2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology

ABiNeS: An Adaptive Bilateral Negotiating Strategy over Multiple Items

Jianye Hao
Department of Computer Science and Engineering
The Chinese University of Hong Kong
jyhao@cse.cuhk.edu.hk

Ho-fung Leung
Department of Computer Science and Engineering
The Chinese University of Hong Kong
lhf@cse.cuhk.edu.hk

***Abstract*—Multi-item negotiations surround our daily life and usually involve two parties that share common or conflicting interests. Effective automated negotiation techniques should enable the agents to adaptively adjust their behaviors depending on the characteristics of their negotiating partners and negotiation scenarios. This is complicated by the fact that the negotiation agents are usually unwilling to reveal their information (strategies and preferences) to avoid being exploited during negotiation. In this paper, we propose an adaptive negotiation strategy, called *ABiNeS*, which can make effective negotiations against different types of negotiating partners. The *ABiNeS* agent employs the non-exploitation point to adaptively adjust the appropriate time to stop exploiting the negotiating partner and also predicts the optimal offer for the negotiating partner based on reinforcement-learning based approach. Simulation results show that the *ABiNeS* agent can perform more efficient exploitations against different negotiating partners, and thus achieve higher overall utilities compared with the state-of-the-art negotiation strategies in different negotiation scenarios.**

I. Introduction

Negotiation is a common and important approach to resolve conflicts and reach agreements between different parties in our daily life. The topic of negotiation has been widely studied in various areas, such as decision and game theory, economics, and social science [1]. Automated negotiation techniques can, to a large extent, alleviate the efforts of human, and also facilitate human in reaching better negotiation outcomes by compensating for the limited computational abilities of humans when they are faced with complex negotiations. Until now, a lot of automated negotiation strategies and mechanisms have been proposed in different negotiation scenarios [2]–[6].

The major difficulty in designing automated negotiation agent is how to achieve optimal negotiation results given incomplete information on the negotiating partner. The negotiation partner usually keeps its negotiation strategy and its preference as its private information to avoid exploitations. A lot of research efforts have been devoted to better understand the negotiation partner by either estimating the negotiation partner's preference profile [4], [7], [8] or predicting its decision function [5], [9]. On one hand, with the aid of different preference profile modeling techniques, the negotiating agents can get a better understanding of their negotiating partners and thus increase their chances of reaching mutually beneficial negotiation outcomes. On the other hand, effective strategy prediction techniques enable the negotiating agents to maximally exploit their negotiating partners and thus receive as much benefit as possible from negotiation. However, in most of previous work, the above two aspects are often investigated separately and little efforts have been devoted to combine them together and evaluate the negotiation performance of various combinations of different techniques. To this end, in recent years a number of negotiation strategies, which take advantage of existing techniques from both aspects as previously mentioned, have been proposed and agents employing these strategies have participated in *automated negotiating agents competition (ANAC)* [10], [11]. During the competitions, their performance has been extensively evaluated in a variety of multi-issue negotiation scenarios and valuable insights have been obtained in terms of the advantages and disadvantages of different techniques, e.g., the efficacy of different acceptance conditions [12]. It is still an open and interesting problem to design more efficient automated negotiation strategies against a variety of negotiating opponents in different negotiation domains.

In this paper, we propose an adaptive negotiation strategy *ABiNeS* for automated agents to negotiate in bilateral multi-issue negotiation environments following the settings adopted in ANAC'12 [13]. Bilateral multi-issue negotiations surround people's daily life and have received lots of attention in the negotiation literature. During negotiation, both the agents' negotiation strategies and preference profiles are their private information, and for each agent the only available information about the negotiating partner is its past negotiation moves. Considering the diversity of the available negotiation strategies that the negotiating agents can adopt, it is usually very difficult (or impossible) to predict which specific strategy the negotiating partner is using based on this limited information. To effectively cope with different types of opponents, we introduce the concept of non-exploitation point λ to adaptively adjust the degree that an *ABiNeS* agent exploits its negotiating opponent. The value of λ is determined by the characteristics of the negotiation scenario and the concessive degree of the negotiating partner, which is estimated based on the negotiation history. Besides, to maximize the possibility that the offer the *ABiNeS* agent

978-0-7695-4880-7/12 $26.00 © 2012 IEEE
DOI 10.1109/WI-IAT.2012.72

proposes will be accepted by its negotiating partner, it can be useful to make predictions on the preference profile of the negotiating partner. Instead of explicitly modeling the negotiation partner's preference profile, we propose a reinforcement-learning based approach to determine the optimal proposal for the negotiating partner based on the current negotiation history. We compare the performance of the *ABiNeS* agent with that of the state-of-the-art negotiation agents and simulation results show that the *ABiNes* agent can make more effective exploitations against a variety of negotiating partners and thus obtain higher average utilities during negotiation.

The remainder of the paper is organized as follows. In section II, we give a description of negotiation model we consider in this paper. In section III, the negotiation strategy *ABiNes* we propose is introduced. In section IV, we present our simulation results to compare the negotiation performance of *ABiNes* with the state-of-the-art agents in different domains. An overview of related work on automated negotiation strategies is given in section V. Lastly conclusion and future work are given in section VI.

