Reaching Good Agreements in Multilateral Agent-based Negotiations*

Siqi Chen, Jianye Hao, Shuang Zhou and Gerhard Weiss

Abstract Negotiations among autonomous agents has been gained a mass of attention from a variety of communities in the past decade. This paper deals with a prominent type of automated negotiations, namely, multilateral multi-issue negotiation that runs under real-time constraints and in which the negotiating agents have no prior knowledge about their opponents' preferences over the space of negotiation outcomes. We propose a novel negotiation approach which enables an agent to reach an efficient agreement with multiple opponents. The proposed approach achieves that goal by, 1) employing sparse pseudo-input Gaussian processes to model the behavior of opponents, 2) learning fuzzy opponent preferences to increase the satisfaction of other parties, and 3) adopting an adaptive decision-making mechanism to handle uncertainty in negotiation.

1 Introduction

Negotiation is ubiquitous in our daily life and serves as an important approach to facilitate conflict-resolving and reaching agreements between different parties. Development of automated negotiation techniques enables software agents to perform negotiations on behalf of human negotiators. This can not only significantly alleviate

---

Siqi Chen
School of Computer and Information Science, Southwest University, Chonqqing, China, e-mail: `siqi.chen09@gmail.com`

Jianye Hao
School of Software, Tianjin University, China, e-mail: `jianye.hao@tju.edu.cn`

Shuang Zhou and Gerhard Weiss
Department of Knowledge Engineering, Maastricht University, The Netherlands, e-mail: `{shuang.zhou,gerhard.weiss}@maastrichtuniversity.nl`

* This paper is a shortened version of our previous work [1]

the efforts of human negotiators, but also aid human in reaching better negotiation outcomes by compensating for the limited computational abilities of humans when they deal with complex negotiations.

During negotiations, a negotiating agent usually keeps its negotiation strategy and its preference as its private information to avoid being exploited. Thus one major research challenge is to effectively estimate the negotiation partner's preference profile [2, 3, 4, 5] and predicate its decision function [6, 7]. On one hand, through getting a better understanding of the negotiation partners' preferences, it would increase the 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 [8]. Until now, a lot of research efforts have been devoted to developing automated negotiation strategies and mechanisms in different negotiation scenarios [3, 7, 9, 10, 11, 12, 13]. Especially recent years have witnessed the emergence of a number of advanced negotiation strategies participated in the last few years' *automated negotiating agents competition (ANAC)* [8]. The ANAC competition provides a general and uniform negotiation platform which enables different negotiation agents to be evaluated against a wide range of opponents within various realistic negotiation environments. However, most research efforts have been devoted to bilateral negotiation scenarios, which only models the strategic negotiation among two parties. However, in real life the more common and general way of negotiations usually involve multiple parties. It is in common agreement from the automated negotiation research community that more attention should be given to multilateral negotiations and investigate effective negotiation techniques for multilateral negotiation scenarios.

In this paper, we propose a novel negotiation approach for automated agents to negotiate in multilateral multi-issue real-time negotiation environments. During negotiation, the agents' negotiation strategies and preference profiles are their private information, and the available information about the negotiating partner is its past negotiation moves. Due to the huge strategy space that a negotiating partner can consider, it is usually very difficult (or impossible) to predict which specific strategy the negotiating partner is using based on this limited amount of information. To this end, instead of predicting the exact negotiation strategies of the opponents, we adaptively adjust the non-exploitation point λ to determine the perfect timing that we should stop further exploits the opponents, and then determine the aspiration level (or the target utility) for proposing offers to opponents before and after the non-exploitation point following different rules. The value of λ is determined as the timing when the estimated expected future utility we can obtain over all opponents is maximized. The future utility that each opponent offers can be efficiently predicted using the Sparse Pesudo-inputs Gaussian Process (SPGP) technique by dividing the negotiation history into a number of atomic intervals.

