been shown [5]. Many effective automated negotiating agents have also been proposed through competitions ([6], [7]). In multi-time negotiation, negotiations are repeated under the same conditions: the same agents, domain and preferences. Therefore, it is important

for agents to proceed the negotiations effectively by utilizing the negotiation history.

The performance of a negotiating agent depends strongly on the negotiation situation, which includes factors such as the opponent, domain, and profiles ([5] etc.). Most existing agents negotiate based on one policy: one bidding strategy, one acceptance strategy, and one method of opponent modeling. These agents negotiate effectively in most situations. However, there are some cases where their performances are poor. In other words, there is no single negotiation strategy that clearly outperforms all other strategies in all situations. Therefore, agents should select a suitable strategy to reach a beneficial agreement. In multi-time negotiation, agents can utilize its own negotiation history to select an effective strategy, accordingly.

In this paper, we propose a meta-strategy, which is used to select an agent's negotiation strategy according to the situation for multi-time negotiation. A meta-strategy is defined as an overarching strategy, determining which individual strategies to use in a given negotiation situation. In addition, an individual strategy is defined as the elements to compose of the meta-strategy. Our meta-strategy selects a negotiation strategy from several individual strategies before starting each negotiation. The selection of the individual strategy has been modeled as a multi-armed bandit problem [8] by regarding an individual strategy as a slot machine and its own utility as a reward. Our meta-strategy selects the individual strategy with the highest average utility based on its use in past negotiations. However, our meta-strategy sometimes selects another strategy to obtain more information based on the multi-armed bandit algorithm.

We implement the meta-strategy as the negotiating agents using existing agents as individual strategies. The experimental results demonstrate the effectiveness of our meta-strategy in various negotiation scenarios. In addition, the result of the tournament between our agent and state-of-the-art agents demonstrate that our agent outperforms these agents with respect to the individual utility.

The remainder of this paper is organized in the following manner. First, we denote the problem settings and describe multi-armed bandit problems and its algorithms. Next, we propose our meta-strategy and the agents that implement it. Then, we show the experimental setup and results. Finally, we discuss related work in automated negotiation and present our conclusions.

Manuscript received February 10, 2020.

Manuscript revised July 1, 2020.

Manuscript publicized September 7, 2020.

[†]The authors are with Department of Computer and Information Sciences, Graduate School of Engineering, Tokyo University of Agriculture and Technology, Koganei-shi, 184–8588 Japan.

*This is an extended version of work published in [1].

a) E-mail: kawata@katfujiiab.tuat.ac.jp

b) E-mail: katfujii@cc.tuat.ac.jp

DOI: 10.1587/transinf.2020SAP0003

2. Problem Setting

2.1 Negotiation Environment

We assume a bilateral multi-issue negotiation that two agents negotiate about a set of issues; the two agents aim to increase its own utility. In multi-issue negotiation, every *issue* has a discrete *value* that represents the options. A *negotiation domain* defines the set of issues and possible value of each issue. A set of values picked for each issue is referred to as an *outcome*. Each agent proposes a *bid* or an *offer* during a negotiation. The outcome can also be regarded as a bid or an offer during a negotiation. Ω is a set of all possible outcomes, and it is common knowledge shared among all agents in a negotiation session.

Every agent has a unique *profile* which represents its own preferences for each outcome $\omega \in \Omega$. A profile consists of a set of weights for each issue and a set of evaluation values for each value. A *utility* of an outcome that is defined as a weighted-sum of evaluation values which is normalized to a real number in the range $[0, 1]$. Let n be the number of issues and w_i ($\sum_{i=1}^n w_i = 1.0$) be the *weight* of each issue i . The utility function U of outcome ω is

$$U(\omega) = \sum_{i=1}^n w_i \cdot eval(\omega_i)$$

where ω_i is the selected value of issue i and $eval(\omega_i)$ is the *evaluation value* of ω_i normalized to the range $[0, 1]$. w_i and $eval(\cdot)$ are defined in a profile, and they are not shared with other agents.

A negotiation session has a timeline t which is represented as a real number in the range $[0, 1]$. $t = 0$ refers to the starting time of a negotiation and $t = 1$ represents the time when the deadline is reached. Each agent has a *discount factor* δ ($0 < \delta < 1$) which decreases the agent's utility depending on t [9]. Discounted utility U_D of outcome ω is

$$U_D(\omega, t) = U(\omega) \cdot \delta^t$$

$\delta = 1$ indicates that the utility is independent of t , and $\delta < 1$ means that the utility decreases as t increases. A *reservation value* is a utility which an agent receives when a negotiation complete with no agreement being reached. Let r be the reservation value, the received utility in failing negotiations is $r \cdot \delta^t$, where t is the time that the negotiation ended. $t < 1$ means the negotiation broke down, and $t = 1$ represents a timeout. δ and r are defined in a profile, they are private information.

We focus on multi-time negotiation which is repeatedly conducted under the same condition. In multi-time negotiation, an agent repeatedly negotiates with the same opponent and the same domain under the same profile. An agent can keep some important logs during the negotiations to use them in the next negotiations.

