Expert Systems with Applications 42 (2015) 2287–2304

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

An approach to complex agent-based negotiations via effectively modeling unknown opponents ☆

Siqi Chen *, Gerhard Weiss

Department of Knowledge Engineering, Maastricht University, The Netherlands

ARTICLE INFO

Article history:
Available online 7 November 2014

Keywords:
Multi-agent systems
Automated multi-issue negotiation
Opponent modeling
Multi-resolution wavelet analysis
Gaussian processes
Empirical game theory

ABSTRACT

Negotiation among computational autonomous agents has gained rapidly growing interest in previous years, mainly due to its broad application potential in many areas such as e-commerce and e-business. This work deals with automated bilateral multi-issue negotiation in complex environments. Although tremendous progress has been made, available algorithms and techniques typically are limited in their applicability for more complex situations, in that most of them are based on simplifying assumptions about the negotiation complexity such as simple or partially known opponent behaviors and availability of negotiation history. We propose a negotiation approach called OMAC★ that aims at tackling these problems. OMAC★ enables an agent to efficiently model opponents in real-time through discrete wavelet transformation and non-linear regression with Gaussian processes. Based on the approximated model the decision-making component of OMAC★ adaptively adjusts its utility expectations and negotiation moves. Extensive experimental results are provided that demonstrate the negotiation qualities of OMAC★, both from the standard mean-score performance perspective and the perspective of empirical game theory. The results show that OMAC★ outperforms the top agents from the 2012, 2011 and 2010 International Automated Negotiating Agents Competition (ANAC) in a broad range of negotiation scenarios.

1. Introduction

Agent-based negotiation is about computational autonomous agents that attempt to arrive at joint agreements in competitive consumer-provider or buyer–seller scenarios on behalf of humans (Jennings et al., 2001). As one of the most fundamental and powerful mechanisms for solving conflicts between parties of different interests, recent years have witnessed a rapidly growing interest in automated negotiation, mainly due to its broad application range in fields as diverse as electronic commerce and electronic markets, supply chain management, task and service allocation, and combinatorial optimization. As a result, agent-based negotiation brings together research topics of artificial intelligence, machine learning, game theory, economics, and social psychology (Chen, Hao, Weiss, Tuyls, & Leung, 2014).

Dependent on the assumptions made about the negotiating agents' knowledge and the constraints under which the agents negotiate, negotiation scenarios show different levels of complexity. The following assumptions, which are reasonable in view of real-world applications and which underly our work, induce high complexity and raise particular demands on the abilities of the negotiators. First, the agents have no usable prior information about their opponents – neither about their preferences (e.g., their preferences over issues or their issue value ordering) nor about their negotiation strategies. Then, the negotiation is constrained by the amount of time being elapsed, the participants therefore do not know at any time during negotiation how many negotiation rounds there are left and they have to take into account at each time point (i) the remaining chances for offer exchange and (ii) the fact that the profit achievable through an agreement decreases over time ("negotiation with deadline and discount"). Third, each agent has a private reservation value below which an offered contract is not accepted.[1] Thereby we adopt the common view that an

☆ This article is a substantially extended version of our ECAI main track paper (Chen & Weiss, 2012). The extension primarily concerns the negotiation approach (OMAC) described in the ECAI paper. An effective negotiation approach called OMAC★ is proposed that advances OMAC in three significant ways. Furthermore, a comprehensive experimental analysis, as well as a useful game-theoretical robustness analysis, is presented. In the experimental analysis a large number of negotiation scenarios are considered and a comparison with the best agents from recent editions of ANAC competitions as well as with the predecessor of OMAC★ (i.e., OMAC) is provided.

* Corresponding author at: Department of Knowledge Engineering, Maastricht University, P.O. Box 616, 6200 MD Maastricht, The Netherlands. Tel.: +31 433883909.

E-mail addresses: siqi.chen@maastrichtuniversity.nl (S. Chen), gerhard.weiss@maastrichtuniversity.nl (G. Weiss).

[1] This reservation value is also called utility of conflict or disagreement solution.

http://dx.doi.org/10.1016/j.eswa.2014.10.048
0957-4174/

agent obtains the reservation value even if no agreement is reached in the end. This implies that breaking-off a negotiation session would be potentially beneficial especially when the time-discounting effect is substantial and the other side is being very tough. Together these assumptions make negotiations complicated (yet realistic), where efficiently reaching agreements are particularly challenging. We thus refer to such type of negotiations as *complex negotiations* afterwards.

Although there exist many research efforts to address the problems of complex negotiations over the past years, two issues still stand out. The first one relates to learning unknown opponents' strategies. While it has been realized by early work that a successful negotiation needs to be based in one way or another on learning opponent models, the existing learning approaches either are limited in their usage in complex negotiations due to the impractical assumptions made about the environment, or have low efficacy in modeling opponents. The other issue is the absence of a decision-making mechanism that is suited for complex negotiations (i.e., the way of how to concede towards opponents in the course of negotiation). The strategies available to complex negotiation tend to consider concession in an intuitive fashion, or neglect the problem of "irrational concession" (see Section 5.2). As a result, the current decision-making methods are not adaptive and effective to respond to the high uncertainty of complex negotiations.

Based on the above motivation, this work proposes a novel strategy called OMAC★ for complex negotiations to address the aforementioned two issues that could further improve performance of a negotiating agent. In particular, it extends the OMAC negotiation strategy, which we introduced in Chen and Weiss (2012), in several important aspects (as detailed in Section 2). The proposed approach manages to integrate two key aspects of a successful negotiation: efficient opponent modeling and adaptive decision-making. Opponent modeling realized by OMAC★ aims at predicting the utilities of opponent future counter-offers (for itself) and is achieved through two standard mathematical techniques known as discrete wavelet transformation (DWT) and Gaussian processes (GPs). Adaptive decision-making realized by OMAC★ consists of two components, namely, concession making and counter offer responding, and it employs the learnt opponent model to automatically adjust the concession behavior and the response to counter-offers from opponents.

The remainder of this article is structured as follows. Section 2 overviews important related work. Section 3 provides the negotiation environment that we have considered. Section 4 describes the main mathematical techniques exploited by OMAC★. Section 5 shows the technicalities of the proposed strategy. Sections 6 and 7 offer a careful empirical evaluation and game-theoretic analysis of OMAC★. Section 8 discusses some interesting experimental results and other related aspects of agent-based negotiation. Finally, Section 9 identifies some important research lines induced by the work.

2. Related work

Negotiation has traditionally been investigated in game theory (Osborne & Rubinstein, 1994; Raiffa, 1982) and in previous years it has also developed into a core topic of multiagent systems (e.g., Lopes, Wooldridge, & Novais, 2008; Mor, Goldman, & Rosenschein, 1996; Weiss, 2013). Numerous approaches to automated negotiation have been proposed that, like the one described in this work, explore the idea to equip an agent with the ability to build a model of its opponent and to use this model for optimizing its negotiation behavior. Modeling the opponent's behavior, however, is practically challenging because negotiators usually do not reveal their true preferences and/or negotiation strategies in order to avoid that others exploit this information to their advantage (e.g., Coehoorn & Jennings, 2004; Raiffa, 1982). Current methods however tend to make simplifying assumptions about the negotiation settings. For example, there are approaches that deal with single-issue negotiation and others that assume that the opponents have a rather simple (e.g., non-adaptive) behavior, or the negotiations take place in scenarios with a low dimension (e.g., a small number of issues and possible choices for each of them). In the following, representative model-based negotiation approaches are overviewed.

Many of the available approaches aim at learning opponents' preferences or the reservation value. Faratin, Sierra, and Jennings (2002) propose a trade-off strategy to increase the chance of getting own proposals accepted without decreasing the own profit. The strategy applies the concept of fuzzy similarity to approximate the preference structure of the opponent and uses a hill-climbing technique to explore the space of possible trade-offs for its own offers that are most likely to be accepted. The effectiveness of this method highly depends on the availability of prior domain knowledge that allows to determine the similarity of issue values. Coehoorn and Jennings (2004) propose a method using Kernel Density Estimation for estimating the issue preferences of an opponent in multi-issue negotiations. It is assumed that the negotiation history is available and that the opponent employs a time-dependent tactic (i.e., the opponent's concession rate depends on the remaining negotiation time, see, e.g., Faratin, Sierra, & Jennings (1998) for details on this kind of tactic). The distance between successive counter-offers is used to calculate the opponent's issue weights and to assist an agent in making trade-offs in negotiation. Some approaches use Bayesian learning in automated negotiation. For instance, Zeng and Sycara (1998) use a Bayesian learning representation and updating mechanism to model beliefs about the negotiation environment and the participating agents under a probabilistic framework; more precisely, they aim at enabling an agent to learn the reservation value of its opponent in single-issue negotiation. Another approach based on Bayesian learning is presented in Lin, Kraus, Wilkenfeld, and Barry (2008). Here the usage of a reasoning model based on a decision-making and belief-update mechanism is proposed to learn the likelihood of an opponent's profile; thereby it is assumed that the set of possible opponent profiles is known as a priori. Hindriks and Tykhonov (2008) present a framework for learning an opponent's preferences by making assumptions about the preference structure and rationality of its bidding process. It is assumed that (i) the opponent starts with optimal bids and then moves towards the bids close to the reservation value, (ii) its target utility can be expressed by a simple linear decreasing function, and (iii) the issue preferences (i.e., issue weights) are obtainable on the basis of the learned weight ranking. Moreover, the basic shape of the issue evaluation functions is restricted to downhill, uphill or triangular. In order to further reduce uncertainty in high-dimensional domains, issue independence is assumed to scale down the otherwise exponentially growing computational complexity. Oshrat, Lin, and Kraus (2009) developed an effective negotiating agent for effective multi-issue multi-attribute negotiations with both human counterparts and automated agents. The successful negotiation behavior of this agent is, to a large extent, grounded in its general opponent modeling component. This component applies a technique known as Kernel Density Estimation to a collected database of past negotiation sessions for the purpose of estimating the probability of an offer to be accepted, the probability of the other party to propose a bid, and the expected averaged utility for the other party. The estimation of these values plays a central role in the agent's decision making. While the agent performs well, the approach taken is not suited for the type of negotiation we are considering (real-time, no prior knowledge, etc.) because opponent modeling is done offline and requires knowledge about previous negotiation traces.

Other available approaches aim at learning the negotiation strategy and decision model of the opposing negotiator. For instance, Saha, Biswas, and Sen (2005) apply Chebychev's polynomials to estimate the chance that an opponent accepts an offer relying on the decision history of its opponent. This work deals with single-issue negotiation, where an opponent's response can only be an accept or a reject. Brzostowski and Kowalczyk (2006) investigate online prediction of future counter-offers on the basis of the past negotiation exchanges by using differentials. They assume that there mainly exist two independent factors that influence the behavior of an opposing agent, namely, time and imitation. The opponent is assumed to apply a weight combination of time- and behavior-dependent tactic.[2] Hou (2004) presents a learning mechanism that employs non-linear regression to predict the opponent's decision function in a single-issue negotiation setting. Thereby it is assumed that the opponent behavior can only be time-, behavior- or resources-dependent (with decision functions as proposed in Faratin et al. (1998)). In Carbonneau, Kersten, and Vahidov (2008), an artificial neural network (ANN) is constructed with three layers that contain 52 neurons to model a negotiation process in a specific domain. The network exploits information about past counter-offers to simulate future counter-offers of opponents. The training process requires a very large database of previous offer exchanges and huge computational resources, and therefore cannot be applied to complex negotiations considered in this work.

Recently, there is a growing body of work dealing with complex negotiations by means of learning opponent strategy. Some good examples are Williams, Robu, Gerding, and Jennings (2011), Chen and Weiss (2013), Chen, Ammar, Tuyls, and Weiss (2013), Chen and Weiss (2014) and Hao, Song, Leung, and Ming (2014). In the work of Williams et al. (2011), the authors employ Gaussian processes to learn opponent models. The learnt opponent model can provide a negotiating agent with the estimated maximal opponent concession so that the agent could optimize its own expected utility. However, suffering from the problem of "irrational concession" (as we will discuss later in Section 5.2), it tends to misinterpret the intention of opponents, especially in the case of competing against tough opponents. As a result, this approach fails to perform efficient negotiations with other state-of-the-art negotiation strategies. Chen and Weiss (2014) introduce a negotiation strategy based on a variant of Gaussian processes regression model. Instead of finding solutions to further improving learning performance, the main focus of the strategy is on alleviating the computational complexity of learning opponent models. In the work of Chen and Weiss (2013) the authors explore a combination of Empirical Mode Decomposition (EMD) and Autoregressive Moving Average (ARMA) to cope with complex negotiation scenarios. Their approach generates a decomposition of the time series based on the received utilities of past counter-offers into a finite number of simpler components, which allow for an easier subsequent utility prediction for each component. A major drawback of the approach is that it has comparably high prediction errors (see the model comparison shown in Table 4). This is because the approach has to perform N prediction tasks simultaneously (where N equals the number of simpler components) and thus the complexity of regression techniques under consideration must be kept low, thereby making more powerful regression methods (which also require higher computation load) inapplicable. Another weakness of this approach is that the offer-generating component proposes new offers in a simple random way, thus limiting negotiation efficacy. Hao et al. (2014) propose another successful approach for complex negotiations. The approach attempts to concede toward opponents as less as possible through adjusting the so-called non-exploitation time point. Furthermore, in order to improve the likelihood of its own proposals being accepted, the authors also employ a reinforcement-learning based approach to predict the optimal offers for the other negotiation party. In addition, a novel knowledge transfer method of learning opponent models for negotiating agents based on deep learning machines is developed in Chen et al. (2013). This method, while useful, varies in that its successful operation needs the knowledge of previous negotiation tasks against the opponents, which are not available in complex negotiations (please note the definitions given in Section 1).