Given the aspiration level for offering proposals, another important question is how should we select an optimal proposal to reach efficient agreements with other parties, which can also improve the possibility of accepting this offer by the negotiating partners. In this work, we measure the efficient degree of an outcome from a practical perspective – the social welfare of participants. We propose modeling the preferences of each opponent using the least square error regression technique based on the negotiation history. After that, the offer with the highest social welfare is selected as the offer to be proposed with certain exploration. We evaluate the performance of our strategy from two different perspectives: *efficiency* in terms of the average payoff obtained under a particular negotiation tournament and *robustness* in terms of how likely the agents have the incentive to adopt our strategy rather than other strategies. First, simulation results show that our strategy is more efficient against a variety of state-of-the-art negotiation strategies in both discounting and non-discounting domains with various domain sizes. Second, we evaluate the robustness of our strategy using empirical game-theoretic analysis. Experimental results show that our strategy is the most robust one compared with the existing state-of-the-art strategies. Moreover, a light-weight implementation of the proposed negotiation approach finished second in the category of Nash product in the ANAC 2015.

The remainder of the paper is organized as follows. Section 2 introduces the multilateral negotiation model we adopt. In Section 3, our negotiation approach is introduced in details. And conclusion and future work are given in Section 4.

2 Multilateral Negotiation Model

To govern the complex process of a multilateral negotiation, we adopt an extension of a basic bilateral negotiation protocol [14] which is widely used in the agents field [10, 11, 8, 15, 16]. The participating agents try to establish a contract for a product (service) or reach consensus on certain matter on behalf of their parties. Precisely, let $A = \{a_1, a_2, ..., a_i, ..., a_m\}$ be the set of negotiating agents, J be the set of issues under negotiation with j a particular issue ($j \in \{1, ..., n\}$ where m is the number of issues). Following the alternating bargaining model of [14], each agent, in turn, has a chance to express its opinion about the ongoing negotiation. The opinion can be communicated in a form of a contract proposal (e.g., a new offer), or an acceptance of the latest offer on the table (note that previous offers would not be accepted once there exists a new proposal), or terminating the negotiation according to its interpretation of the current negotiation situation. A simple illustration of the multilateral negotiation process is shown in Figure 1. Due to space constraints we refer the interested reader to the website of ANAC for more details about the protocol.

An offer is a vector of values, with one value for each issue. The utility of an offer for agent i is calculated by the utility function defined as follows:

$$U^i(O) = \sum_{j=1}^{n} (w^i_j \cdot V^i_j(O_{j,k})) \tag{1}$$

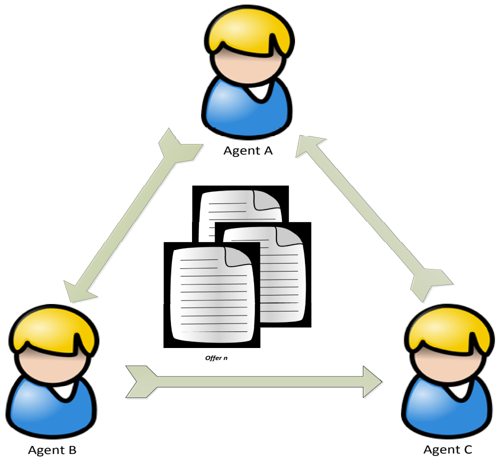

Fig. 1: Multilateral negotiation protocol.

where w^i_j and O are as defined above and V^i_j is the evaluation function of agent i for issue j, mapping every possible value of issue j (i.e., $O_{j,k}$) to a real number. The weight vector $\mathbf{w}$ denotes the weighting preference of an agent, in which w^i_j represents its preference for issue j. The issue weights of an agent are normalized (i.e., $\sum_{j=1}^{n} w^i_j = 1$ for each agent i). In addition an agent has a lowest expectation for the outcome of a negotiation – the reservation value ϑ.

In this work we consider negotiation being conducted in a real-time way instead of being restricted by a fixed number of exchanged offers; specifically, each negotiator has a hard deadline by when it must have completed or withdraw the negotiation. The negotiation deadline of agents is denoted by $t_{\max}$. In negotiations under real-time constraints, the number of remaining rounds are not fixed and the outcome of a negotiation depends crucially on the time sensitivity of the agents' negotiation strategies. For domains where the value of agreements is discounted over time, the discounting factor δ ($\delta \in [0,1]$) is defined to calculate the discounted utility as follows:

$$D(U,t) = U \cdot \delta^t \quad (2)$$

where U is the (original) utility and t is the standardized time. As an effect, the longer it takes for agents to come to an agreement the lower is the utility they can obtain.

