ECAI 2012
Luc De Raedt et al. (Eds.)

doi:10.3233/978-1-61499-098-7-228

An Efficient and Adaptive Approach to Negotiation in Complex Environments

Siqi Chen and **Gerhard Weiss** [1]

Abstract. This paper studies automated bilateral negotiation among self-interested agents in complex application domains which consist of multiple issues and real-time constraints and where the agents have no prior knowledge about their opponents' preferences and strategies. We describe a novel negotiation approach called **OMAC** (standing for "Opponent Modeling and Adaptive Concession") which combines efficient opponent modeling and adaptive concession making. Opponent modeling is achieved through standard wavelet decomposition and cubic smoothing spline, and concession adaptivity is achieved through dynamically setting the concession rate on the basis of the expected utilities of forthcoming counter-offers. Experimental results are presented which demonstrate the effectiveness of our approach in both discounting and non-discounting domains. Specifically, the results show that our approach performs better than the five top agents from the 2011 Automated Negotiation Agents Competition (ANAC).

1 INTRODUCTION

Since some years automated negotiation is achieving steadily growing attention as a mechanism for coordinating interaction among computational autonomous agents which are in a consumer-provider or buyer-seller relationship and thus typically have different interests over possible joint agreements. The main reason for this attention is the broad spectrum of potential applications of automated negotiation in a variety of domains and fields. This paper deals with automated bilateral multi-issue negotiation (e.g., [12]). Characteristic to this negotiation form is that two agents negotiate with the goal to agree on a profitable contract for a product or service, where the contract consists of multiple issues (e.g., price, quantity and quality) which are of conflictive importance for the negotiators. Specifically, the paper focuses on bilateral multi-issue negotiation in applications scenarios in which the agents have no prior information about their opponents – neither about their preferences (e.g., their issue ordering or weight vectors) nor about their offering and acceptance strategies. In addition to that, we concentrate on "negotiation with deadline and discount", that is, on negotiations under real-time constraints where the agents have to take into consideration at each time point the remaining negotiation time and try to avoid any unnecessary delay during negotiation. Last but not least we focus on computational efficiency of negotiation, because agents may have very restricted computing resources. Negotiation scenarios showing these characteristics are particularly challenging but are common in reality. For instance, in the case of open electronic sales platforms an agent may be engaged in bilateral multi-issue negotiations with other agents which it has never met before. Moreover, if a negotiation agent runs on a small device such as a cell phone then computational efficiency is crucial.

This paper introduces a novel negotiation approach called **OMAC** ("Opponent Modeling and Adaptive Concession") for those scenarios. It integrates two key aspects of successful negotiation: efficient opponent modeling and adaptive concession making. Opponent modeling realized by **OMAC** aims at predicting the utilities of an opponent's future counter-offers and is achieved through two standard mathematical techniques, namely, wavelet decomposition and cubic smoothing spline. Adaptive concession making is achieved through dynamically adapting the concession rate (i.e., the degree at which an agent is willing to make concessions in its offers) on the basis of the utilities of future counter-offers it expects according to its opponent model.

The remainder of this paper is structured as follows. Section 2 overviews important related work. Section 3 describes the standard negotiation environment underlying our research. Section 4 presents **OMAC**. Section 5 offers a careful experimental analysis of the approach. Finally, Section 6 identifies some important research lines induced by the described work.

2 RELATED WORK

For automated negotiation, a determining factor for the success of an autonomous agent is how well it can interpret the opponent's intention on the basis of the offers they exchanged during their negotiation. An intuitively obvious approach to this is to equip agents with the ability to build up an opponent model which captures the opponent's preference profile or negotiation strategy – interpretation of the opponent's intention could then be done with the help of this model. Opponent modeling, however, is very challenging in practice because an opponent normally has no motivation to reveal its own preferences and strategies [4, 13]. Many opponent modeling approaches are described in literature (see [9]), but a problem with them is that they make strong or unrealistic assumptions to guarantee effectiveness. Specifically, due to these assumptions existing approaches are not appropriate for the type of complex environment we are dealing with. Saha et al. [15] applies Chebychev polynomials to estimate the chance that the negotiation partner accepts an offer in repeated single-issue negotiations on the same domain against a particular opponent, where the opponent's response can only be acceptance or rejection. In [2], Brzostowski et al. investigate the online prediction of future counter-offers on the basis of the previous negotiation history by using differentials, thereby assuming that the opponent strategy is known to base on a mix of time- and behavior-dependent one. Hou [11] employs non-linear regression to predict