2.2 Alternating Offers Protocol (AOP)

The interaction between negotiating parties is regulated by a *negotiation protocol* that defines the rules of how and when proposals can be exchanged. The Alternating Offers Protocol (AOP) is a negotiation protocol for bilateral negotiation [10]. In AOP, the negotiating parties exchange offers in turns. It is widely studied in the literature, both in game theoretic and heuristic settings of negotiation.

For example, *Agents A and B* take turns in the negotiation. One of the two agents is picked at random to start. When it is the agent's turn X (X being A or B), that agent is informed about the action taken by the opponent. In the negotiation, the two parties take turns in selecting the next negotiation action. The possible actions are:

- *Accept*: It indicates that the agent accepts the opponent's last bid.
- *Offer*: It indicates that the agent proposes a new bid.
- *End Negotiation*: It indicates that the agent terminates the entire negotiation, resulting in the lowest possible score for both agents.

If the action was an *Offer*, agent X is subsequently asked to determine its next action and the turn taking goes to the next round. If it is not an *Offer*, the negotiation has finished. The final score (utility of the last bid) is determined for each of the agents, as follows:

- *Agreement*: An offer ω is accepted by the opponent at time t . Each agent obtains $U_D(\omega, t)$.
- *Failure*: The action is returned as an *EndNegotiation* at time t . Each agent obtains $r \cdot \delta^t$.
- *Timeout*: A negotiation cannot be finished by the deadline ($t = 1$). Each agent obtains $r \cdot \delta$.

In AOP, it is recognized that the agent continues to offer without accepting the opponent's offers in order to consume remaining time on purpose to reach the timeout. However, the failure of agreements is the worst outcome, therefore, each agent tries to make an agreement by making a concession to the opponent's offers. In other words, the usual negotiation strategies can be either a conceder if the agent is willing to concede a lot in the early phase of negotiation, or a brawler if a player is willing to concede considerably only when its time deadline is approaching [11].

3. Multi-Armed Bandit Problem

Multi-armed bandit problems are a popular model of sequential decision problems with an exploration/exploitation trade-off [8]. In multi-armed bandit problems, the gambler faces K slot machines (which are also referred to armed bandits) and plays them T times. Each machine provides random rewards based on its own distribution. The gambler has no knowledge about the distributions. Therefore, the gambler chooses the next machine to play based only on the results he/she obtained while playing. The objective of the

gambler is to maximize the sum of rewards. Multi-armed bandit problems apply not only to casino gambling but also various optimization problems with the trade-off of exploration and exploitation, such as channel access schemes in wireless sensor networks [12] and the allocation of treatment in a clinical trial [13].

3.1 ϵ -greedy Algorithm

ϵ -greedy algorithm is a well-known algorithm for multi-armed bandit problems because of its simplicity. In each round, the algorithm takes an exploration with probability ϵ , and takes an exploitation with probability $1 - \epsilon$. The exploration is the selection of a machine at random, and each machine is selected with probability $1/K$. The exploitation is to select the machine with the highest average of rewards in the past trials. $\epsilon = 0$ is the same as the greedy algorithm that does not take exploration. $\epsilon = 1$ is the algorithm that always chooses a machine randomly without exploitation.

3.2 Upper Confidence Bounds Algorithm

The Upper Confidence Bound (UCB) algorithm is a well-known algorithm to resolve multi-armed bandit problem [14]. Initially, the algorithm selects every machine once. After that, the algorithm calculates the UCB score of each machine and selects the machine with the highest UCB score at every selection. Let S be a set of slot machines, N is the total number of trials, and N_s is the number of trials of slot machine $s \in S$, the UCB score of slot machine $s \in S$ is

$$UCB(s) = \hat{\mu}_s + c \sqrt{\frac{\ln N}{N_s}}$$

where $\hat{\mu}$ is the average reward of s in past trials and $c > 0$ is a parameter that controls the frequency of exploration. The greater the value of c , the greater the correction term ($c \sqrt{\frac{\ln N}{N_s}}$); this means that the algorithm gives a weight to exploration for machines whose average reward is low.

4. Meta-Strategy

4.1 Modelling of Meta-Strategy

We propose a meta-strategy for multi-time negotiation. In this paper, a meta-strategy is defined as an overarching strategy, determining which individual strategies to use in a given negotiation situation. In addition, an individual (negotiation) strategy is defined as the elements to consist of the meta-strategy. Our meta-strategy includes some individual negotiation strategies and selects one from them as an agent's negotiation strategy at the beginning of the negotiation. We modeled the selection of individual negotiation strategy in our meta-strategy as a multi-armed bandit problem[†].

[†]Although we focus on the bilateral negotiation in this paper, our meta-strategy can be applied to multi-party negotiation.

Fig. 1 Overview of our meta-strategy model

The effectiveness of a negotiating agent is influenced by the opponent, the domain, and the profiles. In other words, the negotiation strategy that outperforms clearly all other agents in all negotiation scenarios is none. To reach an effective agreement, an agent needs to take the most suitable strategy for the situation. However, it is not easy to estimate the performance of a negotiation strategy before the negotiation because the opponent's strategy and the profile are not open. After the negotiation, the performance of each negotiation strategy is clearer with the advantage of hindsight about the negotiation result. In multi-time negotiation, we estimate the performance of a strategy on the setting using the history of negotiations. These conditions are similar to those in the multi-armed bandit problem. From the above, we apply the framework of multi-armed bandit problems to the strategy selection of our meta-strategy.