Therefore, learning opponent models in existing literature is either inefficient or limited in usage due to the impractical assumptions. Moreover, an adaptive concession-making mechanism is also lacking in this filed. Against this background, this work describes OMAC* (as an improved version of OMAC) that advances the state-of-the-art of complex bilateral multi-issue negotiations in three significant ways. First, it adopts a new learning scheme for opponent modeling that can effectively predict opponent behavior in real time through Gaussian processes and discrete wavelet transformation. Second, an improved concession-making mechanism is provided that takes into account the agent's estimated real reservation utility and a high-confident estimate of the forthcoming opponent concession for making adaptive concession in response to the high uncertainty of complex negotiations. And third, it has an enhanced response mechanism that supports an agent in selecting offers with high acceptance probability for its opponents and in determining when to withdraw from a negotiation session. Together these new features result in a considerably more effective and adaptive negotiation strategy, as shown by the experimental results that also include a direct comparison of OMAC* and OMAC.

3. Negotiation environment

The work described here adopts a bilateral negotiation environment that is widely used in the agents field (e.g., Baarslag et al., 2013; Faratin et al., 2002; Hao et al., 2014; Williams et al., 2011). The negotiation protocol is based on the standard alternating offers formalized in Rubinstein (1982) but in a real-time manner. 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 ($j \in \{1, \ldots, n\}$ 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 minimum payoff as the outcome of a negotiation; this is called the reservation value ϑ. Further, w_j^i ($j \in \{1, \ldots, n\}$) denotes the weighting preference which agent i assigns to issue j. The issue weights of an agent i are normalized summing to one (i.e., $\sum_{j=1}^{n}(w_j^i) = 1$). During a negotiation session agent a and b act in conflictive roles that are specified by their preference profiles. In order to reach an agreement they exchange offers (i.e., O) in each round to express their demands. An offer is thereby 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:

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

where v_{jk} is the kth value of the issue j and V_j^i is the evaluation function for agent i, mapping a value of issue j (e.g., v_{jk}) to a real number.

Negotiation considered here is time-limited rather than restricted by a fixed number of rounds. Specifically, negotiators have a shared deadline by when they must have completed the negotiation; this deadline is denoted by t_{max}. If no agreement is reached at the end or one side breaks off before deadline, the

[2] The concepts of time-dependent and behavior-dependent tactics were introduced in Faratin et al. (1998).

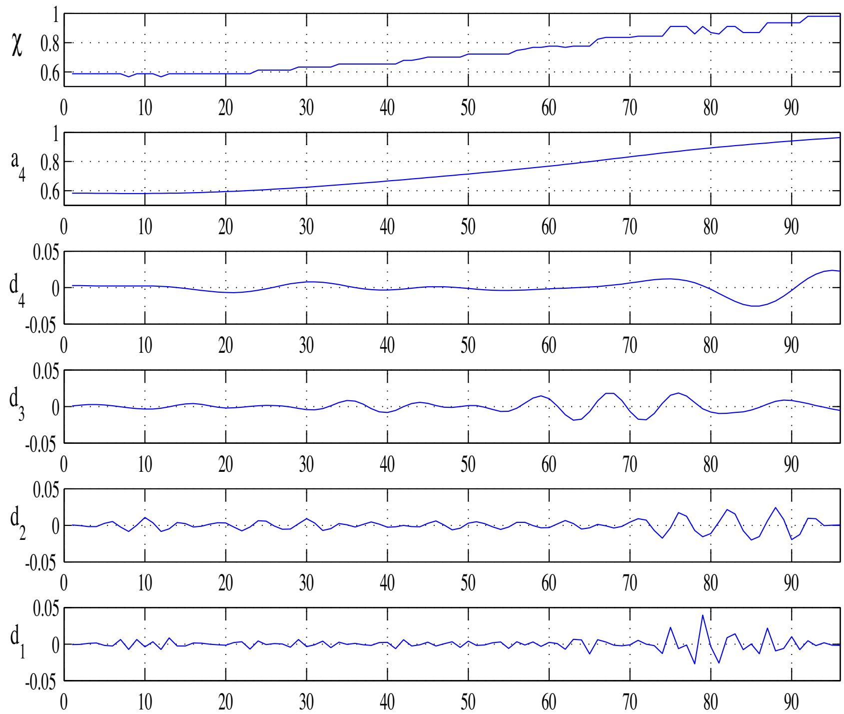

Fig. 1. 4-Level decomposition of the utilities series χ obtained from IAMhaggler2011 in the negotiation domain *Airport site selection* using DB10. The vertical axis shows score and the horizontal axis represents the percentage of time (%). The 4-level DWT decomposes the original input χ into five sub-components including $d_1 \ldots d_4$ (detail parts) and a_4 (approximation part). The resulting components are shown below χ, their frequency increases from d_1 to d_4 (i.e., they are more and more smooth). The final approximation part – a_4 is the long-term trend of χ.

negotiation then ends up with the disagreement solution. Note that 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, in which the utility is discounted with time. We define a so-called discounting factor δ ($\delta \in [0,1]$) and use this factor to calculate the discounted utility as follows:

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

where U is the (original) utility and t is the standardized time (i.e., $t \in [0,1]$). As an effect, the longer it takes for agents to come to an agreement the lower is the utility they can achieve. Note that a decrease in δ increases the discounting effect.

Upon receiving a counter-offer from the opponent, O_{opp}, an agent decides on acceptance, rejection and withdrawal according to the interpretation of its reasoning model.[3] For instance, the acceptance decision can be made in dependence on a certain threshold $Thres^i$: agent i accepts if $U^i(O_{opp}) \geqslant Thres^i$, and rejects otherwise. As another example, the decision could be based on differences in successive utilities.

4. Techniques for opponent modeling

This section briefly introduces the two main techniques used by OMAC* for modeling opponents – discrete wavelet transformation (Section 4.1) and Gaussian processes (Section 4.2). OMAC* uses discrete wavelet transformation to extract the previous main trend of an ongoing negotiation. After that it employs Gaussian processes to predict the future main trend.

4.1. Discrete wavelet transformation

Discrete wavelet transformation (DWT) is a type of multi-resolution wavelet analysis that provides a time–frequency representation of a signal and, based on this, it is capable of capturing time localizations of frequency components. DWT has become increasingly important and popular as an efficient multi-scaling tool for exploring features. This is due to the fact that it can offer with modest computational effort high-quality solutions (with complexity O(n)) to non-trivial problems such as feature extraction, noise reduction, function approximation and signal compression. OMAC* employs DWT to extract the main trend of the opponent's concession over time from its previous counter-offers. In the following, aspects of DWT are described that are relevant to OMAC*; further details can be found in, e.g., Daubechies (2006) and Ruch and Fleet (2009).

In DWT a time–frequency representation of a signal is obtained through digital filtering techniques, where two sets of functions are utilized: scaling functions using low-pass filters and wavelet functions using high-pass filters. More precisely, a signal is passed through a series of high pass filters to analyze the high frequencies, and similarly it is passed through a series of low pass filters to analyze the low frequencies. In so doing, DWT decomposes a signal into two parts, an approximation part and a detail part. The former is smooth and reveals the basic trend of the original signal, and the latter is rough and in general corresponds to short-term noise from the higher-frequency band.

The decomposition process can be applied recursively as follows:

$$\begin{aligned} y_{low}[k] &= \sum_n f[n] \cdot h[2k-n] \\ y_{high}[k] &= \sum_n f[n] \cdot g[2k-n] \end{aligned} \tag{3}$$

[3] If the agents know each other's utility functions, they can compute the Pareto-optimal contract (Raiffa, 1982). However, a negotiator will not make this information available to its opponent in competitive settings.

with $f[n]$ being the signal, $h[n]$ a halfband high-pass filter, $g[n]$ a halfband low-pass filter, and $y_{low}[k]$ and $y_{high}[k]$ the outputs of the low-pass and high-pass filters, respectively. The iterative application of DWT results in different levels of detail of the input signal; in other words, it decomposes the approximation part into a "further smoothed" component and a corresponding detail component. The further smoothed component contains longer period information and provides a more accurate trend of the signal. For instance, f can firstly be decomposed into a rough smooth part (a_1) and a detail part (d_1), and then the resulting part a_1 can be decomposed in finer components, that is, $a_1 = a_2 + d_2$, and so on. This iterative process is captured by the below diagram:

$$\begin{array}{lllll} f \cdots a_1 \cdots a_2 \cdots a_3 \cdots a_n \\ \ddots \quad \ddots \quad \ddots \\ \quad d_1 \quad d_2 \quad d_3 \cdots d_n \end{array}$$

where $a_1, a_2, \ldots, a_n$ are the approximation parts and $d_1, d_2, \ldots, d_n$ are the detail parts of f.

Daubechies wavelets are a family of orthogonal wavelets defining a DWT. They are applied for solving a range of problems, e.g., self-similarity properties of a signal, fractal problems, signal discontinuities, etc. The results reported in this article are achieved through wavelet decomposition using the Daubechies' wavelets of order 10 (referred to as "DB10" afterwards). We use the notation below to represent the decomposition relation in our case:

$$\chi = \omega + \sum_{n=1}^{\lambda} d_n \tag{4}$$

where χ is the time series (i.e., the maximum utilities of counter-offers over intervals, refer to Section 5.1), ω represents the n-layer approximation component of χ, d_n is the n-layer detail part, and λ the number of decomposition level.

A concrete example of applying DWT in negotiation is given in Fig. 1, which shows the curve of the received utilities (i.e., the original signal) in the domain *Airport site selection* when negotiating with the agent IAMhaggler2011 (more information about domains and agents can found in Section 6.1). The decomposition results were obtained with $\lambda = 4$. The curve at the top of figure represents χ, d_n is the detail component of the n^{th} decomposition layer and a_4 is the approximation on the final layer (i.e., the fourth), corresponding to ω in Eq. (4). This figure clearly shows that a_4 is a pretty good approximation of the main trend of the original signal. As can be seen, the noise/variation represented by those detail components (e.g., d_1 to d_4) is irrelevant to its trend.

4.2. Gaussian processes

OMAC★ adopts Gaussian processes (GPs) to learn an opponent model that does not only allow to make confident predictions but also provides a measure of the level of confidence in the predictions. GPs are an important tool in statistical modeling and are widely used to perform Bayesian nonlinear regression and classification. In the following, main aspects of GPs relevant to OMAC★ are overviewed; for a detailed discussion we refer to Rasmussen and Williams (2006).

Formally, GPs are a form of nonparametric regression that perform inference directly in the functional space. Specifically, GPs define probability distributions over functions. Let $\mathcal{D} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{n}$ be a data set where $\mathbf{x} \in \mathbb{R}^d$ is the input vector, $y \in \mathbb{R}$ the output vector and n is the number of available data points. When a function is sampled from a GP, we write:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

where $m(\mathbf{x})$ is the mean function and $k(\mathbf{x}, \mathbf{x}')$ the covariance function. $m(\mathbf{x})$ and $k(\mathbf{x}, \mathbf{x}')$ can fully specify a GP. A common assumption is that GPs have mean zero, which greatly simplifies calculations without loss of generality. We also follow this view in the work.

Rasmussen and Williams (2006) present a wide variety of covariance functions. In this work the *Matérn* covariance function is selected because it is robust and can be computed in real time settings:

$$k_y(\mathbf{x}, \mathbf{x}') = a^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell} \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu} r}{\ell} \right) \tag{5}$$

with r denoting the Euclidean distance between $\mathbf{x}$ and $\mathbf{x}'$. The positive parameters a and ℓ determine the amplitude and length-scale, respectively, the positive parameter ν controls the smoothness of the sample functions, $K_\nu(\cdot)$ is a modified Bessel function Abramowitz and Stegun (1965), and $\Gamma(\cdot)$ is the Gamma function with the form $\Gamma(z) = \int_0^\infty \frac{t^{z-1}}{e^t} dt$.