[1] Dept. of Knowledge Engineering (DKE), Maastricht University, NL, email: {siqi.chen, gerhard.weiss}@maastrichtuniversity.nl

the opponent's tactic (though in single-issue negotiation), thereby supposing that the opponent uses a pure tactic as introduced in [7] and that the types of tactics are fixed. In [3] an artificial neural network (ANN) is constructed with three layers (including 52 neurons in total) to compete against human negotiators in a specific domain, its training however requires a very large database of previous encounters. Another notable work in this area is [16], where Gaussian processes are applied to predict the future opponent concession. The resulting information is then used to adapt the agent's concession strategy accordingly to achieve the "optimum" outcome. This strategy performed better than the best negotiating agents of ANAC 2010 and finally made the 3rd place in ANAC 2011 (ANAC is the International Automated Negotiation Agents Competition [1]). Our approach, **OMAC**, is empirically evaluated against this and other state-of-the-art agents; details are given in Section 5.

3 NEGOTIATION ENVIRONMENT

We adopt a basic bilateral multi-issue negotiation setting which is widely used in the agents field (e.g., [7, 8]) and the negotiation protocol we use is based on a variant of the alternating offers protocol proposed in [13]. Let $I = \{a, b\}$ be a pair of negotiating agents, i represent a specific agent ($i \in I$), J be the set of issues under negotiation, and j be a particular issue ($j \in \{1, ..., n\}$ where n is the number of issues). The goal of a and b is to establish a contract for a product or service. Thereby a contract consists of a package of issues such as price, quality and quantity. Each agent has a lowest expectation for the outcome of a negotiation; this expectation is called reserved utility u_{res}. w_j^i ($j \in \{1, \ldots, n\}$) denotes the weighting preference which agent i assigns to issue j, where the weights of an agent are normalized (i.e., $\sum_{j=1}^{n}(w_j^i) = 1$ for each agent i). During negotiation agents a and b act in conflictive roles which are specified by their preference profiles. In order to reach an agreement they exchange offers O in each round to express their demands. Thereby an offer is a vector of values, with one value for each issue. The utility of an offer for agent i is obtained by the utility function defined as:

$$U^i(O) = \sum_{j=1}^{n}(w_j^i \cdot V_j^i(O_j)) \tag{1}$$

where w_j^i and O are as defined above and V_j^i is the evaluation function for i, mapping every possible value of issue j (i.e., O_j) to a real number.

Following Rubinstein's alternating bargaining model [14], each agent makes, in turn, an offer in form of a contract proposal. Negotiation is time-limited 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 this form of real-time constraints, the number of remaining rounds are not known and the outcome of a negotiation depends crucially on the time sensitivity of the agents' negotiation strategies. This holds, in particular, for discounting domains, that is, domains in which the utility is discounted with time. As usual for discounting domains, we define a so-called discounting factor δ ($\delta \in [0, 1]$) and use this factor to calculate the discounted utility as follows:

$$D(U, t) = U \cdot \delta^t \tag{2}$$

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 achieve.

After receiving an offer from the opponent, O_{opp}, an agent decides on acceptance and rejection according to its interpretation $I(t, O_{opp})$ of the current negotiation situation. For instance, this decision can be made in dependence on a certain threshold *Thres*^{i}: agent i accepts if $U^i(O_{opp}) \geq$ *Thres*^{i}, and rejects otherwise. As another example, the decision can be based on utility differences. Negotiation continues until one of the negotiating agents accepts or withdraws due to timeout.[2]

4 OMAC APPROACH

OMAC includes two core stages – opponent modeling and concession rate adaptation – as described in detail in 4.1 and 4.2, respectively. A third important stage of **OMAC**, its response mechanism to counter-offers, is described in 4.3. An overview of **OMAC** is given in *Algorithm 1*.

Algorithm 1 The **OMAC** approach. t_c refers to the current time, δ the time discounting factor, λ the layer of wavelet decomposition, ψ the wavelet function, and t_{max} the deadline of negotiation. O_{opp} is the latest offer of the opponent, and O_{own} the offer to be proposed by **OMAC**. χ represents the time series comprised of the maximum utilities over intervals. Let υ be the smooth component of λ-th order wavelet decomposition based on ψ, and α the predicted main tendency of χ. t_l is the time we preform prediction process and u_l is the utility of our most recent offer. u' is the target utility at time t_c. R is the reserved utility function.