3 Negotiation approach

Our proposed approach consists of three core components: deciding aspiration level, generating new offers and responding mechanism, all of which are described in de-

Algorithm 1 The overview of the proposed negotiation approach. Let t_c be the current time point, δ the time discounting factor, and t_{max} the deadline of negotiation. O_{opp} is the latest opponent offer, Ω_i the previous offers of opponent i and O_{own} a new offer to be proposed by our agent. χ is the time series including the average utilities over intervals. E denotes the expected utility of incoming counter-offers. λ is the non-exploitation time point and u' the target utility. $\mathbf{W}$ denotes the set of learnt opponent weight vectors.

```
Require: ϑ, δ, t_max
while t_c <= t_max do
    O_opp ⇐ receiveMessage;
    Ω_i ⇐ recordOfferSet(t_c, O_opp, i);
    if myTurn(t_c) then
        if updateModel(t_c) then
            χ ⇐ preprocessData(t_c)
            E ⇐ Predict(χ, Ω);
            (λ, U_min) ⇐ updateParas(t_c);
            W = updatePrefreenceModels();
        end if
    end if
    u' = getTargetUtility(t_c, E, λ);
    O_own ⇐ constructOffer(u', W) ;
    if isAcceptable(u'_c, O_opp, t_c, δ) then
        accept(O_opp);
    else
        checkTermination();
        proposeNewBid(O_own);
    end if
end while
```

tail in this section. We first give an overview of our approach shown in *Algorithm* 1. Following that, the individual steps of *Algorithm* 1 are explained in details.

3.1 Deciding Aspiration Level

Aspiration level indicates the target utility of an agent in the negotiation process. In order to respond to uncertainty in a negotiation where opponents' private information is unknown, the aspiration level is updated due to the environment (e.g., available negotiation time and discounting effect) and opponent behaviors. The agent can therefore predict opponent future moves to assist its decision by analyzing past moves of the opponent. The prediction technique we use here is a computationally efficient variant of standard Gaussian Processes (GPs) – Sparse Pseudo-inputs Gaussian Processes (SPGPs), which proves effective in negotiation context [16]. Another advantage of SPGPs over other type of regression techniques is that it not only provides accurate prediction but also the measure of confidence in the prediction.

Following the notation of GPs in [17], given a data set $\mathscr{D} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{n}$ where $\mathbf{x} \in \mathbb{R}^d$ is the input vector, $y \in \mathbb{R}$ the output vector and m is the number of available data points when a function is sampled according to a GP, we write, $f(\mathbf{x}) \sim \mathscr{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$, where $m(\mathbf{x})$ is the mean function and $k(\mathbf{x}, \mathbf{x}')$ the covariance function, fully specifying a GP. Learning in a GP setting involves maximizing the marginal likelihood of Equation 3.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^T \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{y} - \frac{1}{2}\log|\mathbf{K} + \sigma_n^2 \mathbf{I}| - \frac{n}{2}\log 2\pi, \tag{3}$$

where $\mathbf{y} \in \mathbb{R}^{m\times 1}$ is the vector of all collected outputs, $\mathbf{X} \in \mathbb{R}^{m\times d}$ is the matrix of the data set inputs, and $\mathbf{K} \in \mathbb{R}^{m\times m}$ is the covariance matrix with $|.|$ representing the determinant.

To fit the hyperparameters that best suit the available data set we need to maximize the marginal likelihood function of Equation 3 with respect to Θ, the vector of all hyperparameters. The problem with GPs is that maximizing Equation 3 is computationally expensive due to the inversion of the covariance matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ where n is the number of data points. We for this specific reason employ a fast and more efficient learning technique – SPGPs. The most interesting feature of SPGPs is that these approximators are capable of attaining very close accuracy in both learning and prediction to normal GPs with only a fraction of the computation cost. This property makes them extremely suitable to the multilateral negotiation domain where a complex and low cost function approximation framework is highly demanded.

Using only a small amount of pseudo-inputs, SPGPs are capable of attaining very similar fitting and prediction results to normal GPs. To clarify, the idea is to parametrize the model by $M << n$ pseudo-input points, while still preserving the full Bayesian framework. This leads to the parametrization of the covariance function by the location of $M <<< n$ pseudo-inputs. These are then fitted in addition to the hyperparameters in order to maximize the following new marginal likelihood:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}, \bar{\mathbf{X}}, \Theta) &= \int p(\mathbf{y}|\mathbf{X}, \bar{\mathbf{X}}, \bar{\mathbf{f}}) p(\bar{\mathbf{f}}|\bar{\mathbf{X}}) d\bar{\mathbf{f}} \\ &= \mathscr{N}(\mathbf{y}|\mathbf{0}, \mathbf{K}_{NM}\mathbf{K}_M^{-1}\mathbf{K}_{MN} + \blacksquare + \sigma^{\mathbf{2}}\mathbf{I}), \end{aligned} \tag{4}$$