As the data in GP modeling can be represented as a sample from a multivariate Gaussian distribution, we have the following joint Gaussian distribution:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_T \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathbf{K}_N & \mathbf{K}_{NT} \\ \mathbf{K}_{TN} & \mathbf{K}_T \end{bmatrix} + \sigma_n^2 \mathbf{I}\right) \tag{6}$$

where $\mathbf{K}$ is the covariance matrix, $\mathbf{K}_{ij} = k_y(\mathbf{x}_i, \mathbf{x}_j)$, σ_n the noise variance, $\mathbf{I}$ is the identity matrix, N is the size of the training set, and T is the size of test inputs. The resulting predictive distribution is then obtained by conditioning on the observed target outputs (i.e., response variables) and is given by:

$$p(\mathbf{y}_T | \mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_T), \tag{7}$$

where $\boldsymbol{\mu}_T$ and $\boldsymbol{\Sigma}_T$ are defined as

$$\boldsymbol{\mu}_T = \mathbf{K}_{TN}[\mathbf{K}_N + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{y} \tag{8}$$

$$\boldsymbol{\Sigma}_T = \mathbf{K}_T - \mathbf{K}_{TN}[\mathbf{K}_N + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{K}_{NT} + \sigma_n^2 \mathbf{I} \tag{9}$$

Finally, learning in a GP setting involves maximizing the marginal likelihood given by

$$\begin{aligned} L &= \log p(\mathbf{y}|\mathbf{X}, \theta) \\ &= -\frac{1}{2} \mathbf{y}^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y} - \frac{1}{2} \log |\mathbf{K} + \sigma_n^2 \mathbf{I}| - \frac{n}{2} \log 2\pi \end{aligned} \tag{10}$$

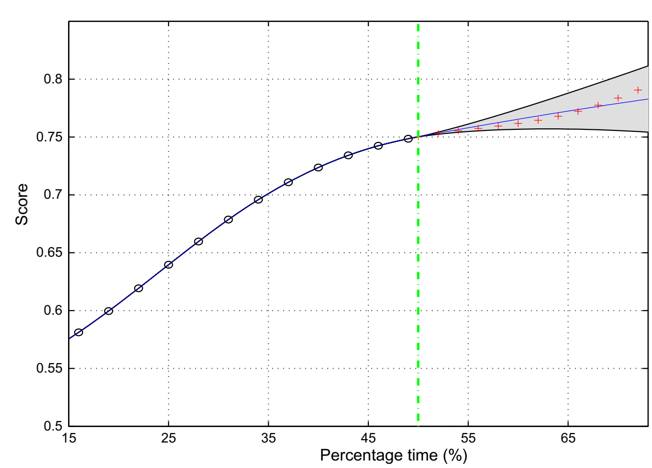

Fig. 2. Illustration of the prediction ability of OMAC★. The dash-dot line indicates the time point t_c at which the current prediction is made. The circles to the left of the dash-dot line are the historical points of ω before t_c. The plus signs to the right of the dash-dot line show the actual points of ω after t_c. The prediction of ω is described by the solid curve and the confidence is shown by the shaded area. The results shown in this figure are achieved in the negotiation domain *Flight booking* when playing against the agent Agent_K (more detail on domains and agents is given in Section 6.1).

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 input data set, and $\mathbf{K} \in \mathbb{R}^{m\times m}$ is the covariance matrix with $|\cdot|$ representing the determinant. We briefly mention here that a desirable property of GPs is that they automatically avoid overfitting.

Fig. 2 illustrates the prediction ability of GPs in the context of automated negotiation. It shows the actual and predicted approximation parts (i.e., ω) of concession curve of the opponent "Agent_K" at different time points. As can be seen, the prediction is accurate in the light of the actual data points, while the errors naturally grow in regions outside the training data where there is high uncertainty about the approximated function.

5. The OMAC* strategy

OMAC* is composed of two functional core components. First, an opponent-modeling component (described in Section 5.1), which learns a model of the opponent through a combination of discrete wavelet decomposition and non-parametric regression. Second, a decision-making component, which is responsible for adaptively making concessions (Section 5.2) and for appropriately responding to a counter-offer (Section 5.3) on the basis of the learnt opponent model. Algorithm 1 shows OMAC* at a glance, the individual steps are explained below.

Algorithm 1. The OMAC* strategy. Let t_c be the current time, δ the time discounting factor, and t_{max} the deadline of negotiation. O_{opp} is the latest offer of the opponent and O_{own} is a new offer to be proposed by OMAC*. χ is the time series comprised of the maximum utilities over intervals. ξ is the lead time for prediction procedure and ω is the central tendency of χ obtained from DWT. $E_\delta(t)$ is the expected discounted received utility at time t. u_{res} is the estimated effective reservation utility, and e^p is the confident estimate of the maximum opponent concession with a probability of p. R is the conservative aspiration level function, u' the target utility at time t_c. As explained above, δ is the discounting factor and ϑ the default reservation value specified by the preference profile

1: *Require:* $t_{max}, \delta, \vartheta, \xi, p, R$
2: **while** $t_c <= t_{max}$ **do**
3: $\quad O_{opp} \Leftarrow receiveMessage()$;
4: $\quad Bids = recordBids(t_c, O_{opp})$;
5: $\quad$ **if** $NewInterval(t_c)$ **then**
6: $\qquad \chi \Leftarrow preprocessData(t_c, Bids)$
7: $\qquad \omega \Leftarrow decompose(\chi)$;
8: $\qquad (E_\delta(t), t_l) \Leftarrow predict(\omega, \chi, \xi)$;
9: $\qquad (u_{res}, e^p_{min}) \Leftarrow updateParas(\omega, \chi, \vartheta, p, t_l)$;
10: $\qquad R \Leftarrow (u_{res}, e^p)$;
11: $\quad$ **end if**
12: $\quad u' = getTargetU(E_\delta(t), R, \delta, t_c)$;
13: $\quad$ **if** $isAcceptable(u', O_{opp}, \delta, t_c)$ **then**
14: $\qquad agree(O_{opp})$;
15: $\quad$ **else**
16: $\qquad checkTermination()$;
17: $\qquad O_{own} \Leftarrow constructOffer(u')$;
18: $\qquad proposeNewBid(O_{own})$;
19: $\quad$ **end if**
20: **end while**

5.1. Opponent modeling

According to OMAC*, the objective of opponent modeling is twofold: to analyze the opponent's past bidding strategy with the goal to reveal the concession trend implied by its behavior ("trend analysis"); and to predict the utilities of the opponent's forthcoming offers (for the agent) on the basis of the identified trend ("trend prediction or extrapolation"). The process of opponent modeling is captured by lines 3 to 11 in Algorithm 1. Opponent modeling is technically done through a combination of discrete wavelet transformation (trend analysis) and regression with Gaussian processes (trend prediction).

When receiving a new counter-offer O_{opp} from an opponent at the time t_c, the agent records this time stamp and the utility $U(O_{opp})$ according to the agent's own utility function (see line 4). The agent divides a negotiation session into a fixed number of intervals (denoted as ζ) of equal duration. The sequence of the highest utility at each previous interval, together with their time stamps, is taken as the basis for predicting the opponent's behavior (line 6). The motivation for using this prediction basis is twofold (a similar motivation is given in Williams et al. (2011)). First, 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 several thousands of data points in every single negotiation session. This computational load would lead to a clear negative impact on the quality of negotiation in a real-time setting. Second, this reduces the risk of misinterpreting the opponent's behavior that exists in multi-issue negotiations because a small change in the utility of an opponent may result in a large utility change for the negotiator. The resulting time series consisting of the maximum utilities at each interval is referred to as χ afterwards.

To analyze the trend, the time series χ is first processed by applying discrete wavelet transformation (DWT); this is captured by line 7. The output of DWT includes an approximation and a detail component as described in Section 4.1. 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 curve and indicates how the concession of opponent will develop in the future. More importantly, it becomes more smooth (compared to the original signal, i.e. χ) to allow for accurate and robust prediction. Third, the detail parts correspond to high frequency short-term signal or random noise. Thus, these detail parts are trivial components of the original signal, and to calculate their precise predictive distribution would require a tremendous computational effort.

Regression is then performed with Gaussian processes to forecast the opponent's future moves using the results of trend analysis. A notable advantage of Gaussian processes is that it not only provides the accurate estimation of the dependent variable(s) but also gives a measure of the level of confidence in that prediction. Since OMAC* adopts a periodical updating mechanism, it is not necessary and also not advantageous to forecast globally (i.e., from the current moment to the end of negotiation), because this is likely to generate noise that results into imprecise predictions. OMAC* limits the range of forecasting to ξ intervals to achieve efficiency and noise reduction.

5.2. Adaptive concession-making mechanism

Based on the predictive distribution available through the learnt opponent model, OMAC* decides how to set the own target utility (see line 12 in Algorithm 1). One possibility is to set the target utility according to the maximum predicted utility. This is straightforward but may be ineffective. Suppose the negotiation opponents are "sophisticated and tough" and always avoid making concessions, the prediction results can then easily lead to a misleading, too low expectation about the future utility offered by the other party. This, in turn, can result in an adverse concession behavior.[4] Moreover, a global prediction approach can make this situation even worse. To deal with this "irrational concession" effect, OMAC★ employs a conservative aspiration level function $R(t)$ that carefully suggests target utilities for the agent. (This negative effect is also considered in Section 6.2.) The R-function is based on two variables, u_{res} and e^p, where the former represents our agent's estimated effective reservation utility and the latter represents the expectation of the maximum opponent concession. These two variables are periodically updated in dependence on the output of learnt opponent model (line 9 and 10 of Algorithm 1). Next, we motivate the usage of these two variables and define the R-function in detail.

Although the default reservation value of a negotiation (i.e., ϑ) is known, it is more like a "default solution" in the failure case (i.e., when no agreement is reached) rather than an indication of the actual minimum compromise the other party will make. Consider, for instance, an opponent that is cooperative in the sense that it is willing to concede more than ϑ (perhaps even in an early negotiation stage); in this case the worst possible outcome for the agent is not longer given by ϑ. The estimated effective reservation utility u_{res} is defined as follows:

$$u_{res} = \max\left(\omega^{low}_{max}(t_l), \vartheta\right) \tag{11}$$

where ϑ is the reservation value predefined by its own preference profile, t_l the last time point when the opponent modeling task is performed, max returns the larger value between arguments. $\omega^{low}_{max}(t)$ is the maximum value of $\omega^{low}(t)$ in $[0, t]$, which is the lower bound of ω and is defined as

$$\omega^{low}(t) = \omega(t) \cdot (\text{mean}(r_{[0,t]}) - \text{stdev}(r_{[0,t]})) \tag{12}$$

with ω being the main tendency of χ, $r_{[0,t]}$ the series representing the ratio between ω over χ in the interval $[0, t]$ and $stdev$ the standard deviation.

OMAC★ is sensitive to u_{res}, that means, an inappropriate setting of u_{res} would result in a negotiation failure (in the case it is too big and the agent thus tends to make no concession) or in a reduction of its potential payoff (in the case it is too small and the agent thus tends to concede more than necessary). Because ω depends on the received counter-offers, using the maximum value from its lower bound assures with low failure risk an increase of the agent's potential profit. When this value is smaller than ϑ, the agent uses ϑ instead (see Eq. (11)).

Another key factor of R(t) is e^p, which aims at keeping track of the best forthcoming compromise. Toward this end, a probability parameter p is used that specifies the likelihood of the prediction (i.e., the higher p, the more confident the prediction). The definition of e^p is based on the error function that is used in the standard cumulative distribution function (CDF) of a Gaussian distribution. More precisely, the CDF is given by

$$F(x; \mu, \sigma^2) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x - \mu}{\sqrt{2}\sigma}\right)\right] \tag{13}$$

where erf is the error function given by

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{14}$$

The complement of the above cumulative distribution function represents the upper tail probability of the Gaussian distribution, and its inverse function specifies an expected value (x) of a random variable X such that X falls into the interval $(x, +\infty)$ with the given probability p. This is expressed by

$$Q(p; \mu, \sigma) = \mu - \sqrt{2}\sigma^2 \text{erf}^{-1}(2p - 1) \tag{15}$$

In order to capture a high-confident estimate of the forthcoming maximum concession, the Q-function then takes as input a probability p, the maximum posterior mean estimate $\tilde{\mu}$, and the corresponding posterior standard deviation $\tilde{\sigma}$ in the resulting predictive distribution about χ. The probability p should be set high enough so that a strong confidence about the maximal opponent concession is ensured. Since u_{res} is the worst possible outcome, the agent takes it as the minimum value for e^p. Overall, this is captured by

$$e^p = \max(u_{\text{res}}, Q(p; \tilde{\mu}, \tilde{\sigma})) \tag{16}$$

The conservative aspiration level function $R(t)$ should decrease the utility expectation of an agent as time proceeds. Moreover, it should take into account the opponent behavior and the discounting effect. More precisely, $R(t)$ should be proportional to e^p and u_{res} so that the agent is likely to benefit from the concessive behavior of an opponent. At the same time, $R(t)$ should be inversely proportional to δ because the benefit of an early agreement becomes increasingly significant as the discounting factor decreases. These requirements on $R(t)$ can be instantiated in different ways. OMAC★ does this as follows:

$$R(t) = (U_{\max} - u_{\text{res}})(1 - t)^{\frac{\beta}{\delta}}\left(\frac{e^p}{U_{\max}}\right)^{\beta} + u_{\text{res}} \tag{17}$$

where β is the concession coefficient controlling the concession rate, $U_{\max}$ is the possible maximum utility in the scenario and δ is the discounting factor.