II. Negotiation Model

In this section, we describe the negotiation model we consider in this work, which follows the settings adopted in ANAC'12 [13]. In this work, we focus on bilateral negotiations, i.e., negotiations between two agents. Specifically, the alternating-offers protocol is adopted to regulate the interactions between the negotiating agents, in which the agents take turns to exchange proposals. For each negotiation scenario, both agents can negotiate over multiple issues (items), and each item can have a number of different values. Let us denote the set of items as $\mathcal{M}$, and the set of values for each item $m_i \in \mathcal{M}$ as $\mathcal{V}_i$.[1] We define a negotiation outcome ω as a mapping from every item $m_i \in \mathcal{M}$ to a value $v \in \mathcal{V}_i$, and the negotiation domain is defined as the set Ω of all possible negotiation outcomes. For each negotiation outcome ω, we use $\omega(m_i)$ to denote the corresponding value of the item m_i in the negotiation outcome ω. We assume that the knowledge of the negotiation domain is known to both agents beforehand, and is not changed during the whole negotiation session.

For each negotiation outcome ω, different agents may have different preferences. Here we assume that each agent i's preference can be modeled by a utility function u_i such that $\forall \omega \in \Omega$, it is mapped into a real-valued number in the range of [0,1], i.e., $u_i(\omega) \in [0, 1]$. In practical negotiation environments, there usually associates a certain cost with each negotiation. To take this factor into consideration, a real-time deadline is imposed on the negotiation process and each agent's actual utilities over the negotiation outcomes are decreased by a discounting factor δ over time. Following the setting adopted in ANAC'12, each negotiation session is allocated 3 minutes, which is normalized into the range of [0,1], i.e., $0 \leq t \leq 1$. Formally, if an agreement is reached at time t before the deadline, each agent i's actual utility function $U_i^t(\omega)$ over this mutually agreed negotiation outcome ω is defined as follows,

$$U_i^t(\omega) = u_i(\omega)\delta^t \tag{1}$$

If no agreement is reached by the deadline, each agent i will obtain a utility of $ru_i^0\delta$, where ru_i^0 is agent i's private reservation value in the negotiation scenario. The agents will also obtain their corresponding reservation values if the negotiation is terminated before the deadline. Note that the agents' actual utilities over their reservation values are also discounted by the discounting factor δ over time t. We assume that the agents' preference information and their reservation values are private and can not be accessed by their negotiating partners.

The interaction between the negotiation agents is regulated by the alternating-offers protocol, in which the agents are allowed to take turns to exchange proposals. During each encounter, if it is agent i's turn to make a proposal, it is allowed to make a choice from the following three options:

Accept the offer from its negotiating partner
: In this case, the negotiation ends and an agreement is reached. Both agents will obtain the corresponding utilities according to Equation 1, where ω is the negotiation outcome that they mutually agree with.

Reject & propose a counter-offer to its negotiating partner
: In this case, the negotiation process continues and it is its negotiating partner's turn to make a counter-proposal next time provided that the deadline is not reached yet.

Terminate the negotiation
: In this case, the negotiation terminates and each agent i gets its corresponding utility based on its private reservation value with the initial value of ru_i^0. Note that their actual utilities in this case are also decreased over time by the same discounting factor δ, i.e., $U_i^t = ru_i^0 * \delta^t$.

Overall, the negotiation process terminates when either of the following conditions is satisfied: 1) the deadline is reached (*End*); 2) an agent chooses to terminate the negotiation before reaching deadline (*Terminate*); 3) an agent chooses to accept the negotiation outcome proposed by its negotiating partner (*Accept*).

For each negotiation session between two agents A and B, let $x^t_{A\rightarrow B}$ denote the negotiation outcome proposed by agent A to agent B at time t. Naturally a negotiation history $H^t_{A\leftrightarrow B}$ between agent A and B until time t can be represented as follows,

$$H^t_{A\leftrightarrow B}: = (x^{t_1}_{p_1\rightarrow p_2}, x^{t_2}_{p_3\rightarrow p_4}, \ldots, x^{t_n}_{p_n\rightarrow p_{n+1}}) \tag{2}$$

where

[1]Here $\mathcal{V}_i$ can be either discrete values or continuous real values.

- $t_k \leq t_l$ for $k \leq l$, i.e., the negotiation outcomes are ordered over time, and also $t_n \leq t$.
- $p_k = p_{k+2} \in \{A, B\}$, i.e., the negotiation process strictly follows the alternating-offers protocol.

Similarly, we denote the negotiation history during a certain period of time (between time t_1 and t_2) as $H_{A\leftrightarrow B}^{t_1\rightarrow t_2}$. As previously mentioned, we know that a negotiation session between two agents A and B will terminate either when one agent chooses an action from the set $\{Accept, Terminate\}$, or the deadline is reached. Thus the last element of a complete negotiation history can be either one of the elements from the set $S = \{Accept, Terminate, End\}$. A negotiation history by time t is active if its last element is not equal to any element in S.