where $\bar{\mathbf{X}}$ is the matrix formed by the pseudo-inputs with $\bar{\mathbf{X}} = \{\bar{\mathbf{x}}\}_{m=1}^{M}$. $\mathbf{K}_{NM}$ is the covariance matrix formed by the pseudo and the real inputs as $\mathbf{K}_{MN} = k(\bar{\mathbf{x}}_m, \mathbf{x}_n)$ with $k(.,.)$ being the covariance kernel. $\mathbf{K}_M^{-1}$ is the inverse of the covariance matrix formed among the pseudo inputs with $\mathbf{K}_M = k(\bar{\mathbf{x}}_m, \bar{\mathbf{x}}_m)$. Λ is a diagonal matrix having the diagonal entries of $\lambda_n = k_{nn} - \mathbf{k}_n^T \mathbf{K}_M^{-1} \mathbf{k}_n$. The noise variance and the identity matrix are represented by σ and $\mathbf{I}$, respectively.

When a counter-proposal from agent i arrives at time t_c, our agent records the time stamp t_c and the utility $U(O^i)$ that is evaluated in our agent's utility space. To reduce misinterpretation of the opponent's behavior as much as possible that is caused by the setting of multi-issue negotiations, the whole negotiation is divided into a fixed number (denoted as ζ) of equal intervals. The average utilities at each interval with the corresponding time stamps, are then provided as inputs to the SPGPs. Results in [18] show a complexity reduction in the training cost (i.e., the cost of finding the parameters of the covariance matrix) to $\mathcal{O}(M^2N)$ and in the prediction cost (i.e., prediction on a new set of inputs) to $\mathcal{O}(M^2)$. The results further demonstrate that SPGPs can fully match normal GPs with small M (i.e., few pseudo-inputs), successfully producing very sparse solutions.

After learning a suitable model, SPGPs makes forecast about the future concession of the opponent as shown in line 7 of Algorithm 1. Our agent keeps track of the expected discounted utility based on the predictive distribution at a new input $t_\star$, which is given by:

$$p(u_*|t_\star,\mathscr{D},\bar{\mathbf{X}}) = \int p(u_\star|t_\star,\bar{\mathbf{X}},\bar{\mathbf{f}})p(\bar{\mathbf{f}}|\mathscr{D},\bar{\mathbf{X}})d\bar{\mathbf{f}} = \mathscr{N}(u_\star|\mu_\star,\sigma_*^2), \tag{5}$$

where

$$\mu_\star = \mathbf{k}_\star^T\mathbf{Q}_M^{-1}(\Lambda+\sigma^2\mathbf{I})^{-1}u$$
$$\sigma_\star^2 = \mathbf{K}_{\star\star} - \mathbf{k}_\star^T(\mathbf{K}_M^{-1}-\mathbf{Q}_M^{-1})\mathbf{k}_\star + \sigma^2$$
$$\mathbf{Q}_M = \mathbf{K}_M + \mathbf{K}_{MN}(\Lambda+\sigma^2\mathbf{I})^{-1}\mathbf{K}_{NM}$$

With the given probability distribution over future received utilities and the effect of the discounting factor, the expected utility $E_{t_\star}$ is then formulated by

$$E_t = \frac{1}{C}\int_0^1 D(u\cdot p(u;\mu_t,\sigma_t),t)du \tag{6}$$

where $\mu_\star$ and $\sigma_\star$ are the mean and standard deviation at time $t_\star$, and the normalizing constant C is introduced to preserve a valid probability distribution.