The details of the modeling of our meta-strategy are as follows. The strategy selection in the meta-strategy for multi-time negotiation is considered to be the same as the machine selection of the gambler in the multi-armed bandit problem. Hence, each individual negotiation strategy can be considered as a slot machine. A machine provides a reward to the gambler from its distribution. A negotiation strategy provides its own utility to the agent after the negotiation. This utility can be regarded as the reward of a machine. The agent tries to maximize the total utility just as the gambler tries to maximize the total reward. Figure 1 shows the overview of applying the framework of multi-armed bandit problems to the strategy selection in our meta-strategy. The slot machines are considered as the individual negotiation strategies, and the reward is considered as its own utility in the negotiation.

Our meta-strategy switches agent's negotiation strategy in order to suit the situation, such as the opponent, the domain and the profiles by resolving multi-armed bandit problems. We apply existing algorithms for multi-armed bandit problems to the strategy selection of our meta-strategy.

4.2 Implementation of the Negotiating Agents

We implement agents which use our meta-strategy. Our agent selects a negotiation strategy from existing negotiation strategies using the multi-armed bandit algorithm at the beginning of the negotiation. Initially, our agent selects every machine once for the first reward in this implementation. After that, our agent selects a machine based on the algorithm for each. We implemented eight agents with different algorithms and parameters:

- *UCB(1)*: UCB algorithm with $c = 1$
- *UCB(0.5)*: UCB algorithm with $c = 0.5$
- *UCB(0.1)*: UCB algorithm with $c = 0.1$
- *UCB(0.05)*: UCB algorithm with $c = 0.05$
- *UCB(0.01)*: UCB algorithm with $c = 0.01$
- *EG(0)*: ϵ -greedy with $\epsilon = 0$ (greedy algorithm)
- *EG(0.1)*: ϵ -greedy with $\epsilon = 0.1$
- *EG(0.2)*: ϵ -greedy with $\epsilon = 0.2$
- *EG(1)*: ϵ -greedy with $\epsilon = 1.0$ (random selection)
- *PureMAB*: The implementation of this agent is same as [15].

Our agents select a strategy to negotiate from existing strategies, namely, *Atlas3*, *CaduceusDC16*, *kawaii*, *ParsCat*, *Rubick*, and *YXAgent*. These strategies are outstanding strategies in the individual utility category of ANAC2015 [16], ANAC2016 [17] and ANAC2017 [18]. Each strategy has different characteristics and outperforms other agents in several situations.

Atlas3 is the winner in the individual utility category and Nash product category (it is also called social welfare category) in the ANAC 2015. It uses appropriate searching method based on the similarity between the utility of the bid and the bid with the maximum utility [19]. It generates a relative utility matrix according to the agent's utility space. The relative utility matrix means that the row corresponds to the issue and the column corresponds to the value of the issue. The matrix contains the similarity between the utility of maximum bids and the bid of the agent.

CaduceusDC16 is the second place agent of individual utility category in ANAC2017. This agent asks the expert agents' opinion whether to make a counteroffer or to accept the received offer as well as what to bid. The expert agents are the ANAC agents that succeeded in the past years. Our agents are similar to this agent because both agents use more than two existing agents' strategies properly. However, these are the difference in that our agents select the strategy at the beginning of the negotiation, while *CaduceusDC16* decides the action at every round by a majority vote of the expert agents.

kawaii is the finalist agent of individual utility category in ANAC 2015. This agent changes its own compromising strategy based on opponent's offers. It takes a bullish strategy when the opponent's strategy is a bearish strategy, and vice versa. It and our agents are similar to each other in that both change their own strategy depending on the

opponent's strategy. However, our agents consider not only the opponent's strategy but also other conditions, such as its own profile and the opponent's profile.

ParsCat is the agent tied for the second place in the individual utility category in ANAC 2016. It mirrors changed in agent's behavior by the passage of time. Thus, it is a time-dependent agent. The main idea is that it swings the offers because it may appear flexible taking the offer as the time passes. Therefore, it keeps the offer that has good utility and would use them again.

Rubick is the agent of the third place agent of individual utility category in ANAC 2017. This agent is a complex time based conceder enriched by derivations of well studied heuristics in the automated negotiation field. The main component of the agent is the target utility, which is actually the lower boundary in the bid generation and acceptance condition. If the history is not available yet, target utility is initialized as the utility of the first received bid and updated to the highest utility received from any of the opponent parties. On the other hand, if a negotiation history with the same opponent is detected, it sets the lower bound to be the highest utility ever received throughout the negotiation; thinking that the opponents is designed in a myopic way.

YXAgent is the agent tied for the second place in the individual utility category in ANAC 2016. The acceptance threshold of it is pretty low in comparison to other agents. Therefore on most occasions, *YXAgent* will accept the opponents offer and this greatly reduces the number of bids offered by it. The outcome makes it harder for an opponent to gather information from its bidding technique.

pureMAB: pureMAB agent by Ilany and Gal [15]. They proposed the strategy selection method based on supervised learning using structural features of the negotiation domain. The selection method is based on a bandit approach. It calculates the UCB score for each its own profile, in other words, they do not consider the opponent's strategy and profile in the strategy selection.