In addition, OMAC★ uses the expected discounted received utility $E_\delta(t)$ in its decision making. This utility, which corresponds to the expectation of how much discounted profit can be received from an opponent at some future time $t_\star$, is defined by:

$$E_\delta(t_\star) = \frac{1}{C}\int_{-\infty}^{+\infty} D_\delta(u \cdot f(u; \mu_\star, \sigma_\star), t_\star) du \tag{18}$$

where C is a constant called normalizing constant, f is the probability density function of Gaussian distribution, and $\mu_\star$ and $\sigma_\star$ are the mean and standard deviation (both obtained from GPs) at $t_\star$. Unlike the approach described in Williams et al. (2011), which truncates the probability distribution to $[0, 1]$, OMAC★ preserves the probability distribution by introducing the normalizing constant C.

OMAC★ distinguishes two cases with respect to the expected discounted received utility. First, the expectation of $E_\delta(t)$ is optimistic, that is, the expected received utility is larger than what is suggested by the conservative aspiration level function. Formally, this means that there exists an interval $\{T \mid T \neq \emptyset, T \subseteq [t_c, t_s]\}$, where t_c is the current time slot and t_s the end point of the predicated time series, so that

$$E_\delta(t) > D^\delta(R(t), t), \quad t \in T \tag{19}$$

In this case the time $\hat{t}$ at which the optimal expected utility $\hat{u}$ is reached is set as follows:

$$\hat{t} = \underset{t \in T}{\text{argmax}}\, E_\delta(t) \tag{20}$$

Moreover, $\hat{u}$ is defined as

$$\hat{u} = E_\delta(\hat{t}) \tag{21}$$

Second, the expected received utility is below the suggested aspiration level. In this pessimistic case $\hat{u}$ is defined as 0 and OMAC★ abides by the target utility determined by $R(t)$. By distinguishing

[4] An opponent model that is too sensitive to opponent behavior tends towards making higher concession than necessary to reach an agreement with that opponent. Throughout this paper this is referred to as "adverse concession behavior" and " irrational concession".

these two cases, OMAC★ aims at "getting the most with lowest possible risk".

Obviously, it would be not rational to concede immediately to $\hat{u}$ when $u_l \geqslant \hat{u}$, where u_l is the utility of the last bid before the opponent model is updated at time t_l. Similarly, it would be not appropriate for an agent to immediately switch to $\hat{u}$ if $u_l < \hat{u}$. Therefore, OMAC★ dynamically adjusts the concession rate by setting the target utility u' as follows:

$$u' = \begin{cases} R(t_c) & \text{if } \hat{u} = 0 \\ \hat{u} + (u_l - \hat{u})\frac{t_c - \hat{t}}{t_l - \hat{t}} & \text{otherwise} \end{cases} \tag{22}$$

5.3. Counter-offer response mechanism

After the target utility u' has been chosen, an agent has to decide how to respond to the opponent's current counter-offer (this corresponds to lines 13–19 in Algorithm 1). OMAC★ first checks whether any of the following two conditions is fulfilled:

- the utility of the latest counter-offer $U(O_{opp})$ is not smaller than u',
- or, the counter-offer has been proposed by the agent itself to its opponent at some earlier point during the ongoing negotiation process.

If any of these two conditions is satisfied, the agent settles the deal and the negotiation ends (line 14).

Otherwise, OMAC★ checks whether u' falls below the best counter-offer received so far. If this is the case, then, for the reason of negotiation efficiency, this counter-offer is proposed to the opponent. Such an action is reasonable because it tends to satisfy the expectation of the opponent. If not the case, then OMAC★ constructs a new offer following a ϵ-greedy strategy (Chen & Weiss, 2014). According to this strategy, a greedy offer with probability 1-ϵ is chosen in order to exploit the opponent behavior, and with probability ϵ, a random offer[5] is made (where $0 \leqslant \epsilon \leqslant 1$). The greedy offer is chosen as follows. For a rational opponent it is reasonable to assume that the sequence of its counter-offers is in line with its decreasing satisfaction. Thus, the more frequent and earlier a value of an issue j appears in counter-offers, the more likely it is that this value contributes significantly to the opponent's overall utility. Formally, let $F(\cdot)$ be the frequency function defined by:

$$F^n(v_{jk}) = F^{n-1}(v_{jk}) + (1-t)^{\varphi} \cdot g(v_{jk}) \tag{23}$$

where the superscript of $F(\cdot)$ indicates the number of negotiation rounds, φ is the parameter reflecting the time-discounting effect, and $g(\cdot)$ is a two-valued function whose output is 1 if the specific issue value (i.e., v_{jk}) appears in the counter-offer and 0 otherwise. The new offer to be proposed is the one whose issue values have the maximal sum of frequencies according to the frequency function and whose utility is not worse than the current target utility. For efficiency purposes, the updating only considers those issue values that have been proposed by the opposition, and takes place for the early stage of a negotiation session. In the case of a random offer, an offer whose utility is within a narrow range around u' is randomly generated and proposed at next round.

An important decision to be made by a negotiating agent is whether or not an ongoing negotiation should be broken off. This can make sense especially in negotiations with tough opponents if the reservation value (ϑ) is non-zero and the time-discounting effect (δ) is severe. In this situation, the agent may obtain a better payoff by aborting the tough negotiation as early as possible, namely, a slightly discounted reservation value rather than only a highly discounted outcome based on a late agreement. OMAC★ uses the following probability η to decide on breaking off a negotiation:

$$\eta = \begin{cases} 0 & \text{if } \tilde{\mu} \geqslant \vartheta \\ (\vartheta - \tilde{\mu})(1-\delta) & \text{otherwise} \end{cases} \tag{24}$$

where $\tilde{\mu}$ is the maximum mean value of the gained prediction, ϑ is the reservation value, and δ is the time-discounting factor. According to this definition, η is proportional to ϑ but inversely affected by the maximum concession prediction and δ. The rationale behind this is that a high reservation tends to make a significant negotiation success less likely, while a small discounting factor (implying high time pressure) reduces the payoff quickly. OMAC★ handles breaking-off rather conservative: before really breaking off, the opponent's forthcoming counter-offers are analyzed for a certain period of time (5% of the overall negotiation time in the experiments reported below), and break-off eventually happens if none of these counter-offers is better (i.e., concedes more) than best counter-offer received so far.

6. Experimental results and analysis

This section is organized as follows. The setup of the experiments is described in Section 6.1. The results and detailed comparisons of OMAC★ with other strategies are shown in Section 6.2. Finally, Section 6.3 summarizes the performance of our proposed approach.

6.1. Experimental setup

6.1.1. Automated Negotiating Agent Competition

The Automated Negotiating Agent Competition (ANAC) is a yearly international competition, which was jointly initialized by the Delft University of Technology and Bar-Ilan University to encourage the development of practical agents that are able to proficiently negotiate against unknown opponents in uncertain circumstances. With a large number of state-of-the-art negotiating agents and negotiation domains, it provides a useful benchmark for objectively evaluating negotiation strategies. ANAC uses the Generic Environment for Negotiation with Intelligent multi-purpose Usage Simulation (Genius) (Hindriks, Jonker, Kraus, Lin, & Tykhonov, 2009) as the official test platform. This framework can support negotiation sessions where the behavior and preferences of opponents are unknown and where the negotiation sessions are subject to discounting effects and real-time constraints. Genius allows to compare new negotiation strategies against various state-of-the-art negotiation agents that have been implemented within this framework.

In a competition, each agent plays against other agents in every considered domain (see 6.1.2), where the two agents involved in a negotiation act in turn in conflicting roles (e.g., "buyer" and "seller"). Suppose agent a negotiates with b in domain $D \in \mathcal{D}$, where $\mathcal{D}$ is the whole set of domains. Let the two roles of domain D be represented by P_1^D and P_2^D, respectively, and $U_{a\to b}^D(a, P_1^D)$ represent the score of agent a in a session where a initializes it with playing as the role of P_1^D. Then, the score of agent a, when playing the role given by P_1^D, is calculated as

$$S(a, P_1^D) = \frac{1}{2(|\Lambda| - 1)} \sum_{b \in \{\Lambda \backslash a\}} (U_{a\to b}^D(a, P_1^D) + U_{b\to a}^D(a, P_1^D)) \tag{25}$$

and the score of a, when playing the other role P_2^D, is given by

[5] Random offers randomly select issue values, rather than utilities. No matter what offers the agent chooses, the utilities are determined by the target utility u' (see Eq. (22)). Please note that, for the sake of efficacy, if no appropriate offers can be found after a number of trials, the agent alternatively searches its own bidding history for the offer that is most close to the target utility.

$$S(a, P_2^D) = \frac{1}{2(|\Lambda| - 1)} \sum_{b \in \{\Lambda \backslash a\}} (U_{a \to b}^D(a, P_2^D) + U_{b \to a}^D(a, P_2^D)) \quad (26)$$

The final score of agent a is then given by

$$S(a) = \frac{S(a, P_1^D) + S(a, P_2^D)}{2} \quad (27)$$

6.1.2. Test domains and benchmark agents

We conducted a variety of experiments with domains of different complexity, where complexity is characterized by two key factors: competitiveness and domain size. Competitiveness represents the minimum distance from all of the points in the outcome space of a domain to the point leading to a complete satisfaction for both sides (note that such an ideal solution may not be always available). As competitiveness increases, it thus becomes more and more difficult to reach an agreement that meets both sides' interests. The domain size refers to the number of possible agreements or the scale of the outcome space of a domain. The larger the domain size is, the more important is the efficiency of an agent's negotiation approach because only a possibly very small fraction of the outcome space can be explored under time constraints.

The domains we used were all chosen from the available ANAC domains. We group the domains into four groups: Groups I, II and III contain domains of low, medium and high competitiveness, respectively, and Group IV contains domains having a large outcome space. Specifically, Group I contains the domains *IS BT Acquisition*, *Music Collection*, and *Laptop*, *Phone*; Group II contains the domains *Amsterdam party*, *Barbecue*, *Flight booking*, and *Airport selection*; Group III consists of the domains *Itex vs Cypress*, *Barter*, *Fifty fifty*, *NiceOrDie*; and Group IV contains the domains *ADG*, *SuperMarket*, *Travel*, and *Energy*. All domains used in the experiments are overviewed in Table 1. For descriptions of these domains the readers are referred to ANAC (2012), Baarslag, Hindriks, Jonker, Kraus, and Lin (2012) and Fujita et al. (2013).

Furthermore, for assessing the effect of the discounting factor and the reservation value on the performance of the strategies (or agents), different values for these two parameters are considered. More precisely, we conducted experiments with three discounting factor parameters (i.e., $\delta = \{0.5, 0.75.1.0\}$) and three reservation value parameters (i.e., $\vartheta = \{0, 0.25, 0.5\}$), which resulted in nine (3×3) different scenarios for each domain.

As benchmark agents for the experimental evaluation of OMAC* we used the three best-performing agents of each of the 2010, 2011 and 2012 ANAC competitions. An overview of these agents, which together form a highly competitive negotiation setting, is given in Table 2. For details on the benchmark agents, we refer to ANAC (2012), Baarslag et al. (2012), Chen and Weiss (2012), Fujita et al. (2013) and Hao et al. (2014).

Table 1
Overview of application domains.

Group	Domain name	Year	Issues	Domain size	Competitiveness
I	IS BT acquisition	2011	5	384	0.117
I	Music Collection	2012	6	4320	0.150
I	Laptop	2011	3	27	0.160
I	Phone	2012	5	1600	0.188
II	Amsterdam party	2011	6	3024	0.223
II	Barbecue	2012	5	1440	0.238
II	Flight booking	2012	3	36	0.281
II	Airport site selection	2012	3	420	0.285
III	Itex vs Cypress	2010	4	180	0.431
III	Barter	2012	3	80	0.492
III	Fifty fifty	2012	1	11	0.707
III	NiceOrDie	2011	1	3	0.840
IV	ADG	2011	6	15,625	0.092
IV	SuperMarket	2012	6	98,784	0.347
IV	Travel	2010	7	188,160	0.230
IV	Energy	2012	8	390,625	0.525

Table 2
Overview of benchmark agents.