Our agent employs the target utility function as given in Equation 7 to determine the aspiration level over time. The function adopts a tough manner (i.e., slowly conceding) before the non-exploitation time point (λ) for seeking higher expected profits, then it quickly goes to the expected minimal utility such that negotiation failure/disagreement could be avoided in the end. The non-exploitation time point is adjusted according to the behavior of other negotiation participants. More precisely, the higher the average opponent concession (measured in the our own utility space), the later our agent begins to compromise.

$$u' = \begin{cases} U_{\max} - \Delta(\frac{t_c}{\lambda})^{1+\delta} & when\ \ t_c \le \lambda, \\ (U_{\max}-\Delta)(1-\frac{t_c-\lambda}{t_{\max}-\lambda})^{1+\delta} & otherwise \end{cases} \tag{7}$$

where $U_{\max}$ is the maximal utility, $U_{\min}$ is the minimal utility ($U_{\min} = \max(\vartheta,\gamma)$ and γ the received lowest opponent concession), constant Δ is the maximal concession amount (i.e., $U_{\max} - U_{\min}$), with

$$\lambda = \operatorname*{argmax}_{t\in T} \frac{1}{|A|-1}\sum_{i\in A\setminus o}\frac{1}{C_i}\int_0^1 D_\delta(u\cdot p(u;\mu_t,\sigma_t),t)du \tag{8}$$

with o representing our agent and $T \in [t_c, t_{\max}]$.

3.2 Generating Offers

Given an aspiration utility level to achieve, our agent next needs to consider what offer to send such that the likelihood of an offer being accepted could be maximized. Performing this task would require certain knowledge about opponents' preferences. However, negotiation opponents unfortunately have no motivation to reveal their true likings over proposals (or their utility functions) to avoid exploitation. In order to address this problem, we model the opponent concession tactics as time-dependent tactics (originated in [15]) shown in Equation 9, which are classic tactic in the current literature.

$$\widetilde{u} = U_{\max} - (U_{\max} - \vartheta)(t_c/t_{\max})^{\alpha} \tag{9}$$

where α is the concession factor controlling the style of concessive behavior (e.g., boulware ($\alpha < 1$), conceder ($\alpha > 1$) or linear ($\alpha = 1$)). Time-dependent tactics are widely used in automated negotiation community to decide concession toward opponents since an negotiator needs to make more or less compromise over time so as to resolve conflicts of the parties. In more detail, boulware tactic maintains the target utility level until the late stage of a negotiation process, whereupon it concedes to the reservation utility. By contrast, conceder tactic makes quick compromise to other parties once a negotiation session starts. For linear tactic, it simply reduces the target utility from the maximal utility to the reservation utility in a linear way.

Learning opponent preferences, while useful, is indeed challenging because information about opponent preferences over different issues (e.g., the weight vector **w**) is severely lacking. To tackle this issue, researchers typically assume that opponent concession tactic is fully known or preferences follow a certain distribution. In many real-world applications, it is however difficult or costly to acquire the exact information about opponent concession.[2] Therefore we make a mild assumption that we could enquire of domain experts about the approximate concession range of an opponent. This fuzzy knowledge is provided in form of a pair of concession factors that indicate the upper and lower concession an opponent makes at each time point. This idea is illustrated in Figure 2. Thus, the agent can estimate opponent preferences with the aid of the fuzzy information about opponent concession. Specifically, the preferences are learnt through minimizing the loss function L, which gives the expected loss associated with estimating opponent concession based on a weight vector. The loss function is constructed as in Equation 11. The loss is calculated by the difference between the mean of concession and the utility of an offer based on a weight vector **w**; moreover an additional penalty is imposed by φ when an expected utility for **w** excesses the upper and lower bounds of opponent concession. When calculating the utility of an offer for opponent i, yet the valuation of each issue choice is needed. We here simply assume that the importance order of issue choices is known, and approximate the valuation like [3] as follow,

[2] Note that the opponent concession is the amount of concession measured in the utility space of the opponent instead of ours.

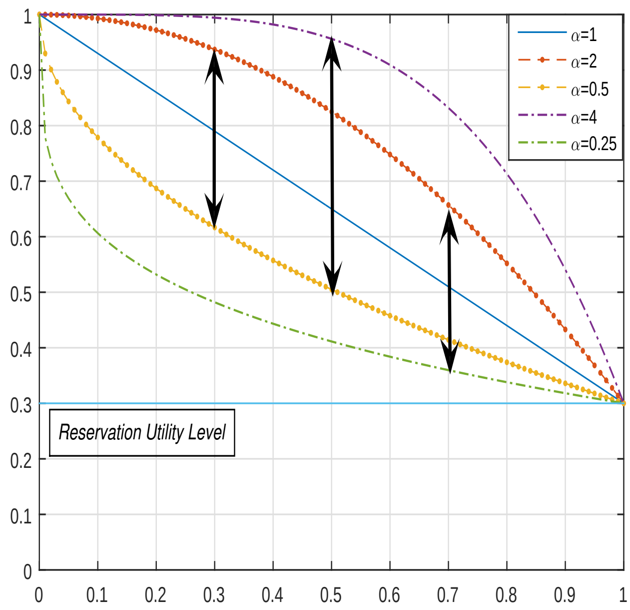

Fig. 2: A toy example of opponent concession ranges given by the pairs of concession factors (0.5,2) at time 0.3, (0.5,4) at 0.5 and (2,0.25) 0.7, respectively.