1: $Require : t_{max}, \delta, \lambda, \psi, R$

2: **while** $t_c <= t_{max}$ **do**

3: $O_{opp} \Leftarrow \text{receiveMessage}();$

4: $\text{recordBids}(t_c, O_{opp});$

5: **if** $\text{needUpdate}(t_c)$ **then**

6: $\chi \Leftarrow \text{preprocessData}(t_c)$

7: $(\alpha, t_l, u_l) \Leftarrow \text{predict}(\chi, \lambda, \psi);$

8: **end if**

9: $u' = \text{getTarUtility}(t_c, t_l, u_l, \delta, \alpha, R);$

10: **if** $\text{getOwnUtility}(O_{opp}, t_c, \delta) \geq u'$ **then**

11: $accept(O_{opp});$

12: **else**

13: $O_{own} \Leftarrow \text{constructOffer}(u');$

14: $\text{proposeBid}(O_{own});$

15: **end if**

16: **end while**

4.1 Opponent modeling

According to **OMAC**, the aim of opponent modeling realized by a negotiating agent is to estimate the utilities of future counter-offers it will receive from its opponent. This corresponds to the lines 3 to 8 in *Algorithm 1*. Opponent modeling is done through a combination of wavelets analysis and cubic smoothing spline. When receiving a new bid from the opponent at the time t_c, the agent records the time stamp t_c and the utility $U(O_{opp})$ this bid has according to the agent's utility function. The maximum utilities in consecutive equal time intervals and the corresponding time stamps are used periodically as basis for predicting the opponent's behavior (line 5 and 6). The reasons for a periodical updating are similar to those mentioned in [16].

[2] If the agents know each other's utility functions, they can compute the Pareto-optimal contract [13]. However, a negotiator will not make this information available to its opponent in general.

Firstly, this degrades the computation complexity so that the agent's response time is kept low. Assume that all observed counter-offers were taken as inputs, then the agent might have to deal with thousands of data points in every single session. This computational load would have a clear negative impact on the quality of negotiation in a real-time constraint setting. Secondly, the effect of noise can be reduced. In multi-issue negotiation a small change in utility of the opponent can result in a large utility change for the negotiator and this can easily result in a misinterpretation of opponent's behavior.

Behavior prediction is mainly done by applying discrete wavelet transformation (DWT) to the time series χ; this is captured by line 7. We decided to use DWT because wavelet analysis is known to be an efficient multi-scaling tool for exploring features in data sets. With DTW a signal can be decomposed into two parts, an approximation and a detail part. The former is smooth and reveals the trend of the original signal, and the latter is rough and corresponds to noise (resulting e.g. from seasonal fluctuations). **OMAC** focuses on the approximation part and intentionally ignores the detail part for three reasons. First, the approximation part represents the trend of the opponent concession in terms of utility and indicates how the concession of opponent will develop in the future. Second, it is smooth enough (compared to the original signals, i.e. χ) to allow for quality prediction performance. Third, the detail part contains information which is of little value in a negotiation setting. As we saw in various empirical investigations, the ratio between the main tendency term and the original signal tends to be about 0.98 with a small standard deviation. Precise extension of those detailed components can improve effectiveness of our model slightly, it is however very costly for a medium-range lead time in real-time negotiation.

Given the discrete wavelet function $\psi_{j,k}(t)$ transformed by a mother wavelet $\psi(t)$,

$$\psi_{j,k}(t) = a_0^{-j/2}\psi(a_0^{-j}t - kb_0), \quad j,k \in Z \tag{3}$$

DWT corresponds to a mapping from the signal $f(t)$ to coefficients $C_{j,k}$ which are related to particular scales, where these coefficients are defined as follows:

$$C_{j,k} = \int_{-\infty}^{+\infty} f(t)\overline{\psi_{j,k}(t)}dt, \quad j,k \in Z \tag{4}$$

The $\psi(t)$ is normally required to be an orthogonal wavelet in practice, the set $\{\psi_{j,k}(t) | j,k \in Z\}$ is then an orthogonal wavelet basis such that the signal $f(t)$ can be reconstructed.