III. *ABiNeS*: An Adaptive Bilateral Negotiating Strategy

In this section, we describe the adaptive negotiating strategy *ABiNeS* in details. For the ease of description, we refer to the *ABiNeS* agent as agent A and its negotiating partner as agent B in the following descriptions. The *ABiNeS* strategy mainly consists of four basic decision components. The first component is *Acceptance-Threshold (AT)* component and it is responsible for determining the *ABiNeS* agent's minimum acceptance threshold l_A^t at time t. The second component is *Next-Bid (NB)* component whose function is to determine the negotiation outcome $x_{A\rightarrow B}^t$ that the *ABiNeS* agent proposes at time t. The third component, *Acceptance-Condition (AC)* component, is used for determining whether to accept the current proposal from agent B or not. Given a negotiation history $H_{A\leftrightarrow B}^t$, its acceptance threshold l_A^t, and its negotiation outcome $x_{A\rightarrow B}^t$ to propose at time t, the AC component returns a boolean value indicating whether to accept the offer or not, which is denoted as $AC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t)$. The last component is *Termination-Condition (TC)* component. It is responsible for determining whether to terminate the current negotiation or not. Similar to the AC component, given a negotiation history $H_{A\leftrightarrow B}^t$, its acceptance threshold l_A^t, its negotiation outcome $x_{A\rightarrow B}^t$ to propose at time t, and its reservation utility ru_A^0, the TC component returns a boolean value indicating whether to terminate the negotiation or not, which is denoted as $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$.

The overall description of the negotiation strategy *ABiNeS* based on the above basic elements is presented in Algorithm 1. At time t the acceptance threshold and the next-round negotiation outcome are calculated first based on the AT and NB components respectively. If the *ABiNeS* agent is the first one to make a proposal, then it is faced with two options: either proposing a negotiation outcome to agent B or terminating the negotiation (Lines 2 - 8). Otherwise, the *ABiNeS* agent makes a choice among three options: proposing a negotiation outcome to agent B, choosing to terminate the negotiation and accepting the offer from agent B (Lines 10 - 22). The decisions on whether to accept the offer or terminate the negotiation are determined by the AC and TC components respectively. We introduce each decision component in details in the following sections.

Algorithm 1 Overall Flow of the *ABiNeS* Strategy

1: **for** each negotiation history $H_{A\leftrightarrow B}^t := (x_{p_1\rightarrow p_2}^{t_1}, x_{p_3\rightarrow p_4}^{t_2}, \ldots, x_{p_n\rightarrow p_{n+1}}^{t_m})$ at current time t **do**
2:   Determine the acceptance threshold l^t and the negotiation outcome $x_{A\rightarrow B}^t$ to be proposed to the negotiating partner B, and $ru_A^t = ru_i^0 * \delta^t$.
3:   **if** $H_{A\leftrightarrow B}^t$ is empty **then**
4:     **if** $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$ is false **then**
5:       Propose the negotiation outcome $x_{A\rightarrow B}^t$ to the negotiating partner.
6:     **else**
7:       Terminate the negotiation.
8:     **end if**
9:   **else**
10:     **if** $AC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t)$ is true and $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$ is false **then**
11:       Accept the offer.
12:     **else if** $AC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t)$ is false and $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$ is true **then**
13:       Terminate the negotiation.
14:     **else if** $AC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t)$ is true and $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$ is true **then**
15:       **if** $U_A^t(x_{p_n\rightarrow p_{n+1}}^{t_m}) > ru_A^t$ **then**
16:         Accept the offer.
17:       **else**
18:         Terminate the negotiation
19:       **end if**
20:     **else**
21:       Propose the negotiation outcome $x_{A\rightarrow B}^t$ to the negotiating partner.
22:     **end if**
23:   **end if**
24: **end for**

A. Acceptance-Threshold (AT) Component

The AT component is responsible for determining the acceptance threshold l of the *ABiNeS* strategy during negotiation. The value of the acceptance threshold reflects the agent's current concession degree and should be adaptively adjusted based on the past experience and the characteristic of the negotiation environment. Besides, it also has explicit influence on the decision-making process of the other three components, which will be introduced later.

We assume that the negotiating partner is self-interested, and it will accept any proposal when the deadline is approaching ($t \sim 1$). Therefore the acceptance threshold of an *ABiNeS* agent should be always higher than the highest utility it can obtain when $t = 1$. Specifically, at any time t, the acceptance threshold l_t of the *ABiNeS* agent should not be lower than $u^{max}\delta^{1-t}$, where u^{max} is its maximum utility over the negotiation domain without discounting. Since the negotiating goal is to reach an agreement which maximize the agent's own utility as much as possible, its negotiating partner should be exploited as much as possible by setting its acceptance threshold as high as possible. One the other hand, due to the discounting effect, the actual utility the agent receives can become extremely low though its original utility over the mutually-agreed negotiation outcome is high, if it takes too long for the agents to reach the agreement. In the worst case the negotiation may end up with a break-off and each agent obtains zero utility. Thus we also need to make certain compromises to the negotiating partner, i.e., lower the acceptance threshold, depending on the type of the partner we are negotiating with. Therefore, the key problem is how to balance the trade-off between exploiting and making compromise to the negotiating partner. Towards this end, we introduce the adaptive non-exploitation point λ, which represents the specific time when we should stop exploitations on the negotiating partner. This value is adaptively adjusted based on the behavior of the negotiating partner. Specifically we propose that for any time $t < \lambda$, the *ABiNeS* agent always exploits its negotiating partner (agent B) by setting its acceptance threshold to a value higher than $u^{max}\delta^{1-\lambda}$ and approaching this value until time λ according to certain pattern of behavior. After time λ, its acceptance threshold is set to be equal to $u^{max}\delta^{1-t}$ forever, and any proposal over which its utility is higher than this value will be accepted. Formally, the acceptance threshold l_A^t of an *ABiNeS* agent at time t is determined as follows,

$$l_A^t = \begin{cases} u^{max} - (u^{max} - u^{max}\delta^{1-\lambda})(\frac{t}{\lambda})^{\alpha} & \text{if } t < \lambda \\ u^{max}\delta^{1-t} & \text{otherwise} \end{cases} \tag{3}$$

where the variable α controls the way the acceptance threshold approaches $u^{max}\delta^{1-t}$ (boulware ($\alpha > 1$), conceder ($\alpha < 1$) or linear ($\alpha = 1$)). One example showing the dynamics of the acceptance threshold with time t with different value of λ is given in Figure 1.