Agent	Affiliation	Achievement	
		Ranking	Competition
CUHKAgent	Chinese University of Hong Kong	1st	ANAC 2012
AgentLG	Bar-Ilan University	2nd	ANAC 2012
OMAC	Maastricht University	3rd	ANAC 2012
HardHeaded	Delft University of Technology	1st	ANAC 2011
Gahboninho	Bar Ilan University	2nd	ANAC 2011
IAMhaggler2011	University of Southampton	3rd	ANAC 2011
Agent_K	Nagoya Institute of Technology	1st	ANAC 2010
Nozomi	Nagoya Institute of Technology	2nd	ANAC 2010
Yushu	University of Massachusetts Amherst	3rd	ANAC 2010

6.1.3. Basic tournament and OMAC* setting

The empirical evaluation is done with GENIUS, which is the official platform used for the international ANAC competition. It allows to compare agents (representing different negotiation strategies) across a variety of application domains under real-time constraints. For each scenario of each domain, we run a tournament consisting of ten agents (including OMAC* and other nine competitors) ten times to get results with statistical confidence. In each tournament each agent repeats negotiation against the same opponent four times, where they exchange both their negotiation roles (i.e., buyer and seller role) and the order in which they start with bidding. No information about the opponents' strategies or other private information is available to any of the agents, and none of them can take advantage of previous encounters with their opponents (which is assured by the GENIUS platform). The maximum time for every negotiation session is 180 s since it is the default setting in GENIUS and ANAC competitions. When there is no agreement reached at the end of a session, then the disagreement solution applies, which means that each agent receives its own reservation value.

In our experiments OMAC* performed effectively and very robust for a broad range of parameter settings. Table 3 shows a concrete parameter setting used in the experiments reported in this article.

6.2. Experimental results

6.2.1. Evaluating effectiveness of opponent models

In this subsection, we compare the proposed opponent modeling scheme with other important methods that also aims at learning the opponent's strategy by means of predicting future concession. The opponent modeling component of OMAC* is benchmarked against two main competitors, EMD + ARMA and Gaussian processes (GPs), which are employed by EMAR (Chen & Weiss, 2013) and IAMhaggler2011 (Williams et al., 2011), respectively. These models are applied to predict the utilities of future offers proposed by negotiation partners in all domains given in Table 1 with two different time-constraint scenarios: negotiations with a short negotiation deadline (i.e., 60 s), and negotiations with the standard ANAC deadline (i.e., 180 s). In this way, the performance w.r.t. small and large numbers of offer exchanges can be both assessed. Moreover, the benefit of combining the two modules (i.e., DWT and GPs) can be verified.

Table 3
Overview of primary parameter settings.

Parameter	Description	Value	Comment
λ	Decomposition level	4	No significant performance differences for more layers (i.e., values $\geqslant 4$)
ξ	Lead time	15	10% of the maximum interval, too large values decrease prediction accuracy
p	Probability used for opponent concession prediction	0.9	The probability should be high enough to ensure a strong confidence about the prediction results
β	Concession coefficient	0.1	The higher the value the more cooperative the agent becomes
ϵ	Probability of random offers	0.5	Equal chances for exploration and exploitation
φ	Time-discounting coefficient	1.5	The higher the value the less important counter-offers later on are taken as

Table 4
The RMS errors averaged over the three opponents on each domain. Bold means the value significantly better than GPs (95% confidence in each case based on Welch's t test).

Opponent model	IS BT Acquisition		Music Collection		Laptop		Phone	
	Short	Regular	Short	Regular	Short	Regular	Short	Regular
GPs	1.15	1.46	4.1	1.7	2.52	2.51	8.11	6.67
GPs + DWT	1.64	**0.85**	3.38	0.99	2.49	**1.45**	**5.66**	**3.72**
EMD + ARMA	4.12	3.15	5.52	4.0	3.36	3.04	6.40	6.44
	Amsterdam party		Barbecue		Flight booking		Airport selection	
GPs	4.14	6.74	7.51	6.59	5.94	6.67	3.93	3.72
GPs + DWT	4.21	**3.12**	6.22	**4.07**	5.41	**3.59**	3.37	**1.98**
EMD + ARMA	5.03	**4.67**	6.57	6.13	5.58	5.36	4.74	4.61
	Itex vs Cypress		Barter		Fifty fifty		Nice or die	
GPs	5.05	3.81	3.66	3.03	1.53	1.99	0.01	1.76
GPs + DWT	**4.37**	**2.38**	**3.10**	**1.83**	1.39	**1.52**	0.15	1.71
EMD + ARMA	**3.77**	**3.34**	3.62	3.08	**1.05**	**1.41**	1.63	2.1
	ADG		SuperMarket		Travel		Energy	
GPs	5.95	5.23	6.05	6.31	7.71	7.25	4.98	4.07
GPs + DWT	**3.48**	**3.17**	**3.05**	**1.71**	**3.97**	**3.24**	**4.62**	**2.95**
EMD + ARMA	5.24	**4.18**	**4.24**	**3.92**	**4.47**	**3.83**	**4.03**	3.80

The results are shown in Table 4, where root mean square errors (RMSE) under short and standard time-constraints are listed for each model in each domain. As can be seen from this table, the performance of models tends to increase with negotiation time. More precisely, there was on average a difference of 14.4% between them. This indicates that a training of the models with more samples improves their performance in terms of prediction accuracy. Furthermore, the results show that the opponent modeling component of OMAC* is also robust for short-time-period negotiations, obtaining an average RMSE of 3.53%. Moreover, OMAC* outperforms others with a much higher accuracy rate in both cases. Specifically, it managed to achieve lower RMSE: around 80% (in the case of short time-constraints) and 58% (in the case of regular time-constraints) of the mean RMSE of others. With respect to regular negotiation deadlines, GPs + DWT was significantly better than the other approaches (using Welch's t test). Overall, these results show that the new learning scheme – DWT + GPs – outperforms GPs as well as EMD + ARMA.

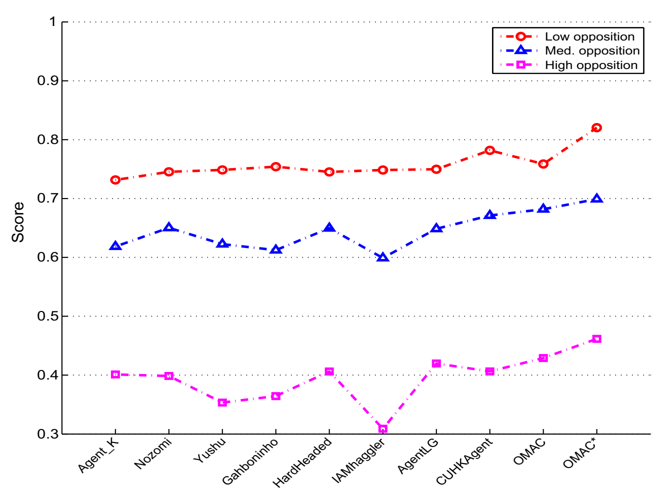

Fig. 3. Scores of agents under different levels of opposition.

6.2.2. Performance in different levels of competitiveness

Based on the domains from Group I to III, Fig. 3 shows the agents' performance under low, medium and high competitiveness. For each domain the influence of the discounting factor and the reservation value are taken into consideration by using the resulting nine possible scenarios (as described above). As can be expected, all agents managed to increase their profit as competitiveness decreases. OMAC* was the winner in all three groups, where the distance to the other agents grows with the level of competitiveness. OMAC made the 2nd place in Groups II and III, and CHUHKAgent made the 2nd place in Group I. The performance of IAMhaggler2011 dropped dramatically as the competitiveness gets stronger.

Table 5 summarizes the results for Groups I, II and III. Overall OMAC* was the most successful agent, which was above the average performance of other negotiators by up to a 19% in the most competitive Group III. According to our analysis the main reason for this is the ability of OMAC* to estimate with high precision the future concession an opponent will make. Due to this estimate, an agent using OMAC* concedes less especially in highly competitive domains. In less competitive domains (Groups I and II), where it is more likely that win–win solutions exist and thus agreements can be found with less compromise, this ability of OMAC* tends to have a lower impact. As can be also seen from the table, OMAC* is a clear improvement of OMAC, which is the second-best agent. OMAC achieved scores of about 94% of OMAC*.

Table 5
Performance summary of each opposition level, ordered by the agents' overall ranking (see Table 8).

Agent	Low opposition (G I)	Medium opposition (G II)	High opposition (G III)
OMAC★	**0.820**	**0.699**	**0.462**
OMAC	0.759	0.682	0.429
CUHKAgent	0.782	0.671	0.406
AgentLG	0.750	0.649	0.420
Nozomi	0.745	0.650	0.399
HardHeaded	0.745	0.650	0.406
Agent_K	0.732	0.619	0.401
Gahboninho	0.754	0.612	0.364
IAMhaggler2011	0.748	0.599	0.309
Yushu	0.749	0.622	0.353

The value in bold means the highest score in each class.

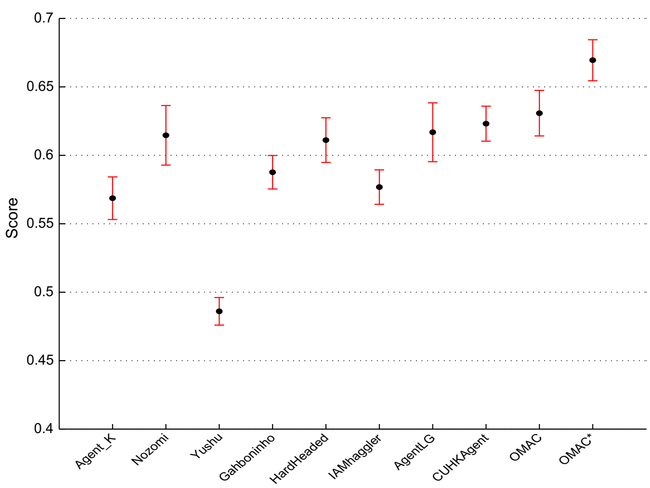

Fig. 4. Performance of agents in the largest outcome space domains (Group IV). The score of each agent is represented by the bold dot. Error bars are indicated by vertical lines.

6.2.3. Performance in the large outcome space domains

Group IV contains the four largest ANAC domains (with sizes ranging from 15,625 to 390,625) and has an average domain size of more than 170,000. The results for this group are shown in Fig. 4. As done for the other groups (see above), the results are averaged over all scenarios of the included domains in order to cover a sufficiently broad range of values for the discounting factor and the reservation value. As can be seen from the table, OMAC★ again outperformed all other competitors. OMAC and CUHKAgent finished second and third, respectively. OMAC★ achieved a score of 0.669, which is 13% higher than the mean score of the opponents. As these results also show, there was an increase in the performance of the best agents of ANAC 2010, ANAC 2011, ANAC 2012. Specifically, the 2010, 2011 and 2012 agents on average obtained a score of 83%, 88% and 93% of OMAC★. OMAC performed better than the other ANAC agents, but remained 6% below the score achieved by OMAC★. Overall, the results for Group IV confirm the suitability of OMAC★ for very large domains.

6.2.4. Evaluation of the impact of the discounting factor

As the discounting factor decreases, the payoff of the participants is increasingly affected over time. It is therefore interesting to investigate the performance of the agents (respectively their negotiation strategies) for different time-discounting levels. For that purpose we partition all available scenarios into three classes ($\delta = \{0.5, 0.75, 1.0\}$) according to their discounting factor. This comparison is presented in Table 6. As this table shows, an increase of δ (hence a decrease of the time pressure) results in an increase of

Table 6
Scores of agents averaged over all scenarios, grouped by the discounting factor and ordered by their overall ranking.

Agent	Score		
	$\delta = 0.50$	$\delta = 0.75$	$\delta = 1.00$
OMAC★	**0.553**	**0.640**	**0.795**
OMAC	0.506	0.613	0.757
CUHKAgent	0.504	0.615	0.743
AgentLG	0.506	0.601	0.718
Nozomi	0.502	0.598	0.706
HardHeaded	0.489	0.577	0.743
Agent_K	0.473	0.572	0.694
Gahboninho	0.497	0.518	0.724
IAMhaggler2011	0.511	0.560	0.604
Yushu	0.451	0.564	0.643

The value in bold means the highest score in each class.

the scores achieved by the agents. A comparison with the non-discounting case ($\delta = 1$) reveals that the agents' mean score dropped by 30% to 0.499 for $\delta = 0.5$ and by 18% to 0.586 for $\delta = 0.75$. Each of the three agents IAMhaggler2011, CUHKAgent and OMAC finished second in one of the three classes. OMAC★ performed best in all three classes, with a performance that was 12.1% above the average performance of the others for $\delta = 0.5$, 10.3% for $\delta = 0.75$, and 13% for $\delta = 1$. Interestingly, the smallest difference between OMAC★ and its opponents occurred for the medium time-discounting factor ($\delta = 0.75$). Our analysis indicates that this was the case because most opponents are optimized for a medium discounting level. Overall, these results show the ability of OMAC★ to adapt effectively to different time-discounting levels.