$$V^i_{j,k}(O_{j,k}) = \frac{2r^i_{j,k}}{K(K+1)} \tag{10}$$

where K is the number of possible choices for issue j, while $r^i_{j,k}$ denotes the ranking of the issue choice $O_{j,k}$.

Let the opponent utility of an offer for a weight vector $\mathbf{w}$ be $\hat{u}_{\mathbf{w}}$. With the opponent concession tactic given in Equation 9 and the two concession factors (which denote the approximate concession range suggested by experts), our agent can estimate the weight vector of opponent i by means of linear least squares. This can be achieved by minimizing the following loss function,

$$L^i(\mathbf{w}) = \begin{cases} |\frac{(u^i_{upper}+u^i_{lower})}{2} - \hat{u}_{\mathbf{w}}| + \varphi(u^i_{lower}, \hat{u}_{\mathbf{w}}), & \hat{u}_{\mathbf{w}} \leq u^i_{lower} \\ |\frac{(u^i_{upper}+u^i_{lower})}{2} - \hat{u}_{\mathbf{w}}| + \varphi(\hat{u}_{\mathbf{w}}, u^i_{upper}), & u^i_{upper} \leq \hat{u}_{\mathbf{w}} \\ |\frac{(u^i_{upper}+u^i_{lower})}{2} - \hat{u}_{\mathbf{w}}|, & otherwise \end{cases} \tag{11}$$

with u^i_{upper} and u^i_{lower} being the upper and lower bound of concession made by opponent i at time t, and φ the penalty function as below,

$$\varphi(x,y) = \beta|x-y|^{\frac{1}{2}} \tag{12}$$

where β denotes the confidence of the expert, and the lower the value, the more confidence the expert has about the perdition (to limit further complexness, we let β be 1).

After the estimation of weight vectors of other parties has been done, our agent chooses an offer being capable of maximizing the social welfare (e.g., the sum of the utility of all participants in the negotiation) given a aspiration level, shown as below:

$$\begin{aligned} &\underset{O}{\text{argmax}} \frac{1}{|A|-1} \sum_{i \in A \setminus o} (\hat{u}^i_{\mathbf{w}}(O) - \vartheta)^2 \\ &\text{subject to} \\ &U^o(O) \geq u' \end{aligned} \tag{13}$$

Although opponent preferences could be learnt on the basis of the provided concession tactics, it sometimes may be ineffective due to the fuzzy nature of the information; therefore our agent needs an alternative approach to choosing new offers. Fortunately, a real-time negotiation typically allows agents to exchange a large number of offers, thereby giving them many opportunities to explore the outcome space. Therefore, the proposed approach generates a new offer for next round following an ε-greedy strategy. The strategy selects either a greedy action (i.e., exploit) with 1-ε probability ($\varepsilon \in [0,1]$) or a random action with a probability of ε. It is worth noting that random action means choosing one offer from the set whose utility is above the given aspiration level by chance. The greedy action aims at choosing an offer that are expected to satisfy other sides' preferences most in order to improve their utilities over the negotiation outcome and the chance of the offer being accepted through fuzzy preference learning. With a probability $1-\varepsilon$, the approach randomly picks one of those offer whose utility is equal or larger than the given aspiration level. In the latter case, the agent constructs a new offer which has an utility within some range around u'. The reason is twofold: 1) it is possible, in multi-issue negotiations, to generate a number of offers whose utilities are the same or very similar to the offering agent, with granting the opposing negotiators different utilities, and moreover 2) it is sometimes not possible to make an offer whose utility is exactly equivalent to u'. Thus it is reasonable that an agent selects an offer whose utility is in the narrow range [(1-0.005)u',(1+0.005)u']. If no such solution can be found, our agent repeats the latest bid again in the next round.