The remaining question is how to calculate the value of non-exploitation point λ. The value of λ is determined by the characteristics of the negotiation scenario (i.e., discounting factor δ) and the concession degree of the negotiating partner. The smaller the discounting factor δ is, the less actual utility we will receive as time goes by, which means more risk we are facing when we continue exploiting the negotiating partner. Therefore the value of λ should be decreased with the decreasing of the discounting factor δ. The concession degree of the negotiating partner is estimated based on its past behaviors. Intuitively, the more number of new negotiation outcomes that the negotiating parter has recently proposed, the more it is willing to make concession to end the negotiation. Specifically, the negotiation partner's concessive degree σ^t is defined as the ratio of new negotiation outcomes it proposed within the most recent finite negotiation history $H_{A\leftrightarrow B}^{t'\to t}$. If we predict that the negotiating partner is becoming more concessive, we can take advantage of this prediction by postponing the time we stop exploitations, i.e., increasing the value of λ.

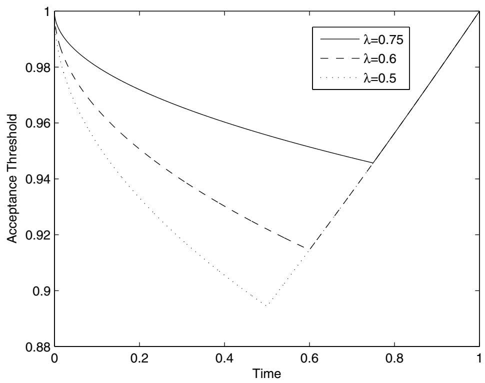

Figure 1: The dynamics of the acceptance threshold ($u^{max} = 1$, $\alpha = 0.5$ and $\delta = 0.8$).

Initial values
- λ_0 - the minimum value of λ,
- β - the controlling variable determining the way the value of λ changes with respect to the discounting factor δ, i.e., boulware ($\beta < 1$), conceder ($\beta > 1$) or linear ($\beta = 1$),
- σ^t - the estimation of the negotiating partner's concessive degree at time t,
- γ - the controlling variable determining the way the value of λ changes with respect to σ^t, i.e., boulware ($\gamma < 1$), conceder ($\gamma > 1$) or linear ($\gamma = 1$),
- w - the weighting factor adjusting the relative effect of σ^t on the non-exploitation point λ.

if $t = 0$ **then**
  $\lambda = \lambda_0 + (1 - \lambda_0)\delta^{\beta}$
end if
if $0 < t \leq 1$ **then**
  $\lambda = \lambda + w(1 - \lambda){\sigma^t}^{\gamma}$
end if

Figure 2: Adjustment rule of λ at time t

Initially the value of λ is determined by the discounting factor δ only since we do not have any information on the negotiating partner yet. After that, it is adaptively adjusted based on the estimation of the concession degree of the negotiating partner. The overall adjustment rule of λ during negotiation is shown in Figure 2.

B. Next-Bid (NB) Component

The next-bid component is responsible for determining the negotiation outcome to be proposed to the negotiating partner. Given the current acceptance threshold l_A^t at time t, any negotiation outcome over which the *ABiNeS* agent's utility is higher than l_A^t can be a reasonable outcome to propose. To maximize the likelihood that the offer will be accepted by agent B, we need to predict the negotiation outcome ω_{max} which can maximize its utility among the set $\mathcal{C}$ of candidate negotiation outcomes (i.e., $\mathcal{C} = \{\omega \mid u_A(\omega) \geq l_A^t\}$). The negotiation outcome ω_{max} will be returned by the NB component as the offer to be proposed to agent B.

To obtain ω_{max}, we need to estimate agent B's private preference based on its past negotiation moves. Different approaches [2], [4], [7], [14] have been proposed to explicitly estimate the the negotiating partner's utility function in bilateral negotiation scenarios. To make the estimation feasible with the limited information available, we usually need to put some restrictions on the possible structures that the negotiation partner's utility function can have [4] or assume that the preference profile of the negotiation partner is chosen from a fixed set of profiles [7]. Instead of estimating agent B's utility function directly, here we adopt a model-free reinforcement learning based approach to predict the current best negotiation outcome for agent B. The only assumption we need here is that the negotiating partner (agent B) is individually rational and follows some kind of concession-based strategy when proposing bids, which is the most commonly used assumption in both game-theoretic approaches and negotiations [4], [15].