6.2.5. Evaluation of the impact of the reservation value

For the purpose of better understanding the impact of the reservation value, we divide all scenarios according to the used reservation value ($\vartheta = 0, 0.25$, and 0.5) into three classes. Table 7 shows the performance results of the agents achieved for each of these three classes. A general observation from these results is that these agents achieved higher scores for higher reservation values (only one agent, Yushu, performs somewhat worse for $\vartheta = 0.25$). Importantly, OMAC★ performed best in all three classes, and OMAC was the second best agent, obtaining an average score of about 6% below OMAC★. The advantage of OMAC★ over the others decreased gradually with increasing reservation values. Specifically, the largest difference (12%) was achieved for $\vartheta = 0$, whereas the difference dropped to 10.4% and 9.6% for $\vartheta = 0.25$ and $\vartheta = 0.5$, respectively. The reason behind it is that for higher values of ϑ (especially when $\vartheta = 0.5$), the negotiations tend to end up with disagreement solutions since in some domains (e.g., Fifty fifty and NiceOrDie) utility of proposals is hard to meet the expectation of each other. On the contrary, it is much easier for our agent OMAC★ to realize an increased advantage for lower ϑ by exploring opponents as generated offers are normally better than reservation value[6] in this case.

6.3. Performance summary

The overall performance of the agents is summarized in Table 8, where the normalized mean score and standard deviation are given. Normalization is done in the standard way, using the maximum and minimum utility obtained by all agents. In addition, to calculate the rank of each agent, Welch's t test was used to check for statistically significant differences between the agents' ANAC scores (also see ANAC, 2012). More precisely, we computed this for every single pair of agents in order to determine with 95% confidence which agents defeat a specific agent, and which agents are

[6] Note that an agent does not know the reservation value of its opponent.

Table 7
Scores of agents averaged over all scenarios, grouped by the reservation value and ordered by their overall ranking.

Agent	Score		
	$\vartheta = 0.00$	$\vartheta = 0.25$	$\vartheta = 0.50$
OMAC★	**0.656**	**0.657**	**0.675**
OMAC	0.609	0.619	0.647
CUHKAgent	0.604	0.615	0.643
AgentLG	0.585	0.607	0.634
Nozomi	0.583	0.599	0.624
HardHeaded	0.581	0.602	0.626
Agent_K	0.568	0.576	0.595
Gahboninho	0.568	0.575	0.596
IAMhaggler2011	0.553	0.557	0.565
Yushu	0.553	0.547	0.558

The value in bold means the highest score in each class.

Table 8
Overall performance of all agents across all tournaments in descending order. The letter in bold of each strategy is taken as its identifier for the later EGT analysis.

Agent	anking	Normalized score	Standard deviation
O**M**AC★	1	0.667	0.010
OMAC	2–3	0.603	0.007
CUHKAgent	2–3	0.601	0.005
Agent**LG**	4	0.581	0.009
Nozomi	5–6	0.570	0.009
HardHeaded	5–6	0.566	0.008
Agent_K	7	0.535	0.008
Gahboninho	8	0.526	0.007
IAMhaggler2011	9	0.511	0.005
Yushu	10	0.502	0.006

beaten by that agent. (For instance, in this setting an agent that is beaten by one agent and beats seven agents is considered to have rank 2–3, that is, with 95% certainty the rank of this agent lies between 2 and 3.) The best overall performance was achieved by OMAC★, with a noticeable distance of 20% above the average overall performance (normalized score) of the other agents. Moreover, the performance of OMAC★ was 10% above that of OMAC, which achieved the second highest score. The difference between OMAC and CHUKAgent was not significant and therefore both made the 2–3 place. Given these results, OMAC★ clearly outperformed all considered state-of-the-art agents in a variety of scenarios. It also performed much better than its predecessor OMAC, especially in domains with a high time-discounting factor and domains with a low competitiveness.

Interestingly, IAMhaggler2011 (Williams et al., 2011), which also employs GPs, achieved merely 77% of the scores of OMAC★. We looked into this and found that there are two main reasons causing this performance gap: IAMhaggler2011 adjusts the concession rate according to the maximum predicted utility and the corresponding time, and the prediction of an opponent's future moves is done in a "global" way, that is, on the basis of the whole preceding negotiation process. This kind of adaptive behavior makes IAMhaggler2011 vulnerable to "irrational concession" induced by pessimistic predictions (see Section 5.2 where it is explained how OMAC★ avoids this problem). The phenomenon of irrational concession becomes increasingly apparent when IAMhaggler2011 bargains in no-time-pressure scenarios with "tough" opponents. For instance, when competing against the top three agents listed in Table 8 in the *Amsterdam party* domain, IAMhaggler2011 obtained an average score of only 0.533, which was 55% of the mean score of those three agents.

Having compared OMAC★ against the best overall performance agents (i.e., the ANAC winners), it is also of interest whether OMAC★ can outperform the winner of each domain (note that the winner of ANAC is not necessarily the winner of each domain). Therefore, in the following experiments OMAC★ competes in each domain against the domain winner. The results are given in Fig. 5. The scoring bar of OMAC★ in a domain is marked with dots if the difference between the two agents in that domain is not significant (again using Welch's t test with 95% confidence). As can be seen, in most domains OMAC★ achieved higher scores than the domain winner: the performance difference is significant in 10 domains. It was only beaten in the Itex vs. Cypress domain, but the difference was not significant.

7. Empirical game theoretical analysis

The experimental analysis we provided above aims at investigating the performance of negotiation strategies from the common mean-scoring perspective using a tournament setting. Although this analysis gives valuable insight (and thus is common in the field), it says only little about the robustness of the strategies because the basic setting of the tournament (especially the number of participating players and strategies) is fixed. In particular, this

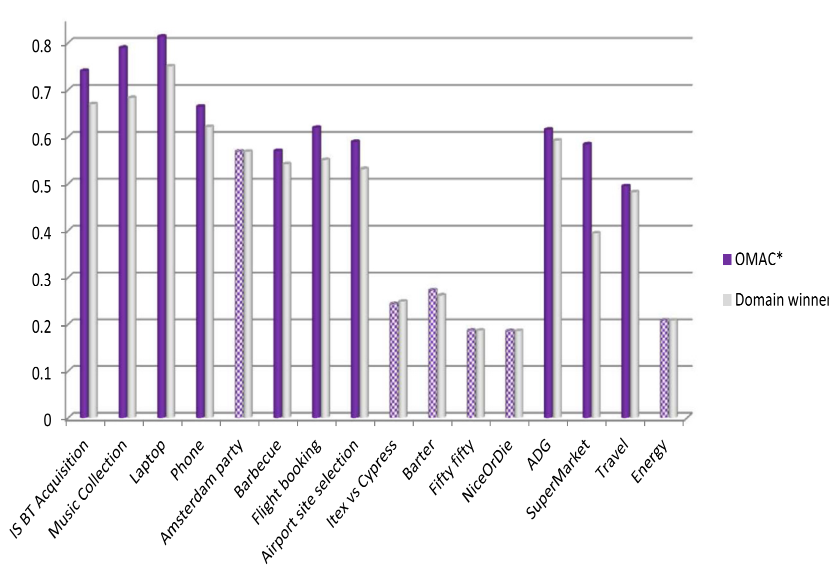

Fig. 5. Comparing our agent against the winner of each domain. Insignificant differences are indicated by bars marked with dots.

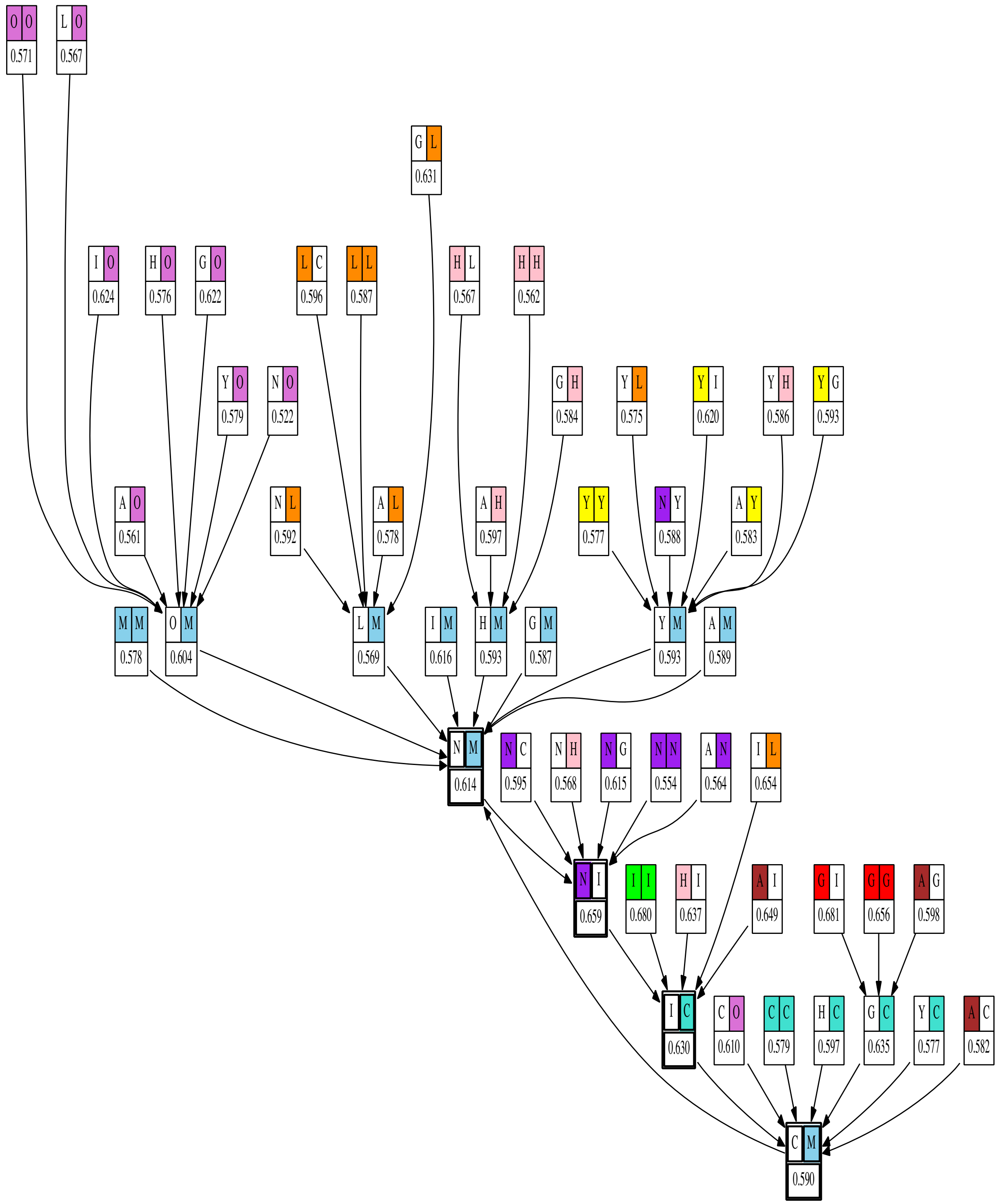

Fig. 6. Deviation analysis for two-player encounters composed of all strategies. Each node shows a strategy profile. The first row of a node shows the involved strategies, where the strategy with the higher score is indicated by a colored background. The second row gives the average score of the strategy pair. An arrow indicates a statistically significant single-agent deviation between strategy profiles. The profiles in the best cycle are the nodes with a bold frame. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

measurement failed to shed light on the performance of a strategy in case agents are allowed to switch their strategy. To address robustness appropriately, empirical game theory (EGT) analysis (Baarslag et al., 2013; Jordan, Kiekintveld, & Wellman, 2007; Williams et al., 2011) is applied to the tournament results. The purpose of this analysis technique is to identify pure Nash equilibria where none of the agents has an incentive to change its current strategy or, in case no such equilibrium exists, to find the best reply cycle (Young, 1993). Such a cycle consists of a set of profiles (e.g., the combination of strategies chosen by players) for which a path of statistically significant single-agent deviations (whose definition is given next) exists that connect them, with no deviation leading to a profile outside of the set. These two types are both referred to as empirical equilibria in this work.