With recursive application of DWT to the signal $f(t)$, the approximation (low frequency) and detail (high frequency) components are recovered, respectively. For instance, f can first be decomposed into $a_1 + d_1$ and the resulting part a_1 can then be decomposed in finer components, that is, $a_1 = a_2 + d_2$, and so on. Based upon this recursive process, the signal can be expressed as $f = a_1 + a_2 + \ldots + a_n + d_n$ (further details on wavelets are given in e.g. [5]). The results reported in this paper are achieved through wavelet decomposition using the Daubechies' wavelets of order 10. We use the following notation:

$$\chi = \upsilon + \sum_{n=1}^{\lambda} d_n \tag{5}$$

where υ represents the approximation component of χ and d_n is n-layer detail part (n is determined by the decomposition level λ). An example can be found in Figure 1 which shows χ and its corresponding approximation part υ along with the estimated upper and lower bounds of χ. The two bounds are represented by $\upsilon \pm \sigma$, where σ is the standard deviation of the ratio between χ and υ.

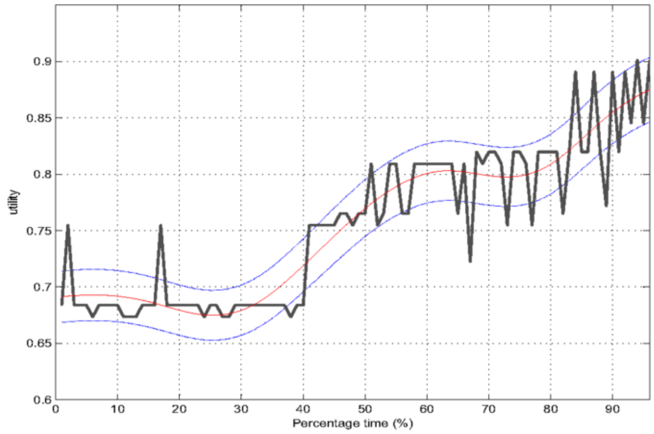


Figure 1. Illustrating the opponent's concession (given by χ, the thick solid line) and the corresponding approximation part υ (the thin solid line) when negotiating with Agent_K2 in the $Camera$ domain. The two dash-dot lines represent the estimated upper and lower bounds of χ. (Details of the agents and domains are given in Section 5.1.)

In order to forecast the opponent's future behavior, cubic smoothing spline is used to extend the smooth component υ . Cubic spline is widely used as a tool for prediction, see [17]. For equally spaced time series, a cubic spline is a smoothing piecewise function, denoted as the function $\hat{g(t)}$ which minimizes:

$$p\sum_{t=1}^{n} w(t)(f(t) - \hat{g}(t))^2 + (1-p)\int (\hat{g}(u)'')^2 du \tag{6}$$

where p is the smoothing parameter controlling the rate of exchange between the residual error described by the sum of squared residuals and local variation represented by the square integral of the second derivative of g and w is the weight vector (for further details, refer to [6]).

Figure 2 shows the actual and the predicted smooth parts of opponent concession at different time points for the opponent "Iamhaggler2011": as this figure illustrates, cubic spline is able to forecast the given signal within a medium range very well. Since **OMAC** applies a periodical updating mechanism, it is not necessary and not wise to forecast globally (i.e., from the current moment to the end point of negotiation), because this probably brings too much noise into the prediction. **OMAC** limits the range of forecasting to ζ intervals and in this way achieves efficiency and noise reduction.

4.2 Adaptive adjustment of concession rate

Given the extended version of the smooth part – α, we now discuss how to use it for adaptively setting the concession rate of our expected utility (see line 9 in *Algorithm 1*). A possibility is to maximize the expected utility merely according to the predicted opponent move. This is quite straightforward but may be not so effective. Suppose the negotiation partners are "tough" and always avoid making any concession in bargaining. In this case the result of prediction could indicate a very low expectation about the utility offered by the opponent and this, in turn, would result in an adverse concession. In our approach a simple function *R*, called reserved utility function, is used to realize concession adaptation. This function guarantees the minimum utility at each given time step. This is because the function values are set as the lower bound of our expected utilities. Moreover, in principle it makes concession over time, thereby taking into account the impact of the discounting factor. Specifically, the reserved