Based on the above assumption, it is natural to assume that the sequence of past negotiation outcomes proposed by agent B should be in accordance with the decreasing order of its preference over those outcomes. Intuitively, for a value v_i of an item m_i, the earlier and the more frequent it appears in the negotiation outcomes of the past history, the more likely that it weights more in contributing to the negotiation partner's overall utility. Therefore, for each value of each item m_i in the negotiation domain, we keep record of the number of times that it appears in the negotiating partner's past negotiation outcomes and update its value each time a new negotiation outcome ω' is proposed by agent B as follows,

$$n(\omega'(m_i)) = n(\omega'(m_i)) + \eta^k \ \forall m_i \in \mathcal{M} \qquad (4)$$

where ω' is the most recent negotiation outcome proposed by agent B, η is the discounting factor reflecting the decreasing speed of the relative importance of the negotiation outcomes as time increases, and k is the number of times that the value $\omega'(m_i)$ of item m_i has appeared in the history.

For each negotiation outcome ω, we define its accumulated frequency $f(\omega)$ as the criterion for evaluating the relative preference of agent B over it. The value of $f(\omega)$ is determined by the value of $n(\omega(m_i))$ for each item $m_i \in \mathcal{M}$ based on the current negotiation history. Formally, for any negotiation outcome ω, its accumulated frequency $f(\omega)$ is calculated as follows,

$$f(\omega) = \sum_{m_i} n(\omega(m_i)) \ \forall m_i \in \mathcal{M} \qquad (5)$$

The negotiation outcome ω_{max} is selected based on the ϵ-greedy exploration mechanism. With probability $1 - \epsilon$, it chooses the negotiation outcome with the highest f-value from the set $\mathcal{C}$ of candidate negotiation outcomes, and chooses one negotiation outcome randomly from $\mathcal{C}$ with probability ϵ. The value of ϵ controls the exploration degree during prediction. One exception is that the negotiation outcome ω_{max} will be selected as the best negotiation outcome proposed by the negotiating agent B in the history if the NB component has received the corresponding signal from the AC component, and this will be explained in details in next section.

C. Acceptance-Condition (AC) Component

Given the current negotiation history $H^t_{A\leftrightarrow B}$, agent A's acceptance threshold l_A^t, and its negotiation outcome $x^t_{A\rightarrow B}$ to propose at time t, the AC component determines whether to accept the current proposal of agent B or not. The overall acceptance conditions are described in Algorithm 2. The *ABiNeS* agent accepts the proposal ω_1 from its negotiating agent B if its utility over ω_1 is either higher than its current acceptance threshold l_A^t or its utility over its next-to-bid negotiation outcome $u_A(x^t_{A\rightarrow B})$ (Lines 2 - 3). Otherwise, it checks whether there exists some negotiation outcome ω_{best} previously proposed by its negotiating agent B satisfying the above condition. If the answer is yes, then it will notify the NB component to propose ω_{best} next time (Lines 4 - 5).

Algorithm 2 Acceptance Conditions $AC(H^t_{A\leftrightarrow B}, l_A^t, x^t_{A\rightarrow B})$

1: Initialization: ω_{best} = BestNegotiationOutcome($H^t_{A\leftrightarrow B}$), and let ω_1 be the negotiation outcome proposed by agent B obtained from $H^t_{A\leftrightarrow B}$.
2: **if** $u_A(\omega_1) > l_A^t$ or $u_A(\omega_1) > u_A(x^t_{A\rightarrow B})$ **then**
3:     accept the offer (return true).
4: **else if** $u_A(\omega_{best}) > l_A^t$ or $u_A(\omega_{best}) > u_A(x^t_{A\rightarrow B})$ **then**
5:     not accept (return false) and notify the NB component to propose the negotiation outcome ω_{best} next time.
6: **else**
7:     not accept (return false).
8: **end if**

D. Termination-Condition (TC) Component

This component is responsible for deciding whether to terminate the negotiation and receive the corresponding reservation value or not. Here we treat the reservation value as an alternative offer from the negotiating agent B with a constant utility ru_A^0. Thus the termination conditions of TC component are similar to the acceptance conditions except that $u_A(\omega_1)$ is replaced with the reservation value ru_A^0. The only difference is that we do not need to check the best negotiation outcome proposed by agent B in the history since the reservation value is unchanged throughout the negotiation session. The detailed mechanism for TC component is shown in Algorithm 3.

Algorithm 3 Termination Conditions $TC(H_{A\leftrightarrow B}^t, l_A^t, x_{A\rightarrow B}^t, ru_A^0)$

1: **if** $ru_A^0 > l_A^t$ or $ru_A^0 > u_A(x_{A\rightarrow B}^t)$ **then**
2: accept the offer (return true).
3: **else**
4: not accept (return false).
5: **end if**

IV. Experiments and Evaluations

A. Experimental Settings

We evaluate the negotiation performance of the *ABiNeS* agent in different multi-issue negotiation scenarios. We adopt the GENIUS (General Environment for Negotiation with Intelligent multi-purpose Usage Simulation) [16] platform to evaluate the negotiation efficiency of the *ABiNeS* agent. GENIUS is a negotiation platform developed for facilitating research on bilateral multi-issue negotiations, and has been employed in ANAC competitions in previous years [10], [11]. All the requirements described in the negotiation model in Section II are supported in GENIUS. Following the default setting of GENIUS, the maximum negotiation time of each negotiation session is set to 3 minutes and normalized into the range of [0,1].