In EGT analysis a profile consists of a combination of strategies used by players in the game, where some of them may use the same strategy. Strictly speaking, it is called the pure profile since players are only allowed to use a single strategy instead of choosing a strategy probabilistically. The payoff of each strategy in a profile is determined by the tournament results (see Wellman (2006) for a description of this approach). A profile is represented by a node in the resulting graph. To study the behavior of agent switching strategies (or profile transition), we consider the statistically significant single-agent deviations (as done in Williams et al. (2011)), where there is an incentive for an agent to unilaterally change its strategy in order to statistically improve its own profit, given the strategies of others are known. In the following, three cases of varying complexity are considered to analyze the robustness of strategies in view of game theory:

Case 1: Single negotiation encounters between two players.
Case 2: Tournament setup composed of seven players where only the top three strategies in Table 8 are available.
Case 3: Tournament setup with a full combination of players and strategies.

Regarding Case 1, we apply EGT analysis to the single negotiation case where two players are involved and each of them can freely choose among the strategies that are considered in the experiments described above. For brevity, in the following each strategy is referred to by a single letter, namely the respective bold letter in Table 8 (e.g., **C** stands for CUHKAgent). Let S be the complete strategy set, that is, $S = \{\mathbf{A}, \mathbf{N}, \mathbf{Y}, \mathbf{G}, \mathbf{H}, \mathbf{I}, \mathbf{L}, \mathbf{C}, \mathbf{O}, \mathbf{M}\}$. The score of a strategy in a specific profile is the payoff achieved when playing against the other strategy and averaged over all scenarios considered in the above experiments. The analysis results are shown in Fig. 6. The first row of each node gives the pair of strategies of a profile; the second row shows the average score achieved by the strategy pair. This average score is used as a measure of the social welfare achieved by the two involved strategies, which can be interpreted as overall benefit achieved by a profile for all involved agents. The stronger strategy in each profile is marked with a color background. Each arrow indicates a statistically significant single agent-deviation to a different strategy profile. Under this EGT analysis, no pair of strategies is in equilibrium; instead, there exists a best cycle of statistically significant single-agent deviations. This best cycle contains four profiles given by NM, NI, IC, CM. For any other strategy profile not included in this cycle, there exist a path of statistically significant deviations (i.e., strategy changes) that lead to a profile within the cycle. The highest social welfare (i.e., 0.681) is achieved by the profile GI; this profile, however, is not included in the best cycle. Moreover, despite the fact OMAC is the second best strategy of the competition, it is not contained in any of the best-cycle profiles.

To summarize, in single negotiation encounters there are four strategies – OMAC*, CUHKAgent, IAMhaggler2011, Nozomi – that are robust in the sense that they are in the empirical equilibria into which all other strategies eventually lead to (i.e., they are the basins of attraction Baarslag et al. (2013) is 100%) and which are chosen by the negotiating players with equal probability (because they are symmetric). This result also indicates that high-scoring strategies (e.g., OMAC) do not necessarily perform well in single encounters, or in other words, they are not necessarily stable. However, it is important to see that the single encounter analysis, while useful, provides quite limited information when the setup gets more complicated. Therefore, in the following we look into a setting involving seven players and the top three strategies.

We now consider a seven-player tournament setting (Case 2) where the players can freely switch among the top three strategies (i.e., OMAC*, OMAC and CUHKAgent, see Table 8). The strategy deviations are visualized in Fig. 7, where each node represents a profile, the first row of each node lists the three strategies, and the second row shows how many players use each strategy. The strategy with the highest score is marked with a color background. In this restricted 3-strategy tournament, there exists only one equilibrium and this equilibrium contains OMAC* and OMAC. It is clear that for any non-Nash equilibrium strategy profile there exist a path of statistically significant deviations that leads to this equilibrium state. OMAC is the winner of the tournament specified by the equilibrium state, whereas OMAC* is more popular voted by most agents. The two players using OMAC are not interested in switching to OMAC* (although OMAC lags behind OMAC*),

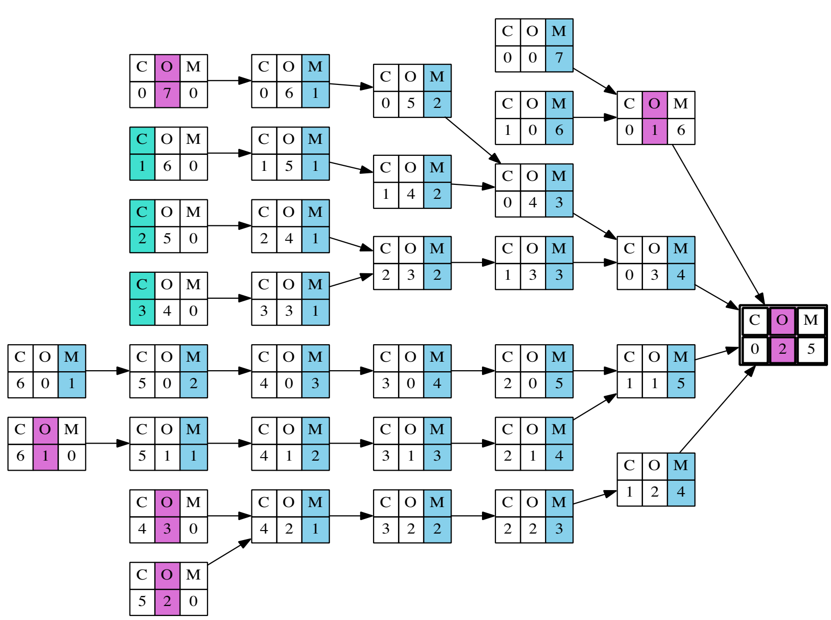

Fig. 7. Deviation analysis for a seven-player three-strategy tournament setting. Each node represents a strategy profile, where the strategy with a color background achieves the highest score. Arrows indicate the statistically significant deviations between strategy profiles. The node with a bold frame is the only equilibrium (no outgoing arrow). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

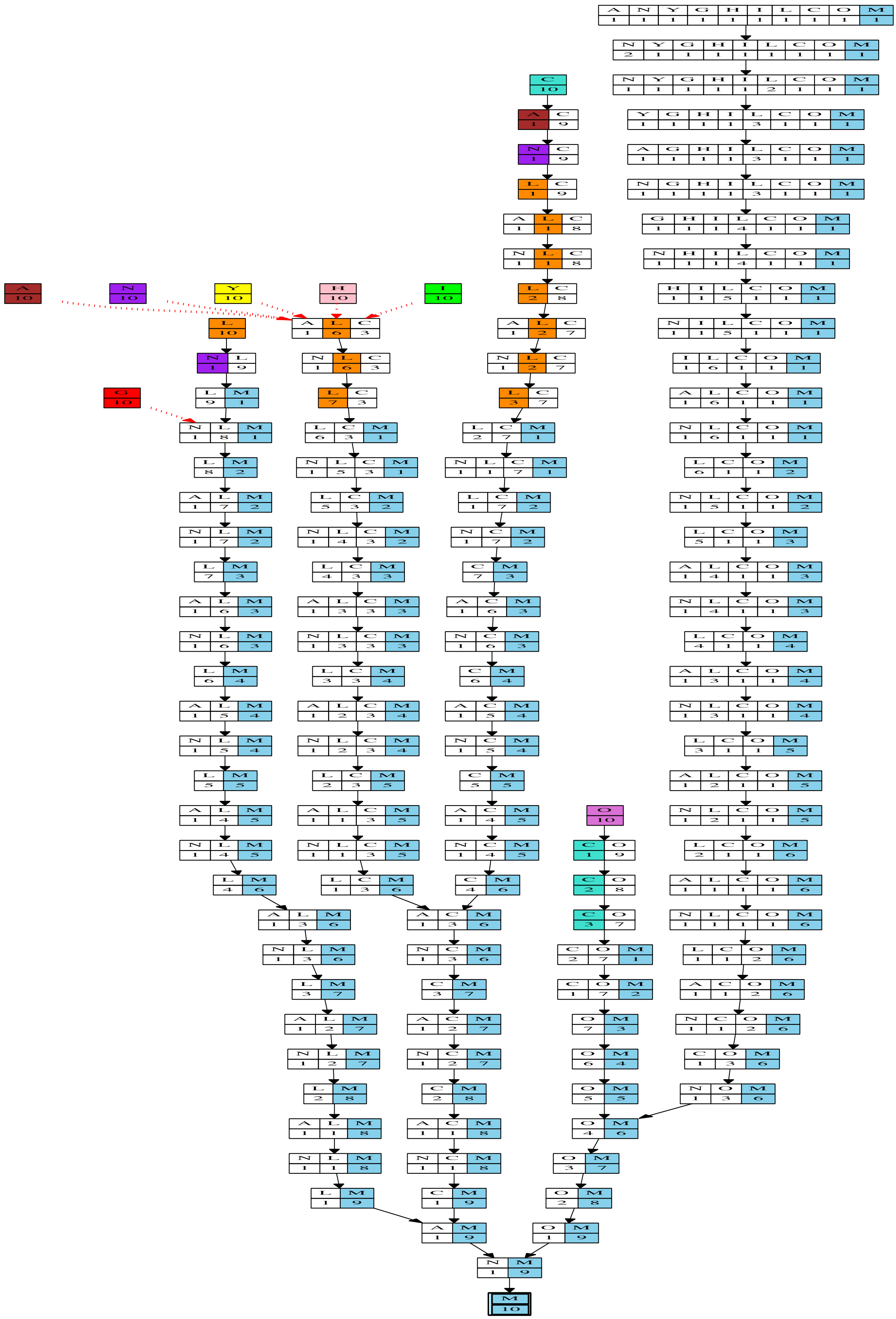

Fig. 8. The deviation analysis for ten-player tournaments composed of all strategies. Each node shows a strategy profile, where the best strategy is marked with a color background. Each arrow indicates a statistically significant deviation to a different strategy profile. The only equilibrium is the node having a bold frame and no outgoing arrow. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

because it offers them a profit which is better than the profit that could be achieved in an OMAC* self-play setting. On the other hand, there is no incentive for the other players to switch from OMAC* to OMAC because this would result in a decrease of their benefit. However, if a player knows in advance that opponents are in favor of OMAC*, it probably benefits (i.e., obtains a higher score) from playing OMAC as well.

Regarding Case 3, we investigate the most complex case – the tournament setting with all players and strategies involved, thus with ten players that freely choose from ten strategies. Visualization of the complete resulting graph is not possible due to the large number of distinct nodes. More precisely, the complete graph includes $\binom{|p|+|s|-1}{|p|} = \binom{19}{10} = 92,378$ distinct nodes, where $|p|$ is the number of players and $|s|$ is the number of strategies. Therefore, we prune the graph to concentrate on relevant features in a way similar to Baarslag et al. (2013). In more detail, we blank all nodes in the graph except those being on a path that starts either with an initial profile where all players choose the same strategy or with an initial profile where each agent uses a different strategy and that ends with a pure Nash equilibrium. Furthermore, for the sake of a compact visualization we omit the beginning parts of all deviation paths that start with a profile where all agents choose the same strategy from the following set of strategies: Agent_K, Nozomi, Yushu, Gahboninho, HardHeaded and IAMhaggler2011. The resulting graph is shown in Fig. 8. The first row of each node indicates the involved strategies, and the second row gives the number of players using each of the strategies. A strategy that is not used by any players is not displayed in order to keep the graph as compact as possible. Using this EGT analysis, it can be seen that there is only one equilibrium profile, namely the one where all players choose OMAC*; any other strategy profile eventually converges to this equilibrium state. This also means that OMAC* becomes the dominant strategy in this context. Overall, EGT analysis shows that OMAC* is robust in all three negotiation setups and its robustness increases with the complexity of the negotiation setup.

8. Discussion

Our analysis provides general insights into the proficiency of practical bargaining agents. For instance, a cooperative attitude towards opponents in complex negotiations should be implemented very carefully. In particular, the results show that the willingness to easily adapt to opponent behavior (irrational concession) can prevent a successful negotiation outcome. Moreover, the reported results show that the performance of an agent can be very diverse in different negotiation scenarios – an agent can obtain rather high scores in low-opposition domains but only low scores in high-opposition domains, and vice versa. Similar effects can be caused by the discounting factor. This implies that the evaluation of an agent's performance should be based on an as-broad-as-possible range of negotiation settings in order to make sure that all relevant abilities and disabilities of the agent are captured. This is not sufficiently taken into consideration in performance evaluations, which nowadays typically cover only specific, comparatively narrow negotiation (parameter) settings.

Regarding strategy robustness, OMAC* is particularly stable in different types of negotiation games. A main reason for this lies in its adaptive and flexible style of concession making – OMAC* always attempts to minimize its concession, but adaptively relaxes its minimization effort if the opponent behaves in such a way that disagreement becomes likely. Due to its opponent learning and decision-making scheme, OMAC* achieved high scores against opponents as well as a good performance in self-play. Moreover, the experiments demonstrate that OMAC* clearly improves over its predecessor OMAC when playing against others as well as against itself.