3.3 Responding mechanism

This responding mechanism of the proposed approach corresponds to lines $15-20$ of Algorithm 1. After receiving a counter-proposal, the agent should decide whether to accept the proposal by checking two conditions. First the agent has to validate whether the utility of the latest counter-offer is better than u', while in the second the agent has to determine whether it had already proposed this offer (i.e., the opponent's counter-offer) earlier in the negotiation process. If either one of these two conditions is satisfied, the agent then accepts the offer as shown in line 16 and the negotiation will be completed if the proposal is also supported by the remaining agents.

Moreover, when the negotiation situation becomes hard and might offer our agent a utility even lower than the reservation utility, the agent should consider whether to terminate/leave the negotiation to receive the corresponding reservation utility or not. Here we treat the reservation value as an alternative offer from a negotiating partner with a constant utility. Thus the agent needs to check if the aspiration utility is smaller than the reservation utility. If positive, our agent is going to leave the negotiation table in the next round. If our agent decides neither to accept the latest counter-proposal nor to leave the negotiation, it proposes a new offer following the steps of lines 19 of Algorithm 1.

4 Conclusion

This work introduced a novel approach for multilateral agent-based negotiation in complex environments (multi-issue, time-constrained, and unknown opponents). Our proposed strategy, based on the adaptive decision-making scheme and the effective preference learning method, outperformed the top agents of the recent International Automated Negotiation Agents Competitions. Experiments show that our agent ont only generates a higher mean individual utility but also leads to better social welfare compared to the state-of-the-art negotiation agents. Further game-theoretic analysis clearly manifests the robustness of the proposed approach. We think the exceptional results justify to invest further research efforts into this approach. In the future work, we plan on comparing the opponent modeling scheme with the other available approachers and further, extend this framework to other negotiation settings like concurrent negotiation negotiation.

References

1. S.Chen, J.Hao, G.Weiss, S.Zhou, and Z.Zhang, “Toward efficient agreements in real-time multilateral agent-based negotiations,” in *ICTAI'15*, 2015 (In Reivew).
2. D. Zeng and K. Sycara, “Bayesian learning in negotiation,” *Internaltional Journal of Human Computer Systems*, vol. 48, pp. 125–141, 1998.
3. K. Hindriks and D. Tykhonov, “Opponent modeling in auomated multi-issue negotiation using bayesian learning,” in *AAMAS'08*, 2008, pp. 331–338.
4. 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.
5. T. Baarslag, K. Hindriks, and C. Jonker, “Acceptance conditions in automated negotiation,” in *ACAN'11*, 2011.

6. 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.
7. B. Jakub and K. Ryszard, "Predicting partner's behaviour in agent negotiation," in *AAMAS '06*, 2006, pp. 355–361.
8. 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*, vol. 198, pp. 73–103, 2013.
9. S. Chen and G. Weiss, "An efficient and adaptive approach to negotiation in complex environments," in *Proceedings of the 20th European Conference on Artificial Intelligence*. Montpellier, France: IOS Press, 2012, pp. 228–233.
10. C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings, "Negotiating concurrently with unkown opponents in complex, real-time domains," in *ECAI'12*, 2012, pp. 834–839.
11. J. Hao, S. Song, H.-F. Leung, and Z. Ming, "An efficient and robust negotiating strategy in bilateral negotiations over multiple items," *Eng. Appl. Artif. Intell.*, vol. 34, pp. 45–57, Sep. 2014.
12. S. Chen and G. Weiss, "An approach to complex agent-based negotiations via effectively modeling unknown opponents," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2287–2304, 2015.
13. S. Chen, H. B. Ammar, K. Tuyls, and G. Weiss, "Using conditional restricted boltzmann machine for highly competitive negotiation tasks," in *Proceedings of the 23th Int. Joint Conf. on Artificial Intelligence*. AAAI Press, 2013, pp. 69–75.
14. A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica*, vol. 50, no. 1, pp. 97–109, 1982.
15. P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents." *Rob. Autom. Syst.*, vol. 24, no. 4, pp. 159–182, 1998.
16. S. Chen and G. Weiss, "An intelligent agent for bilateral negotiation with unknown opponents in continuous-time domains," *ACM Trans. Auton. Adapt. Syst.*, vol. 9, no. 3, pp. 16:1–16:24, Oct. 2014.
17. C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.
18. E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," in *Advances in Neural Information Processing Systems 18*. MIT press, 2006, pp. 1257–1264.