We compare the negotiation efficiency of the *ABiNeS* agent with eight state-of-the-art negotiation strategies which enter the final round of ANAC'11: HardHeaded, Gahboninho, IAMHaggler'11, BRAM Agent, AgentK2, The Negotiator, Nice Tit for Tat agent and Value Model Agent. which are implemented by negotiation experts from different research groups.[2] Following the competition rules of ANAC'11, for each negotiation scenario, each agent matches with all the other eight agents except itself. Besides, for each negotiation domain, the preference profile on both sides are usually different, thus each pair of negotiating agents negotiates with each other under both profiles. Finally, the result of each negotiation session between any pair of agents is averaged over three independent runs to eliminate the effects of randomness in their strategies. We adopt each agent's average utility obtained against all opponents over both preference profiles as the metric for evaluating the negotiation performance of the agents. Next we will describe the negotiation domains we consider here in Section IV-B, and then present and analyse the negotiation results in Section IV-C.

B. Negotiation Domains

For each negotiation domain, the agents negotiate over a large number of different negotiation outcomes in terms of which value to choose for each issue involved in this domain. The negotiation domains can be classified based on the notion of weak and strong opposition [12]. Domains with strong opposition mean that the agents have strongly opposite interests over the negotiation outcomes and the gain of one agent must come at the loss of the other agent. Domains with weak opposition refer to those domains in which it is possible for the agents to reach a win-win agreement.

In this work, we consider two different types of negotiation domains: Itex vs Cypress domain with strong opposition and Grocery domain with weak opposition. The domain of Itex vs Cypress is adopted in [10], which models the negotiation process between a buyer and a seller. In this domain, it involves two parties: Itex Manufacturing (a bicycle components producer) and Cypress Cycles (a bicycle builder). They negotiate over four issues here: the prices of the components, the delivery time, payment arrangement and terms for return of defective components. Both parties have strong opposition in this domain since the bicycle producers and consumers usually have opposite requirements. The size of the domain is 180, and the discounting factor and the reservation value are set to 0.5 and 0.1 respectively.

Another domain, the Grocery domain [11], is much less competitive compared with Itex vs Cypress domain. In this domain, two parties negotiate over which types of food to store in the grocery. In particular five different types of food are considered: bread type, fruit, snacks, spreads, and vegetables, with a number of choices available for each issue. The total number of possible agreements in this domain is 1600, and the discounting factor and the reservation value are set to 0.81 and 0.1 respectively.

C. Experimental Results and Analysis

Table I shows the expected overall utility of each agent averaged over all negotiation sessions in the aforementioned two negotiation scenarios. The *ABiNeS* agent receives the highest average utility among all agents in both scenarios. Besides, it is worth noticing that the utility that each agent achieves in the domain of Itex vs Cypress is much lower than that in the domain of grocery, which is in accordance with

[2] All the eight agents participated in the final round of ANAC'11 are available at http://anac2012.ecs.soton.ac.uk/Negotiator3.2.1.zip

Table I: Average utility of each agent against all opponents in different negotiation domains

	Itex vs Cypress domain	Grocery domain
ABiNeS	**0.445**	**0.803**
HardHeaded	0.397	0.710
Gahboninho	0.424	0.662
IAMHaggler'11	0.343	0.690
BRAM Agent	0.377	0.664
AgentK2	0.425	0.687
The Negotiator	0.331	0.693
Nice Tit for Tat	0.414	0.713
ValueModelAgent	0.366	0.759

the characteristics of these domains. The more competitive the domain is, the less utilities the agents can receive during negotiation.

Table II: Negotiation results in the domain of Itex vs Cypress

		Hard-Headed	Gahbo-ninho	IAM-Haggler	BRAM-Agent	Agent-K2	The Ne-go-tia-tor	Nice Tit for Tat	Value Model Agent	ABi-NeS
ABi-NeS	utility	0.500	0.583	0.357	0.320	0.430	0.499	0.510	0.357	X
	time	0.509	0.413	0.998	0.998	0.995	0.635	0.478	0.998	X
HardHeaded	utility	X	0.294	0.328	0.350	0.547	0.454	0.465	0.295	0.439
	time	X	0.998	0.835	0.834	0.505	0.580	0.523	0.990	0.509
Gahboninho	utility	0.369	X	0.441	0.277	0.450	0.521	0.561	0.324	0.451
	time	0.998	X	0.542	0.999	0.404	0.939	0.314	0.999	0.413
IAMHaggler'11	utility	0.393	0.471	X	0.363	0.510	0.475	0.356	0.356	0.246
	time	0.835	0.542	X	0.91	0.575	0.671	0.971	0.996	0.998
BRAMAgent	utility	0.404	0.38	0.373	X	0.443	0.431	0.321	0.277	0.39
	time	0.834	0.999	0.91	X	0.836	0.836	0.903	0.981	0.998
AgentK2	utility	0.425	0.566	0.408	0.303	X	0.333	0.498	0.387	0.480
	time	0.505	0.404	0.575	0.836	X	0.553	0.489	0.613	0.995
TheNegotiator	utility	0.475	0.098	0.366	0.263	0.533	X	0.459	0.219	0.24
	time	0.58	0.939	0.671	0.836	0.553	X	0.6	0.862	0.635
NiceTitforTat	utility	0.457	0.560	0.355	0.356	0.496	0.458	X	0.260	0.37
	time	0.523	0.314	0.971	0.903	0.489	0.604	X	0.981	0.478
ValueModelAgent	utility	0.391	0.270	0.247	0.353	0.510	0.459	0.373	X	0.324
	time	0.990	0.999	0.996	0.981	0.613	0.862	0.981	X	0.998