Apart from complex negotiations studied in this paper, extensive work exists that deals with other aspects in the field of agent-based negotiations. For example, Carbonneau and Vahidov (2014) develop a model for defining and analyzing time-dependent concession behavior in general multi-issue bilateral negotiations. The model can fit the empirical data of a negotiator to its utility concession curve. Moreover, it also permits testing hypotheses about a range of negotiation behavior (e.g., slightly collaborative, neutral, quite competitive) based on the utility concession curve center. Li, Vo, Kowalczyk, Ossowski, and Kersten (2013) propose a generic framework for agent-based negotiations in open and dynamic environments. This framework enables negotiating agents to capture the social dynamics of the negotiation process through dynamically updating the resistance force and the concession force of the negotiation model. Other features of framework include that the agents are able to capture the dynamic changes of negotiation environment, e.g., the newly arrived negotiation partners and the change of their positions; agents can simultaneously engage in a few activities like searching for options outside of the counter-proposals or concurrently negotiating for a similar deal with more than one negotiation party. For solving the problem of how to evaluate negotiation in a dynamic and spatial setting, Chen et al. (2014) introduce evolutionary game theory to the evaluation of strategy performance in a spatial negotiation game, where there is a large population of agents with each being at certain location on a graph and the interaction range of agents may be restricted. In addition, in the work of Garcia and Sebastia (2014) the authors address the problem of recommendation for a group of users each of who may have different expectations about the recommendation and may act differently w.r.t other group members. To this end, a UserAgent is implemented to model a user's behavior. With the purpose of building a group profile that satisfies the particular requirements of each group member, these UserAgents on behalf of their users hold a multilateral negotiation, under a NegotiatorAgent governing the negotiation and acting as a mediator to facilitate the agreement.

9. Conclusion and future work

In this paper we introduced an advanced approach called OMAC* for effective and efficient automated negotiation in bilateral, multi-issue, time-constraint, reservation-valued scenarios without prior knowledge. OMAC*, which extends OMAC (Chen & Weiss, 2012) in several important aspects, overcomes severe limitations of previous approaches to opponent modeling (e.g., expensive computation costs for high-dimensional domains and unrealistic simplifying assumptions about the basic negotiation setting). The experimental results show a clear performance advantage of OMAC* over available state-of-the-art agents (chosen from the previous three editions of ANAC competitions) in various aspects. The experimental analysis took various key aspects of automated negotiation into account, including the level of domain competitiveness, the domain size, and discounting factors and reservation values.

The major strength of the work is the effectiveness of the proposed approach to learning unknown opponents in complex negotiations. This is achieved through the employed decomposition technique that performs trend analysis of the received utility curve and the Gaussian processes that permit accurate trend prediction and also provide a measure of confidence about the prediction. Another strength is the adaptive concession-making mechanism. On the basis of learnt opponent model and conservative aspiration level function, this mechanism suggests the desired utility at each step of the negotiation to concede towards opponents in a rational manner. Last but not the least, the work includes the extensive simulations that take a variety of performance criteria into account, using a standard and open competition infrastructure and state-of-the-art negotiating agents. The major weakness of the approach is the high computation load of the proposed approach, which results in its inability to deal with negotiation scenarios where a large number of proposal exchanges are needed in a short period.

Research contributions of the work include providing an agent-based negotiation approach that researchers in the community could employ to: (1) learn an opponent's strategy given no prior information regarding opponent privacy (e.g., strategy/preference) is available; (2) make concession in the course of a negotiation in an adaptive manner in response to uncertainty of complex negotiators; (3) propose new offers with high likelihood of being accepted by the other negotiation party. Also, our work presents a useful game-theoretic analysis based on the empirical results to investigate the robustness of the proposed negotiation approach. A practical contribution is in providing a good benchmark for measuring the efficiency of a newly proposed approach to complex negotiations.

OMAC★ opens several new research avenues, among which we consider the following as most promising. First, as preference learning is another helpful way to improve the efficiency of a negotiation, especially when the opponents are unknown, we plan to consider integrating some preference learning technique into the proposed approach for further boosting its performance. Second, another important negotiation form, which is also common in practice, is concurrent negotiation. However, this negotiation form is relatively poorly understood compared to sequential negotiation as considered in this article. We suggest to explore whether and in how far principles and mechanisms underlying OMAC★ can be successfully used and adapted to concurrent negotiation scenarios. Third, human negotiators are more flexible and less predictable than automated negotiators. Playing against human negotiators therefore pose particularly high demands on the adaptive and predictive abilities of an automated negotiator. As OMAC★ is strong in these abilities when playing against other computational agents, it appears to be a promising choice for human–machine negotiations. It would therefore be interesting to find out how well OMAC★ (equipped with an appropriate communication interface) performs when playing against different types of human negotiators. We believe that this can lead to valuable insights w.r.t. the design of automated negotiation strategies as well as the strategic behavior of human negotiators. In our current work we concentrate on the usage of preference learning techniques in the proposed approach.

Acknowledgments

We would like to thank the helpful discussions with the members of the DKE Swarmlab. Special thanks also go to the anonymous reviewers of this work for their valuable comments.

References

Abramowitz, M., & Stegun, I. (1965). *Handbook of mathematical functions*. New York, USA.

ANAC (2012). The third international automated negotiating agent competition (ANAC 2012). URL:<http://anac2012.ecs.soton.ac.uk/results/final/>.

Baarslag, T., Fujita, K., Gerding, E. H., Hindriks, K. V., Ito, T., Jennings, N. R., et al. (2013). Evaluating practical negotiating agents: Results and analysis of the 2011 international competition. *Artificial Intelligence, 198*, 73–103.

Baarslag, T., Hindriks, K. V., Jonker, C. M., Kraus, S., & Lin, R. (2012). *The first automated negotiating agents competition (ANAC 2010). Studies in computational intelligence* (Vol. 383). Springer-Verlag (pp. 113–135).

Brzostowski, J., & Kowalczyk, R. (2006). Predicting partner's behaviour in agent negotiation. In *Proceedings of the fifth international joint conference on autonomous agents and multiagent systems* (pp. 355–361). New York, NY, USA: ACM.

Carbonneau, R., Kersten, G. E., & Vahidov, R. (2008). Predicting opponent's moves in electronic negotiations using neural networks. *Expert Systems with Applications, 34*, 1266–1273.

Carbonneau, R. A., & Vahidov, R. M. (2014). A utility concession curve data fitting model for quantitative analysis of negotiation styles. *Expert Systems with Applications, 41*, 4035–4042.

Chen, S., Ammar, H. B., Tuyls, K., & Weiss, G. (2013). Using conditional restricted boltzmann machine for highly competitive negotiation tasks. In *Proceedings of the 23th international joint conference on artificial intelligence* (pp. 69–75). AAAI Press.

Chen, S., Hao, J., Weiss, G., Tuyls, K., & Leung, H. f. (2014). Evaluating practical automated negotiation based on spatial evolutionary game theory. In C. Lutz & M. Thielscher (Eds.), *KI 2014: Advances in artificial intelligence. Lecture notes in computer science* (vol. 8736, pp. 147–158). Springer International Publishing.

Chen, S., & Weiss, G. (2012). An efficient and adaptive approach to negotiation in complex environments. In *Proceedings of the 20th European conference on artificial intelligence* (pp. 228–233). Montpellier, France: IOS Press.

Chen, S., & Weiss, G. (2013). An efficient automated negotiation strategy for complex environments. *Engineering Applications of Artificial Intelligence, 26*, 2613–2623.

Chen, S., & Weiss, G. (2014). An intelligent agent for bilateral negotiation with unknown opponents in continuous-time domains. *ACM Transactions on Autonomous and Adaptive Systems, 9*, 16:1–16:24.

Coehoorn, R. M., & Jennings, N. R. (2004). Learning on opponent's preferences to make effective multi-issue negotiation trade-offs. In *Proceedings of the sixth international conference on electronic commerce* (pp. 59–68). New York, NY, USA: ACM.

Daubechies, I. (2006). *Ten lectures on wavelets. Society for industrial and applied mathematics*.

Faratin, P., Sierra, C., & Jennings, N. R. (1998). Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems, 24*, 159–182.

Faratin, P., Sierra, C., & Jennings, N. R. (2002). Using similarity criteria to make issue trade-offs in automated negotiations. *Artificial Intelligence, 142*, 205–237.

Fujita, K., Ito, T., Baarslag, T., Hindriks, K. V., Jonker, C. M., Kraus, S., et al. (2013). *The second automated negotiating agents competition (ANAC2011). Studies in computational intelligence* (Vol. 435). Berlin/Heidelberg: Springer (pp. 183–197).

Garcia, I., & Sebastia, L. (2014). A negotiation framework for heterogeneous group recommendation. *Expert Systems with Applications, 41*, 1245–1261.

Hao, J., Song, S., Leung, H. F., & Ming, Z. (2014). An efficient and robust negotiating strategy in bilateral negotiations over multiple items. *Engineering Applications of Artificial Intelligence, 34*, 45–57.

Hindriks, K., Jonker, C., Kraus, S., Lin, R., & Tykhonov, D. (2009). Genius: Negotiation environment for heterogeneous agents. In *Proceedings of the eighth international joint conference on automomous agents and multi-agent systems* (pp. 1397–1398). Budapest, Hungary: ACM.

Hindriks, K., & Tykhonov, D. (2008). Opponent modelling in automated multi-issue negotiation using Bayesian learning. In *Proceedings of the seventh international joint conference on autonomous agents and multiagent systems* (pp. 331–338). Estoril, Portugal: ACM.

Hou, C. (2004). Predicting agents tactics in automated negotiation. In *IEEE/WIC/ACM international conference on intelligent agent technology* (pp. 127–133). Los Alamitos, CA, USA: IEEE Computer Society.

Jennings, N. R., Faratin, P., Lomuscio, A. R., Parsons, S., Sierra, C., & Wooldridge, M. (2001). Automated negotiation: Prospects, methods and challenges. *International Journal of Group Decision and Negotiation, 10*, 199–215.

Jordan, P. R., Kiekintveld, C., & Wellman, M. P. (2007). Empirical game-theoretic analysis of the tac supply chain game. In *Sixth international joint conference on automomous agents and multi-agent systems* (pp. 1188–1195). Honolulu, Hawaii, USA: ACM.

Lin, R., Kraus, S., Wilkenfeld, J., & Barry, J. (2008). Negotiating with bounded rational agents in environments with incomplete information using an automated agent. *Artificial Intelligence, 172*, 823–851.

Li, M., Vo, Q. B., Kowalczyk, R., Ossowski, S., & Kersten, G. (2013). Automated negotiation in open and distributed environments. *Expert Systems with Applications, 40*, 6195–6212.

Lopes, F., Wooldridge, M., & Novais, A. (2008). Negotiation among autonomous computational agents: Principles, analysis and challenges. *Artificial Intelligence Review, 29*, 1–44.

Mor, Y., Goldman, C. V., & Rosenschein, J. S. (1996). Learn your opponent's strategy (in polynomial time)! In *Proceedings of IJCAI-95 workshop on adaptation and learning in multiagent systems* (pp. 164–176). Springer-Verlag.

Osborne, M., & Rubinstein, A. (1994). *A course in game theory*. MIT Press.

Oshrat, Y., Lin, R., & Kraus, S. (2009). Facing the challenge of human-agent negotiations via effective general opponent modeling. In: *Proceedings of the eighth international conference on autonomous agents and multiagent systems: Vol. 1. International foundation for autonomous agents and multiagent systems* (pp. 377–384).

Raiffa, H. (1982). *The art and science of negotiation*. Cambridge, MA: Harvard University Press.

Rasmussen, C., & Williams, C. (2006). *Gaussian processes for machine learning*. The MIT Press.

Rubinstein, A. (1982). Perfect equilibrium in a bargaining model. *Econometrica, 50*, 97–109.

Ruch, D., & Fleet, P. (2009). *Wavelet theory: An elementary approach with applications*. John Wiley & Sons.

Saha, S., Biswas, A., & Sen, S. (2005). Modeling opponent decision in repeated one-shot negotiations. In *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems* (pp. 397–403). New York, NY, USA: ACM.

Weiss, G. (Ed.). (2013). *Multiagent systems* (2nd ed). MIT Press.

Wellman, M. P. (2006). Methods for empirical game-theoretic analysis. In *Proceedings of the 21st national conference on artificial intelligence* (pp. 1552–1555). Boston, MA: AAAI Press.

Williams, C., Robu, V., Gerding, E., & Jennings, N. (2011). Using gaussian processes to optimise concession in complex negotiations against unknown opponents. In *Proceedings of the 22nd internatioanl joint conference on artificial intelligence* (pp. 432–438). Barcelona, Spain: AAAI Press.

Young, H. P. (1993). The evolution of conventions. *Econometrica: Journal of the Econometric Society*, 57–84.

Zeng, D., & Sycara, K. (1998). Bayesian learning in negotiation. *International Journal of Human–Computer Studies, 48*, 125–141.