Unspecified: This is the same as our proposed Meta-Strategy, however, it can't specify the opponent agent in the tournaments. In other words, this agent considers that the opponent agents are selected from *Atlas3*, *CaduceusDC16*, *kawaii*, *ParsCat*, *Rubick*, and *YXAgent*, randomly. This agent solves the individual multi-armed bandit problem in each profile to the unspecified opponent.

5. Experimental Results

5.1 Settings

The tournaments are run on GENIUS platform (version 9.1.1) [20]. The tournaments consisted of some agents and they depend on each experiment. Each negotiation session consists of two different agents, and it is called bilateral negotiation. All negotiations with every combination of agents and profiles are conducted, and each negotiation repeats 100 times with the same conditions. The negotiation protocol is AOP and the deadline of each negotiation is 10 seconds.

The domain used in the experiments consists of 5 issues, and each issue has 5 values. There are 3,125 outcomes in the domain. The profiles of the domain is generated by the parameters presented in Table 1.

Each setting generates two profiles, and there are 16 profiles in the domain. Every setting has four parameters:

- $eval(v)$: evaluation value $eval(v)$ for each value v
- w : a set of weights for each issue
- δ : discount factor
- r : reservation value

$Beta$ distribution $Beta(\alpha, \beta)$ provides $eval(v)$, and the $Dirichlet$ distribution $Dir(\alpha)$ provides w . The domain includes various kinds of scenarios such as cooperative, competitive, and unfair, as shown in Fig. 2.

5.2 Experimental Results

We conducted eight tournaments to evaluate the performance of our agents. Each tournament consisted of the meta strategy indicating the header row and the individual strategies. Namely, *Atlas3*, *CaduceusDC16*, *kawaii*, *ParsCat*, *Rubick*, and *YXAgent* are as the opponents to the Meta-Strategy. The negotiations between the same negotiation strategies have been omitted.

Table 2 shows the average of the individual utility of each agent. A column describes the results of each tournament which consists of the meta-strategy and existing agents. It demonstrates that our meta-strategy except for *UCB(0.5)* and *UCB(1)* outperform all existing agents. Our

meta-strategy with an appropriate parameter works effectively in multi-time negotiation. As c and ϵ get smaller, the individual utility of our agents become higher. It means that exploring is not important for the strategy selection of our meta-strategy. Agents rarely overestimate or underestimate the performance of negotiation strategy because the variance of individual utilities is small in the negotiations in the same situation. Therefore, agents can find the best strategy for the situation with a small amount of exploration.

Table 3 shows the average social welfare of each agent. The social welfare is the sum of all agents' individual utilities in the negotiation. It demonstrates that our agents have a high social welfare agreement, totally. As c and ϵ get smaller, the social welfare of our agents increase because for the same reason in the case of individual utility.

Comparing with our meta-strategy with pure-MAB, our meta-strategy except for *UCB(0.5)* and *UCB(1)* outperform pure-MAB. This is because that pure-MAB calculates the UCB score for each its own profile, in other words, they do not consider the opponent's strategy and profile in the strategy selection. Our approach calculates the UCB score for each combination of its own profile and the opponent's strategy and profile. Also, the scores of UCB are a little better than the ones of EG. This is because that UCB can explore more efficiently than EG. EG is equally likely to pick either of individual strategies when exploring random arms, on the other hands, UCB can periodically give less-explored arms a chance since the confidence intervals are wider.

Our meta-strategy can work well when the opponent is unspecified considering the results of *Unspecified* in Table 2 and Table 3. The scores of them were almost the same as the specified ones (*Meta-Strategy*). Especially, the difference between *UCB(0.01)* and *EG(0)* (Greedy selection) in the unspecified situation is larger than the ones in the specified situation. It means that the exploration in selecting the individual strategy is more important in the unspecified situation because our meta-strategy often overestimates the performance of the individual strategy by the large variances of the obtained individual utilities.

Figure 3 shows the rates of ranking of the individual

Table 1 Parameter settings for profile generation in the experiments

Name	$eval(v)$	w	δ	r
AAA	$Beta(2, 4)$	$Dir(9, 9, 9, 9, 9)$	1.0	0.5
AAB	$Beta(2, 4)$	$Dir(9, 9, 9, 9, 9)$	0.5	0.5
ABA	$Beta(2, 4)$	$Dir(0.3, 0.3, 0.3, 0.3, 0.3)$	1.0	0.5
ABB	$Beta(2, 4)$	$Dir(0.3, 0.3, 0.3, 0.3, 0.3)$	0.5	0.5
BAA	$Beta(4, 2)$	$Dir(9, 9, 9, 9, 9)$	1.0	0.5
BAB	$Beta(4, 2)$	$Dir(9, 9, 9, 9, 9)$	0.5	0.5
BBA	$Beta(4, 2)$	$Dir(0.3, 0.3, 0.3, 0.3, 0.3)$	1.0	0.5
BBB	$Beta(4, 2)$	$Dir(0.3, 0.3, 0.3, 0.3, 0.3)$	0.5	0.5

(a) Competitive scenario

(b) Cooperative scenario

(c) Unfair scenario

Fig. 2 Scenarios of the domain in the experiments

Table 2 The average of individual utilities of agents (the header row indicates the kind of meta-strategy).