Table III: Negotiation results in the domain of Grocery

		Hard-Headed	Gahbo-ninho	IAM-Haggler	BRAM-Agent	Agent-K2	The Ne-go-tia-tor	Nice Tit for Tat	Value Model Agent	ABi-NeS
ABiNeS	utility	0.79	0.8213	0.804	0.810	0.818	0.821	0.779	0.779	X
	time	0.995	0.757	0.710	0.972	0.775	0.878	0.998	0.996	X
HardHeaded	utility	X	0.838	0.781	0.707	0.821	0.741	0.627	0.600	0.566
	time	X	0.757	0.769	0.813	0.765	0.814	0.999	0.88	0.995
Gahboninho	utility	0.4456	X	0.7597	0.696	0.760	0.658	0.658	0.658	0.658
	time	0.757	X	0.757	0.758	0.757	0.757	0.758	0.758	0.757
IAMHaggler'11	utility	0.680	0.696	X	0.76	0.72	0.741	0.661	0.755	0.51
	time	0.769	0.757	X	0.501	0.445	0.335	0.998	0.482	0.710
BRAMAgent	utility	0.614	0.751	0.818	X	0.788	0.742	0.5	0.718	0.383
	time	0.813	0.758	0.501	X	0.520	0.816	0.904	0.789	0.972
AgentK2	utility	0.656	0.668	0.809	0.711	X	0.665	0.675	0.72	0.595
	time	0.765	0.757	0.445	0.520	X	0.71	0.898	0.998	0.775
TheNegotiator	utility	0.703	0.814	0.892	0.705	0.755	X	0.559	0.632	0.480
	time	0.814	0.757	0.335	0.816	0.710	X	0.828	0.819	0.878
NiceTitforTat	utility	0.733	0.821	0.57	0.810	0.821	0.766	X	0.716	0.465
	time	0.999	0.758	0.998	0.904	0.898	0.828	X	0.938	0.998
ValueModelAgent	utility	0.808	0.871	0.828	0.774	0.772	0.758	0.636	X	0.624
	time	0.88	0.758	0.482	0.789	0.998	0.819	0.938	X	0.996

Table II shows the detailed negotiation results of each agent against all other negotiating partners in the domain of Itex vs Cypress. We can see that the *ABiNeS* agent can more efficiently exploit most of the negotiating partners and thus obtain higher utility by reaching an agreement earlier or later compared with other agents. For example, consider the case of negotiating with the *Gahboninho* agent, the *ABiNeS* agent reaches an agreement with it in the middle of the negotiation period and obtain the utility of 0.583. However, the *HardHeaded* agent does not make timely concession to the *Gahboninho* agent, and it ends up with a very low utility of 0.294 due to discounting effects.

The detailed negotiation results of each agent against different negotiating partners in the domain of Grocery are given in Table III. We can observe that *ABiNeS* agent can always achieve high utility against most of negotiating opponents compared with other agents. Specifically, the *ABiNeS* agent can make better prediction on the appropriate time to accept the negotiating parter's offer and thus obtain relatively high utility in most cases. For example, consider negotiating against the *IAMHaggler'11* agent. The reason that the *ABiNeS* agent's utility is higher than that of the *HardHeaded* agent is that for the same proposal offered from the *IAMHaggler'11* agent, the *ABiNeS* agent chooses to accept it a little bit earlier than the *HardHeaded* agent. Similarly, when negotiating against the *ValueModelAgent* agent, the *ABiNeS* agent can successfully predict that it is better to not accept the offer from the *ValueModelAgent* agent until the very last moment, and thus receives higher utility than the *HardHeaded* agent. Another interesting observation is that when negotiating against the *NiceTitforTat* agent, the *ABiNeS* agent gets higher utility than the *HardHeaded* agent though both of them choose to not concede to the *NiceTitforTat* agent until the last moment ($t \approx 0.999$). We hypothesise that it is due to the fact that the *NiceTitforTat* agent's strategy is highly dependent on the behavior of its negotiation partner, and the the *ABiNeS*'s negotiation moves induce it to become more willing to make larger concessions.

V. Related Work

Until now great effects have been devoted to develop efficient negotiation strategies for automated negotiating agents in the literature. The most typical approach existing in the literature is concession-based strategy such as ABMP strategy [17]. The ABMP strategy agent decides on the next move based on its own utility space only and makes concession to its negotiating partners according to certain concession pattern. However, since the dynamics and influences of the negotiating partners are not taken into consideration, the weakness of this type of strategies is that it is difficult to reach those mutually beneficial outcome and thus is inefficient in complex negotiation scenarios.