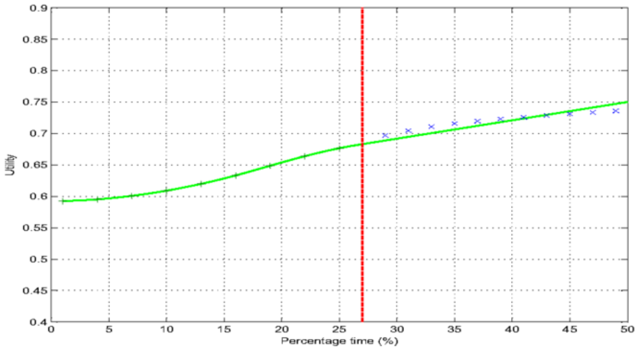


Figure 2. Illustration of the predictive power of our approach in two consecutive ranges. The dash line indicates the time point t_c at which the current prediction is made. The plus signs on left of the dash line are the actual points of υ before t_c. The crosses to the right of the dash line show the actual points of υ after t_c. The extended version of $\upsilon - \alpha$ (i.e., the prediction of υ) is shown by the solid line. These results are achieved for agent Iamhaggler2011 in the domain $Amsterdam\ party$.

utility function is given by:

$$R(t) = u_{res} + (1 - t^{1/\beta})(\mathit{maxUtility(p)} \cdot \delta^{\eta} - u_{res}) \quad (7)$$

where u_{res} is the minimum utility the agent would accept, β is a parameter which has a direct impact on the concession rate, *maxUtility(p)* is the function specifying the maximum utility given by the preference profile p of a negotiation domain, and η is a parameter called risk factor which reflects the agent's expectation about the maximum utility it can achieve.

We define the estimated received utility $E_{ru}(t)$, which gives our agent the expectation of opponent's future concession, as follows:

$$E_{ru}(t) = D(\alpha(t)(1 + \mathit{Stdev}(ratio_{[t_b,t_c]})), t), \quad t \in [t_c, t_s] \quad (8)$$

where $\mathit{Stdev}(ratio_{[t_b,t_c]})$ is the standard deviation of ratio between the smooth part υ and the original signal χ from the beginning of negotiation(t_b) till now and t_s is the end of α.

Suppose the future expectation the agent has obtained from $E_{ru}(t)$ is optimistic, in other words, there exists an interval $\{T | T \neq \varnothing, T \subseteq [t_c, t_s]\}$, so that

$$E_{ru}(t) \geq D(R(t), t), \;\; t \in T \quad (9)$$

OMAC then sets the time $\hat{t}$ at which the optimal estimated utility $\hat{u}$ is reached as:

$$\hat{t} = argmax_{t \in T}(E_{ru}(t) - D(R(t), t)) \quad (10)$$

and $\hat{u}$ is simply assigned by:

$$\hat{u} = E_{ru}(\hat{t}) \quad (11)$$

When the opponent's future concession is estimated to be below the agent's expectations according to $R(t)$ (i.e., there is no such interval T described above), **OMAC** investigates whether the best possible outcome under that "pessimistic" expectation of opponent concession should be accepted given the threshold ρ. This outcome is denoted as ξ and is given by:

$$\xi = \rho^{-1} \cdot E_{ru}(t_\xi)/D(R(t_\xi), t_\xi), \quad t_\xi \in [t_c, t_s] \quad (12)$$

where ρ is the tolerance threshold to accept $E_{ru}(t_\xi)$ as target utility and t_ξ is given by:

$$t_\xi = argmin_{t \in [t_c, t_s]}(|E_{ru}(t) - D(R(t), t)|) \quad (13)$$

The rationality behind it is that if the agent rejects the "locally optimal" counter-offer, the agent will probably loose the opportunity to reach a "globally good" agreement (especially in discounting domains). If $\xi > 1$, $\hat{u}$ and $\hat{t}$ are assigned to $E_{ru}(t_\xi)$ and t_ξ, respectively. Moreover, the agent records the utility and time of its last bid as u_l and t_l, respectively. Otherwise, the estimated utility is set to -1, meaning it does not take effect anymore, and $D(R(t_c), t_c)$ is used to set the target utility u'.

When the agent expects to achieve better outcomes (see Equation 9), the optimal estimated utility $\hat{u}$ is chosen as the target utility for our agent's future bids. Obviously, it is not rational to concede immediately to $\hat{u}$ when $u_l \geq \hat{u}$, nor should it shift to $\hat{u}$ without delay given $u_l < \hat{u}$, especially because the predication may be not absolutely accurate. To simplify the strategy, **OMAC** applies a linear concession making and the concession rate is dynamically adjusted to grasp every chance to maximize its profit. Overall, the target utility u' is given as follows:

$$u' = \begin{cases} D(R(t), t) & if \;\; \hat{u} = -1 \\ \hat{u} + (u_l - \hat{u})\frac{t - \hat{t}}{t_l - \hat{t}} & otherwise \end{cases} \quad (14)$$

4.3 Response mechanism

The response stage corresponds to lines 10 to 15 in *Algorithm* 1. With the target utility u' known (Equation 14), the agent then needs to examine the counter-offer to see if the utility of that offer $U(O_{opp})$ is higher than the target utility. If so, it accepts this counter-offer and, with that, terminates the negotiation session. Otherwise, the agent construct a bid to be proposed next round whose utility is indicated by u'.