Meta-Strategy is:	<i>UCB(0.01)</i>	<i>UCB(0.05)</i>	<i>UCB(0.1)</i>	<i>UCB(0.5)</i>	<i>UCB(1)</i>	<i>EG(0)</i>	<i>EG(0.1)</i>	<i>EG(0.2)</i>	<i>EG(1)</i>	<i>PureMAB</i>
Meta-Strategy	0.7788	0.7765	0.7725	0.7382	0.7201	0.7787	0.7706	0.7631	0.6979	0.7406
<i>Atlas3</i>	0.7444	0.7443	0.7451	0.7474	0.7477	0.7441	0.7445	0.7451	0.7475	0.7501
<i>CaduceusDC16</i>	0.7138	0.7135	0.7134	0.7119	0.7104	0.7137	0.7130	0.7127	0.7090	0.7133
<i>kawaii</i>	0.7305	0.7304	0.7304	0.7274	0.7254	0.7312	0.7299	0.7293	0.7221	0.7265
<i>ParsCat</i>	0.6867	0.6872	0.6877	0.6875	0.6869	0.6869	0.6862	0.6862	0.6863	0.6881
<i>Rubick</i>	0.6658	0.6664	0.6652	0.6652	0.6648	0.6658	0.6654	0.6646	0.6644	0.6717
<i>YXAgent</i>	0.7132	0.7129	0.7121	0.7050	0.7006	0.7130	0.7110	0.7097	0.6944	0.6994
Unspecified	0.7536	0.7464	0.7440	0.7177	0.7035	0.7420	0.7391	0.7337	0.6788	0.7183

Table 3 The average of social welfare of agents (the header row indicates the kind of meta-strategy).

Meta-Strategy is:	<i>UCB(0.01)</i>	<i>UCB(0.05)</i>	<i>UCB(0.1)</i>	<i>UCB(0.5)</i>	<i>UCB(1)</i>	<i>EG(0)</i>	<i>EG(0.1)</i>	<i>EG(0.2)</i>	<i>EG(1)</i>	<i>PureMAB</i>
Meta-Strategy	1.5108	1.5088	1.5042	1.4602	1.4335	1.5110	1.4983	1.4883	1.3992	1.4673
<i>Atlas3</i>	1.5095	1.5100	1.5095	1.5065	1.5048	1.5101	1.5094	1.5086	1.5003	1.5061
<i>CaduceusDC16</i>	1.4091	1.4082	1.4075	1.3995	1.3945	1.4088	1.4066	1.4049	1.3896	1.4015
<i>kawaii</i>	1.4302	1.4298	1.4289	1.4207	1.4162	1.4310	1.4284	1.4267	1.4091	1.4216
<i>ParsCat</i>	1.4778	1.4775	1.4772	1.4723	1.4699	1.4776	1.4761	1.4751	1.4675	1.4756
<i>Rubick</i>	1.3496	1.3492	1.3483	1.3423	1.3386	1.3490	1.3472	1.3454	1.3361	1.3492
<i>YXAgent</i>	1.3794	1.3789	1.3776	1.3637	1.3545	1.3793	1.3754	1.3723	1.3413	1.3581
Unspecified	1.4850	1.4874	1.4875	1.4412	1.4132	1.4725	1.4718	1.4632	1.3620	1.4416

Fig. 3 Ranking rates of the individual utility in each scenario

utility in each tournament. Our agent (*UCB(0.01)*) rarely rank as 5th or 6th because it can adopt the individual strategy by the meta strategy considering the negotiation situation. However, the first rank of our agent is small in each tournament despite that the averages of individual utility in all tournaments are high. This is because that our agent searches the effective negotiation strategy, although other agents not using the meta strategy can use the optimal strategy in all tournaments when their strategy is most effective in the tournament. In other words, one of the advantages of our agent is to avoid total defeats in tournaments.

Figure 4 shows that the anytime-curves for each kind of the meta-strategy, which the horizontal axis is the number of trials and the vertical axis is the average individual utilities of each trial in all tournaments. The proposed meta-strategies except for *UCB(0.5)*, *UCB(1)* and *EG(1)* can converge with the sufficient scores in about 7-10 trials. Therefore, the useful number of trials of the proposed approach

Fig. 4 The anytime-curves for each kind of the meta-strategy

is about 7-10. It is explainable results considering that the number of individual strategies in our meta-strategy is six. In other words, our proposed meta-strategy in the experimental settings can select an effective individual strategy in a sufficient number of trials. Comparing our meta-strategies with PureMAB, our meta-strategies can reach the convergence in the smaller trials. Especially, the meta-strategies with more exploitations can reach the convergence in the higher scores in the smaller trials.