To overcome the aforementioned limitation, various techniques have been proposed to model certain aspects of the negotiation scenario to improve the efficiency of the negotiation outcome such as the opponent's preference profile and the knowledge of the negotiation domain [2]–[4], [12]. Saha et al. [3] propose a learning mechanism using Chebychev's polynomials to approximately model the negotiating opponent's decision function in the context of repeated two-player negotiations. They prove that their algorithm is guaranteed to converge to the actual probability function of the negotiating partner under infinite sampling. Experiments also show that the agent using their learning mechanism can outperform other simple learning mechanisms and also be robust to noisy data. However, in their approach, it is assumed that the agents negotiate over one indivisible item (price) only, thus is not applicable to more general multi-issue negotiation scenarios.

Hindriks and Tykhonov [4] propose a bayesian learning based technique to model the negotiating opponent's private preference in the context of bilateral multi-issue negotiations. They test this technique on several negotiation domains and show that it can improve the efficiency of the bidding process by incorporating it into a negotiation strategy. One application of this modeling technique is that it is integrated into a negotiation strategy called *TheNegotiator* [11] which participated in ANAC'11 [11].

Brzostowski and Kowalczyk [5] propose a mechanism for predicting the negotiating partner's future behaviors based on the difference method. Based on the prediction results, the negotiation can be modeled as multi-stage control process and the task of determining the optimal next-step offer is equivalent to the problem of determining the sequence of optimal control. Simulation results show that the agents using their mechanism can greatly outperform the classical approach in terms of utilities. However, their mechanism is only applicable in the single-item negotiation scenario. Besides, the underlying assumption of their mechanism is that the negotiation partner's strategy is the combination of time-dependent and behavior-dependent tactics, and thus may not work well against other types of negotiation partners.

VI. Conclusion and Future Work

In this paper, we propose an adaptive negotiation strategy *ABiNeS* for automated agents to negotiate in bilateral multi-issue negotiation scenarios. We introduce the concept of non-exploitation point λ to adaptively adjust the *ABiNeS* agent's concession degree to its negotiating opponent, and propose a reinforcement-learning based approach to determine the optimal proposal for the negotiating partner to maximize the possibility that the offer will be accepted by the opponent. As future work, one worthwhile direction is to further refine the estimation of the negotiating partner's concessive degree to further exploit the negotiating partner, by taking into consideration the magnitude of the utility that the negotiating partner proposes. Besides, we are going to further investigate the negotiation performance in ANAC 2012 and integrate the current existing negotiating techniques (i.e., preference modeling and strategy prediction) into the *ABiNeS* strategy under different combinations in different negotiation scenarios.

References

[1] G. M. Lai, C. H. Li, K. Sycara, and J. A. Giampapa, "Literature review on multi-attribute negotiations," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Technical Report CMU-RI-TR-04-66, December 2004.

[2] P. Faratin, C. Sierra, and N. R. Jennings, "Using similarity criteria to make negotiation trade-offs," *Artifical Intelligence*, vol. 142(2), pp. 205–237, 2003.

[3] S. Saha, A. Biswas, and S. Sen, "Modeling opponent decision in repeated one-shot negotiations," in *AAMAS'05*, 2005, pp. 397–403.

[4] K. Hindriks and D. Tykhonov, "Opponent modeling in auomated multi-issue negotiation using bayesian learning," in *AAMAS'08*, 2008, pp. 331–338.

[5] J. Brzostowski and R. Kowalczyk, "Predicting partner's behaviour in agent negotiation," in *AAMAS '06*, 2006, pp. 355–361.

[6] J. Y. Hao and H. F. Leung, "An efficient negotiation protocol to achieve socially optimal allocation," in *PRIMA'12*, 2012, pp. 46–60.

[7] D. Zeng and K. Sycara, "Bayesian learning in negotiation," *Internaltional Journal of Human Computer Systems*, vol. 48, pp. 125–141, 1998.

[8] R. M. Coehoorn and N. R. Jennings, "Learning an opponent's preferences to make effective multi-issue negotiation trade-offs," in *ICEC'04*, 2004, pp. 59–68.

[9] D. Zeng and K. Sycara, "Bayesian learning in negotiation," in *AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems*, 1996, pp. 99–104.

[10] T. Baarslag, K. Hindriks, C. Jonker, S. Kraus, and R. Lin, "The first automated negotiating agents competition (anac 2010)," *New Trends in Agent-Based Complex Automated Negotiations*, pp. 113–135, 2010.

[11] T. Baarslag, K. Fujita, E. H. Gerding, K. Hindriks, T. Ito, N. R. Jennings, C. Jonker, S. Kraus, R. Lin, V. Robu, and C. R. Williams, "Evaluating practical negotiating agents: Results and analysis of the 2011 international competition," *Artificial Intelligence Journal*, Accepted.

[12] T. Baarslag, K. Hindriks, and C. Jonker, "Acceptance conditions in automated negotiation," in *ACAN'11*, 2011.

[13] *The Third International Automated Negotiating Agent Competition (ANAC 2012)*, 2012, http://anac2012.ecs.soton.ac.uk/.

[14] R. M. Coehoorn and N. R. Jennings, "Learning an opponent's preferences to make effective multi-issue negotiation trade-offs," in *ICEC'04*. ACM Press, 2004, pp. 59–68.

[15] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, Cambridge, 1994.

[16] 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," *Comput. Intell.*, 2012.

[17] C. Jonker, V. Robu, and J. Treur, "An agent architecture for multi-attribute negotiation using incomplete preference information," *AAMAS*, vol. 15(2), pp. 221–252, 2006.