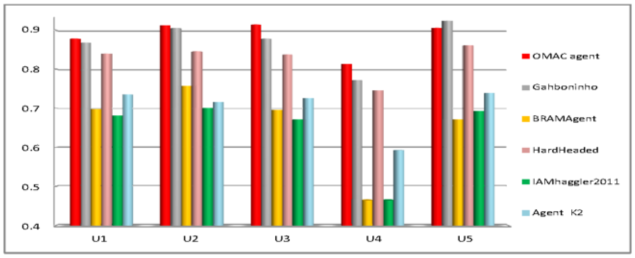
In multi-issue negotiation, offers with exactly the same utility for one side can have different values for the other party. Moreover, in time-limited negotiation scenarios no explicit limitation is imposed on the number of negotiation rounds and it is possible to generate many offers having a utility close to u'. **OMAC** takes advantage of this and aims at generating many offers in order to explore the space of possible outcomes and to increase the acceptance chance of own bids. Specifically, offers are constructed in such a way that the agent randomly selects an offer whose utility is in the range $[0.99u', 1.01u']$. If no such solution is found, the latest offer made by the agent is used again in the subsequent round. Moreover, in view of negotiation efficiency, if u' drops below the utility of the best counter-offer according to the agent's utility function, this best counter-offer is proposed by the agent as its next offer. This makes sense because the counter-offer tends to satisfy the expectation of opponent and is thus likely to be accepted by the opponent.

5 EXPERIMENTAL ANALYSIS

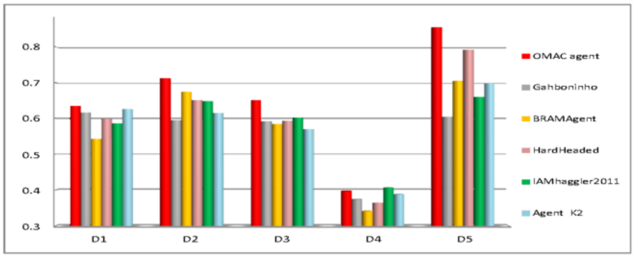
The performance evaluation of **OMAC** is done with GENIUS (General Environment for Negotiation with Intelligent multipurpose Usage Simulation [10]) which is also used as a competition platform for ANAC. It allows to compare agents (representing different negotiation strategies) across a variety of application domains under real-time constraints, where the preference profiles of two negotiating agents are specified for the individual domains.

5.1 Experimental settings of experiments

OMAC-agent is compared against the top five agents of ANAC 2011 (**OMAC-agent** refers to the agent which applies OMAC). Moreover, we use five application domains created for ANAC, all of



(a) Average raw scores of all agents in all domains without discount



(b) Average raw scores of all agents in all domains with discounting factor

Figure 3. Average raw scores of all agents in ten domains. The vertical axis shows utility and horizontal axis shows domain.

which are originally non-discounting except *Camera*. We refer to the non-discounting domain *Travel* as U_1, *Amsterdam party* as U_2, *England vs Zimbabwe* as U_3, *Itex vs Cypress* as U_4, *Camera* as U_5. The corresponding domains with time-dependent discounting factor are referred to as $D_1, D_2, ..., D_5$, respectively. Three of these domains were used in ANAC2010, and the others have been selected by ANAC2011. This choice of domains from ANAC2010 *and* 2011 makes the overall setting balanced (a wide range of domain characteristics are covered) and fair, and avoids any advantageous bias for **OMAC** resp. **OMAC-agent**. (With respect to fairness, note that the developers of the 2011 winners knew the ANAC2010 domains and thus had the opportunity to optimize their agents accordingly). For each of the ten domains, we run a tournament consisting of 6 agents 10 times to get results with high statistical confidence, where each agent repeats the negotiation against all other negotiators playing different roles in turn. In competition, agents will not be given any information about the opponents' strategies or other private information. Furthermore, they are prohibited from taking advantage of previous encounters with their opponents. The time limit for every single negotiation session is 180 seconds in total (to be shared by the two negotiators). In the implementation of **OMAC-agent**, ρ is set to 0.95 and the overall duration of a session is dividend into 100 consecutive intervals of 1.8 seconds each. In addition, the maximum predictive range ζ is limited to 15 intervals, the risk factor η is set to 0.2 and ψ = Daubechies wavelets of order 10 with λ being 4. (In our experiments we found out that **OMAC** works well for a very broad range of parameters.)