Table 4 shows that the average coverage rates of the individual strategy in each meta-strategy. The average coverage rates are calculated as follows: First, the rank in each scenario based on the average of individual utilities in the six individual strategies is decided. Next, the average coverage rates of each rank in all profiles of the domains is calculated. *UCB(0.01)* can select an appropriate individual

Table 4 The average coverage rates of the individual strategy in each meta-strategy

%	<i>UCB(0.01)</i>	<i>UCB(0.05)</i>	<i>UCB(0.1)</i>	<i>UCB(0.5)</i>	<i>UCB(1)</i>	<i>EG(0)</i>	<i>EG(0.1)</i>	<i>EG(0.2)</i>	<i>EG(1)</i>	<i>PureMAB</i>
1st	65.369	63.258	60.229	48.449	44.021	65.073	62.703	59.523	39.578	23.397
2nd	12.069	12.756	13.468	17.356	17.869	11.426	12.888	14.520	17.706	18.855
3rd	12.847	11.775	12.140	10.919	11.664	12.784	11.977	10.972	14.085	16.001
4th	4.948	6.215	7.278	11.518	12.453	5.988	5.869	7.165	13.227	15.819
5th	2.699	3.688	4.050	6.994	8.471	2.529	3.922	4.538	9.249	14.950
6th	2.067	2.309	2.835	4.765	5.522	2.200	2.642	3.282	6.155	10.978

Table 5 The results of our agent and state-of-the-art agents

Agent	Individual utility	Social welfare
UCB(0.01)	0.7734	1.4575
<i>Agent33</i>	0.6901	1.4579
<i>AgentNP2018</i>	0.7082	1.4362
<i>Appaloosa</i>	0.7067	1.3706
<i>Ellen</i>	0.6083	1.2223
<i>TimeTraveler</i>	0.7142	1.4573

strategy from the statistic information of trials because the most appreciative individual strategy is about 65%. In addition, our proposed meta-strategy rarely selects the inappropriate individual strategies (5th and 6th), and selects the best individual strategy mostly for each opponent in a greedy manner. Our meta-strategy with an appropriate parameter such as *UCB(0.01)*, *UCB(0.05)*, *EG(0)*, and *EG(0.1)* can select the best individual strategies with the higher coverage rates. Therefore, the average coverage rates of our meta-strategy has the same tendencies with the results of the individual utilities.

Furthermore, if the negotiation scenario slightly changes, like the number of issues grows, the impatient discount factor or deadline becomes longer than 10 seconds, *UCB(0.01)* remains the best position. This is because that the characteristics of multi-armed bandit mainly depend on the arms which is the agent strategies in this case.

The tournament of our agent and state-of-the-art agents is conducted to compare our agent with state of the art agents. The tournament consisted of *UCB(0.01)*, *Agent33*, *AgentNP2018*, *Appaloosa*, *Ellen*, and *TimeTraveler*[†]. Table 5 shows the individual utility and the social welfare of each agent in the tournaments. The result of the tournament demonstrates that our agent outperforms state-of-the-art agents in individual utility. In addition, the score of our agent has significant difference to all other agents in Mann-Whitney U test ($p < .001$). Therefore, our meta-strategy is one of the effective strategies.

6. Related Work

Regarding bilateral closed negotiations, Chen et al. [21] proposed a negotiation approach called OMAC, which learns an opponent's strategy to predict future utilities of counter-offers by means of discrete wavelet decomposition and cubic smoothing splines. They also present a negotiation strategy called EMAR for this kind of environment that

relies on a combination of Empirical Mode Decomposition (EMD) and Autoregressive Moving Average (ARMA) [7]. EMAR enables a negotiating agent to acquire an opponent model and to use this model to adjust its target utility in real time on the basis of an adaptive concession-making mechanism. Hao et al. [22] proposed a negotiation strategy called ABiNeS, which was introduced for negotiations in complex environments. ABiNeS adjusts the time to stop exploiting the negotiating partner and also employs a reinforcement-learning approach to improve the acceptance probability of its proposals. Williams et al. [23] proposed a novel negotiating agent based on Gaussian processes in multi-issue automated negotiation against unknown opponents. Kawaguchi et al. [24] proposed a strategy for compromising the estimated maximum value based on estimated maximum utility. These papers have been important contributions for bilateral multi-issue closed negotiation; however, they do not deal with multi-time negotiation with learning and reusing the past negotiation sessions. After that, Fujita [25] proposed the compromising strategy with adjusting the speed of making agreements using the conflict mode, and focused on multi-time negotiation. Tunalı et al. [26] proposed the opponent model by introducing a novel frequency opponent modeling mechanism, which soothes some of the assumptions introduced by classical frequency approaches. Liu et al. [27] design a negotiating strategy searching the space of suitable bids that provide high utilities for both sides near the Nash Bargaining Solution (NBS) using a heuristic method. Ros et al. [28] proposed a meta-strategy that selects an action either a trade-off action or a concession during the negotiation, and Mansour et al. [29] extended the meta-strategy for multiple continuous issues and one-to-many negotiation. Their previous approaches focus on single-time negotiations, not consider the multi-time negotiation as this paper. Therefore, there are strong doubts that their previous approaches can work well in multi-time negotiations, not only single-time negotiations.