5.2 Experimental outcomes

Figure 3 depicts experimental results based on non-discounting domains (3(a)) and discounting domains (3(b)). As these figures show, **OMAC-agent** demonstrates excellent bargaining skills in each of the ten domains. More specifically, the agent finished first in eight domains and finished second in the remaining two domains (U5 and D4) with a very small disadvantage (about 2.35% at most). The domains used for testing cover a variety of domains in terms of three significant aspects. First, the outcome space ranges from large ($U1$ with 188,160) over medium (U_2 having 3,024) to small ($U4$ with 180). Second, the opposition of domains (i.e. how compatible the interests of two parties are w.r.t. obtaining a joint benefit) has a broad spectrum, ranging from weak (U_1) over medium (U_3) to strong (U_4). Third, the domains show a broad spectrum of discounting factors, ranging from 0.4 (for D_1) to 1.0 (for all non-discounting domains). The distinct discounting factors (from the highest D_1 to non-discounting domains, i.e. U_n) of the ten domains require agent's appropriate decision-making under real-time pressure against unknown opponents. Together these aspects make the used domains very challenging. In view of these aspects the obtained results clearly indicate that **OMAC** is not limited to usage in a few specific domains, but is of value for a broad range of applications.

Agent	Non-discounting domain		Discounting domain	
	mean	variance	mean	variance
OMAC-agent	**0.831**	**0.0008**	**0.714**	**0.0016**
HardHeaded	0.741	0.0010	0.623	0.0013
Gahboninho	0.807	0.0023	0.555	0.0031
Agent_K2	0.557	0.0025	0.591	0.0023
BRAMAgent	0.502	0.0089	0.577	0.0084
Iamhaggler2011	0.474	0.0046	0.602	0.0021

Table 1. The average normalized score of agents for discounting and non-discounting domains.

Table 1 shows the average normalized scores for each agent in the discounting and non-discounting domains, and Table 2 shows for each agent the overall raw and normalized scores averaged over the ten domains. As usual, the scores were standardized by using the maximum and minimum utility obtained by all participants in the domain. According to Table 1, where the agents are ordered by the final ranking shown in Table 2, **OMAC-agent** completely outclassed others with a small variance. Another interesting observation is that Gahboninho performed very well in non-discounting domains (closely following **OMAC-agent**), while it achieved the worst score in discounting domains. A similar performance behavior was shown by Iamhaggler2011. This means that their behaviors are somewhat "extreme" and not adaptive enough to cope with the given range of negotiation domains. Compared to these agents, the champion of ANAC 2011 – Hardhead – achieved a more balanced and effective behavior.

Agent	Raw Score		Normalized Score	
	mean	variance	mean	variance
OMAC-agent	**0.767**	**0.0013**	**0.772**	**0.0101**
HardHeaded	0.713	0.0011	0.682	0.0035
Gahboninho	0.713	0.0027	0.681	0.0083
Agent_K2	0.641	0.0034	0.574	0.0059
BRAMAgent	0.613	0.0087	0.540	0.0208
IAMhaggler2011	0.612	0.0024	0.538	0.0088

Table 2. Overall performance.

From Table 2 it can be seen that the best overall performance is achieved by **OMAC-agent** – its utility (in terms of the normalized score) is 28.1% higher than the average utility of the other agents. The second best performer is Hardheaded, with a disadvantage of about 13% compared to **OMAC-agent**, and the lowest performance was shown by Iamhaggler2011.

6 CONCLUSIONS

This paper introduced an effective approach called **OMAC** ("Opponent Modeling and Adaptive Concession") for automated negotiation in complex – bilateral multi-issue, time-constrained, no prior knowledge, low computational load, etc. – scenarios. This approach, based on wavelet decomposition and cubic smoothing spline, outperformed the five best agents from the 2011 International Automated Negotiation Agents Competition (ANAC).