Recently, some studies have focused on divided parts of negotiating strategies in the alternating offering protocol: proposals, responses, and opponent modeling. Effective strategies can be achieved by combining the modules of top agents' strategies in the competitions depending on the opponent's strategies and negotiation environments. Many of the sophisticated agent strategies that currently exist are comprised of a fixed set of modules. Therefore, the studies for proposing the negotiation strategies focusing on the modules are important and influential. Baarslag et al. [6] focused on the acceptance dilemma: accepting the current

[†]These agents are the agents submitted to PRIANAC (one of the latest negotiation competition).

offer may be suboptimal, as better offers may still be presented. On the other hand, accepting too late may prevent an agreement from being reached at all, resulting in a break off with no gain for either party. This paper proposed new acceptance conditions and investigated correlations between the properties of the negotiation environment and the efficacy of acceptance conditions. Highly accurate, individual, and socially efficient opponent preference modeling in bilateral multi-issue negotiation has also been analyzed [30]. Their approaches are similar to our proposed approach from the point of view of combining the modules of top agents' strategies in the competitions depending on the opponent's strategies and negotiation environments. However, their previous approaches don't consider the meta-strategy of switching the individual strategies in multi-time negotiations like this paper.

On the whole, in previous studies related to automated negotiations, most of the agents negotiate based on only one policy: one bidding strategy, one concession strategy, and one opponent modeling method. There are few studies about meta-strategy despite the fact that it plays an important role in adjusting agents to various situations.

Ilany and Gal proposed the strategy selection method based on supervised learning using structural features of the negotiation domain [15]. Further, they presented two selection methods: the selection method based on a bandit approach (pure-MAB) and the method that combines pure-MAB and the supervised learning approach (prior-MAB). Although prior-MAB outperforms other single negotiation strategy agents, it needs training data for each class of negotiating problems. In addition, pure-MAB and prior-MAB calculate the UCB score for each its own profile. In other words, they do not consider the opponent's strategy and profile in the strategy selection. Our approach calculates the UCB score for each combination of its own profile and the opponent's strategy and profile.

7. Conclusion and Future Work

In this paper, we proposed a meta-strategy, which provides a suitable negotiation strategy for the situation to the agent, for multi-time negotiation. The strategy selection of our meta-strategy was modelled as a multi-armed bandit problem which regards negotiation strategies as slot machines and individual utilities as rewards. We implemented agents with our meta-strategy, and conducted the experiments to evaluate the performance of our agents. The experimental results demonstrated that our agents outperformed other existing agents in various situations, and it demonstrated the effectiveness of our meta-strategy. The results also revealed that the exploration of multi-armed bandit problem is not important in the conditions of this paper.

One of the possible future work is the improvements of individual negotiation strategies included in our meta-strategy. We used the existing negotiation strategies as the individual strategies in this paper. To improve the performance of our agent, the optimized strategies for certain

situations should be included. Other possible future research is to consider dynamic conditions like changing the preferences through repetitions, or if the opponent adopts a meta-strategy.

References

- [1] R. Kawata and K. Fujita, "Metax-strategy for multi-time negotiation: A multi-armed bandit approach," *Proc. 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS2019)*, pp.2048–2050, 2019.
- [2] N.R. Jennings, P. Faratin, A.R. Lomuscio, S. Parsons, M.J. Wooldridge, and C. Sierra, "Automated negotiation: prospects, methods and challenges," *Group Decision and Negotiation*, vol.10, no.2, pp.199–215, 2001.
- [3] S. Fatima, S. Kraus, and M. Wooldridge, *Principles of Automated Negotiation*, Cambridge University Press, 2014.
- [4] R. Aydogan, T. Baarslag, K. Fujita, T. Ito, C. Jonker, D. de Jonge, and J. Mell, "The Ninth International Automated Negotiating Agents Competition (ANAC2018)." <http://web.tuat.ac.jp/katfuj/ANAC2018/>, 2018.
- [5] T. Baarslag, K. Fujita, E. Gerding, K. Hindriks, T. Ito, N.R. Jennings, C. Jonker, S. Kraus, R. Lin, V. Robu, and C. Williams, "Evaluating practical negotiating agents: Results and analysis of the 2011 international competition," *Artificial Intelligence Journal (AIJ)*, vol.198, pp.73–103, 2013.
- [6] T. Baarslag and K.V. Hindriks, "Accepting optimally in automated negotiation with incomplete information," *Proc. 12th International conference on Autonomous agents and multi-agent systems (AAMAS 2013)*, pp.715–722, 2013.
- [7] S. Chen, H.B. Ammar, K. Tuyls, and G. Weiss, "Conditional restricted boltzmann machines for negotiations in highly competitive and complex domains," *Proc. 23th International Joint Conference on Artificial Intelligence (IJCAI 2013)*, pp.69–75, 2013.
- [8] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol.58, no.5, pp.527–535, 1952.
- [9] S.S. Fatima, M. Wooldridge, and N.R. Jennings, "Multi-issue negotiation with deadlines," *Journal Artificial Intelligence Research*, vol.27, no.1, pp.381–417, 2006.
- [10] A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica: Journal of the Econometric Society*, vol.50, no.1, pp.97–109, 1982.
- [11] T. Baarslag, K. Hindriks, and C. Jonker, "Towards a quantitative concession-based classification method of negotiation strategies," *Agents in Principle, Agents in Practice*, ed. D. Kinny, J.Y.j. Hsu, G. Governatori, and A.K. Ghose, Berlin, Heidelberg, pp.143–158, Springer Berlin Heidelberg, 2011.
- [12] J. Zhu, Y. Song, D. Jiang, and H. Song, "Multi-armed bandit channel access scheme with cognitive radio technology in wireless sensor networks for the internet of things," *IEEE Access*, vol.4, pp.4609–4617, 2016.
- [13] S.S. Villar, J. Bowden, and J. Wason, "Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges," *Statistical science: a review journal of the Institute of Mathematical Statistics*, vol.30, no.2, pp.199–215, 2015.
- [14] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol.47, no.2-3, pp.235–256, 2002.
- [15] L. Ilany and Y. Gal, "Algorithm selection in bilateral negotiation," *Autonomous Agents and Multi-Agent Systems*, vol.30, no.4, pp.697–723, July 2016.
- [16] T. Baarslag, R. Aydogan, K.V. Hindriks, K. Fujita, T. Ito, and C.M. Jonker, "The automated negotiating agents competition, 2010-2015," *AI Magazine*, vol.36, no.4, pp.115–118, 2015.
- [17] R. Aydogan, T. Baarslag, K. Fujita, K. Hindriks, T. Ito, and C.

- Jonker, "The Seventh International Automated Negotiating Agents Competition (ANAC2016)." <http://web.tuat.ac.jp/katfujii/ANAC2016/>, 2016.
- [18] R. Aydoğan, T. Baarslag, K. Fujita, T. Ito, D. de Jonge, C. Jonker, and J. Mell, "The Eighth International Automated Negotiating Agents Competition (ANAC2017)." <http://web.tuat.ac.jp/katfujii/ANAC2017/>, 2017.
- [19] A. Mori and T. Ito, "Atlas3: A negotiating agent based on expecting lower limit of concession function," in *Modern Approaches to Agent-based Complex Automated Negotiation*, pp.169–173, Springer, 2017.
- [20] R. Lin, S. Kraus, T. Baarslag, D. Tykhonov, K. Hindriks, and C.M. Jonker, "Genius: An integrated environment for supporting the design of generic automated negotiators," *Computational Intelligence*, vol.30, no.1, pp.48–70, 2014.
- [21] S. Chen and G. Weiss, "An efficient and adaptive approach to negotiation in complex environments," *Proc. 19th European Conference on Artificial Intelligence (ECAI 2012)*, pp.228–233, 2012.
- [22] J. Hao and H.-F. Leung, "Abines: An adaptive bilateral negotiating strategy over multiple items," *2012 IEEE/WIC/ACM International Conferences on Intelligent Agent Technology (IAT 2012)*, pp.95–102, 2012.
- [23] C.R. Williams, V. Robu, E.H. Gerding, and N.R. Jennings, "Using gaussian processes to optimise concession in complex negotiations against unknown opponents," *Proc. 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pp.432–438, 2011.
- [24] S. Kawaguchi, K. Fujita, and T. Ito, "Compromising strategy based on estimated maximum utility for automated negotiation agents competition (anac-10)," *24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2011)*, pp.501–510, 2011.
- [25] K. Fujita, "Compromising adjustment strategy based on TKI conflict mode for multi-times bilateral closed negotiations," *Computational Intelligence*, vol.34, no.1, pp.85–103, 2018.
- [26] O. Tunalı, R. Aydoğan, and V. Sanchez-Anguix, "Rethinking frequency opponent modeling in automated negotiation," *Proc. 20th Principles and Practice of Multi-Agent Systems (PRIMA 2017)*, pp.263–279, Springer International Publishing, 2017.
- [27] S. Liu, A. Moustafa, and T. Ito, "Agent33: An automated negotiator with heuristic method for searching bids around nash bargaining solution," *Proc. 21st Principles and Practice of Multi-Agent Systems (PRIMA 2018)*, pp.519–526, 2018.
- [28] R. Ros and C. Sierra, "A negotiation meta strategy combining trade-off and concession moves," *Autonomous Agents and Multi-Agent Systems*, vol.12, no.2, pp.163–181, 2006.
- [29] K. Mansour and R. Kowalczyk, "A meta-strategy for coordinating of one-to-many negotiation over multiple issues," *Foundations of Intelligent Systems*, ed. Y. Wang and T. Li, Berlin, Heidelberg, pp.343–353, Springer Berlin Heidelberg, 2012.
- [30] F. Zafari and F.N. Mofakham, "POPPONENT: highly accurate, individually and socially efficient opponent preference model in bilateral multi issue negotiations," *Proc. Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI 2017)*, pp.5100–5104, 2017.

Ryohei Kawata is a master student of the faculty of Engineering, Tokyo University of Agriculture and Technology. He received the B.S. degree from the Departments of Computer and Information Sciences, the Faculty of Engineering, Tokyo University of Agriculture and Technology, Japan, in 2018. His research interests are in automated negotiation and multi-agent systems.

Katsuhide Fujita is an Associate Professor of the Institute of Engineering, Tokyo University of Agriculture and Technology. He received the B.E., M.E. and Doctorate of Engineering from Nagoya Institute of Technology in 2008, 2010, and 2011, respectively. From 2010 to 2011, he was a research fellow of the Japan Society for the Promotion of Science (JSPS). During 2010 and 2011, he was a visiting researcher at MIT Sloan School of Management. From 2011 to 2012, he was a Project Researcher of the School of Engineering, the University of Tokyo. He is an Associate Professor of the Institute of Engineering, Tokyo University of Agriculture and Technology since 2012. His main research interests include multi-agent systems and decision support systems.