We think the experimental results justify to invest further research efforts into this approach and we see several promising research directions. Specifically, we find research which addresses the following three questions particularly relevant. First, are there opponent modeling techniques which are even more efficient than wavelet decomposition and cubic smoothing spline? Second, are there techniques for concession rate adaptation which are more accurate than the basic technique currently used? And third, can opponent modeling of **OMAC**, which currently focuses on modeling the opponent's strategies, be extended toward modeling the opponent's preferences as well? Our current research aims at exploring the first question.

ACKNOWLEDGEMENTS

We would like to thank the China Scholarship Council (CSC) for providing a PhD scholarship to Siqi Chen. Moreover, we appreciate the helpful discussions with the members of the DKE Swarmlab and the valuable comments of the reviewers.

REFERENCES

[1] Tim Baarslag, Koen Hindriks, Catholijn M. Jonker, Sarit Kraus, and Raz Lin, 'The first automated negotiating agents competition (ANAC 2010)', in *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence*. Springer-Verlag, (2010).

[2] Jakub Brzostowski and Ryszard Kowalczyk, 'Predicting partner's behaviour in agent negotiation', in *Proceedings of the Fifth Int. Joint Conf. on Autonomous Agents and Multiagent Systems*, AAMAS '06, pp. 355–361, New York, NY, USA, (2006). ACM.

[3] Réal Carbonneau, Gregory E. Kersten, and Rustam Vahidov, 'Predicting opponent's moves in electronic negotiations using neural networks', *Expert Syst. Appl.*, **34**, 1266–1273, (February 2008).

[4] Robert M. Coehoorn and Nicholas R. Jennings, 'Learning on opponent's preferences to make effective multi-issue negotiation trade-offs', in *Proceedings of the 6th Int. conf. on Electronic commerce*, ICEC '04, pp. 59–68, New York, NY, USA, (2004). ACM.

[5] I. Daubechies, *Ten lectures on wavelets*, Society for Industrial and Applied Mathematics, 2006.

[6] de Boor, *A Practical Guide to Splines*, Springer-Verlag, 1978.

[7] Peyman Faratin, Carles Sierra, and Nicholas R. Jennings, 'Negotiation decision functions for autonomous agents.', *Robotics and Autonomous Systems*, **24**(4), 159–182, (1998).

[8] Peyman Faratin, Carles Sierra, and Nicholas R. Jennings, 'Using similarity criteria to make issue trade-offs in automated negotiations', *Artif. Intell.*, **142**(2), 205–237, (2002).

[9] M. Hendrikxn, *A Survey of Oppnent Models in Automated Negotiation*, September 2011.

[10] K. Hindriks, C. Jonker, S. Kraus, R. Lin, and D. Tykhonov, 'Genius: negotiation environment for heterogeneous agents.', in *Proceedings of AAMAS 2009*, pp. 1397–1398, (2009).

[11] Chongming Hou, 'Predicting agents tactics in automated negotiation', in *Intelligent Agent Technology, IEEE / WIC / ACM International Conference on*, volume 0, pp. 127–133, Los Alamitos, CA, USA, (2004). IEEE Computer Society.

[12] Guoming Lai, Cuihong Li, Katia Sycara, and JosephAndrew Giampapa, 'Literature review on multi-attribute negotiations', Technical Report CMU-RI-TR-04-66, Robotics Institute, Pittsburgh, PA, (December 2004).

[13] Howard Raiffa, *The art and science of negotiation*, Harvard University Press Cambridge, Mass, 1982.

[14] Ariel Rubinstein, 'Perfect equilibrium in a bargaining model', *Econometrica*, **50**(1), 97–109, (1982).

[15] Sabyasachi Saha, Anish Biswas, and Sandip Sen, 'Modeling opponent decision in repeated one-shot negotiations', in *Proceedings of the Fourth international joint conference on Autonomous agents and multiagent systems*, AAMAS '05, pp. 397–403, New York, NY, USA, (2005). ACM.

[16] C.R. Williams, V. Robu, E.H. Gerding, and N.R. Jennings, 'Using gaussian processes to optimise concession in complex negotiations against unknown opponents', in *In Proceedings of the 22nd Internatioanl Joint Conference on Artificial Intelligence.* AAAI Press, (2011).

[17] Shahriar Yousefi, Ilona Weinreich, and Dominik Reinarz, 'Wavelet-based prediction of oil prices', *Chaos Solitons Fractals*, **25**(2), 265–275, (2005).