A novel strategy for efficient negotiation in complex environments

Siqi Chen and Gerhard Weiss

Department of Knowledge Engineering, Maastricht University
Maastricht, The Netherlands
{siqi.chen,gerhard.weiss}@maastrichtuniversity.nl

Abstract. A complex and challenging bilateral negotiation environment for rational autonomous agents is where agents negotiate multi-issue contracts in unknown application domains against unknown opponents under real-time constraints. In this paper we present a novel negotiation strategy called **EMAR** for this kind of environment which is based on a combination of Empirical Mode Decomposition (EMD) and Autoregressive Moving Average (ARMA). **EMAR** enables a negotiating agent to adjust its target utility and concession rate adaptively in real-time according to the behavior of its opponent. The experimental results show that this new strategy outperforms the best agents from the latest Automated Negotiation Agents (ANAC) Competition in a wide range of application domains.

1 Introduction

Automated negotiation has a broad spectrum of potential applications in domains and fields such as task and service allocation, web and grid, electronic commerce and electronic markets, online information markets, and automated procurement. This potential has led to rapidly increasing research efforts on automated negotiation in recent years. The work described in this paper focuses on automated bilateral multi-issue negotiation (e.g., [16]). A key feature of this negotiation form is that two agents negotiate with the intention to agree on a profitable contract for a product or service, where the contract consists of multiple issues which are of conflictive importance for the negotiators. Examples of such issues are price and quality. More specifically, the paper concentrates on realistic scenarios for bilateral multi-issue negotiations which are particularly complex for the following four reasons. First, the negotiating agents do not know each other (i.e., they have not encountered before) and thus have no information about the preferences or strategies of their respective opponents. Second, the negotiators have no prior knowledge about the negotiation domain (e.g., about resource limitations) and thus have to cope with uncertainty about the domain. Third, we concentrate on negotiation with deadline and discount, that is, negotiation happens are under real-time constraints (the agents thus should take into consideration at each time point the remaining negotiation time) and the final utility decreases over time according to some discounting factor. And fourth, computational efficiency is important because agents may have very limited computing resources. Negotiation scenarios showing these characteristics are particularly challenging but common in reality.

This paper introduces a novel negotiation strategy called **EMAR** for those scenarios. **EMAR** integrates two key aspects of successful negotiation: efficient opponent modeling and adaptive concession making. Opponent modeling realized by **EMAR** aims at predicting the utilities of the opponent's future counter-offers through two standard mathematical techniques, namely, Empirical Mode Decomposition (EMD, e.g. [7]) and Autoregressive Moving Average (ARMA, e.g. [2]).[1] Adaptive concession making is achieved by dynamically adapting the concession rate (i.e., the degree at which an agent is willing to make concessions in its offers) on the basis of the utilities of future counter-offers which can be expected according to the acquired opponent model.

The remainder of this paper is structured as follows. Section 2 overviews important related work. Section 3 describes the standard negotiation environment used in the our research. Section 4 presents **EMAR** in detail. Section 5 offers a careful experimental analysis of **EMAR**. Section 6 identifies some important research lines induced by the described work and concludes the paper.

2 Related Work

An early influential work in the field of automated negotiation is [8]. This work raised awareness of issues related to concession making and tactical negotiation which are also relevant to the approach described here. Based on this early work and subsequent works it triggered, it had been realized that successful negotiation needs to be based in one way or another on opponent modeling. Various approaches today are available that aim at generating and utilizing opponent models in order to optimize an agent's negotiation behavior (see [11] for a useful overview). Available approaches can be classified into two groups. First, approaches that aim at learning the opponent's *preference profile*, including e.g. the opponent's reservation value (i.e., the minimum utility an agent wants to obtain) and issue/value ordering. An example of such an approach is [17], where Lin et al. use Bayesian learning to approximate the opponent preference profile; another example is [6] where kernel density estimation is used as an approximation technique. A critical drawback of preference modeling is that it tends to quickly become computationally intractable for domains having a large outcome space (especially if real-time constraints apply). Second, approaches that aim at learning the opponent's *negotiation strategy*. For instance, Saha et al. [20] make use of Chebychev polynomials to estimate the chance that the negotiation partner accepts an offer in repeated single-issue negotiations. Brzostowski et al. [3] investigate the prediction of future counter-offers online on the basis of the previous negotiation history by using differentials, thereby assuming that the opponent strategy is based on a mix of time- and behavior-dependent one.

[1] As the underlining shall indicate, the acronym **EMAR** is composed of "EM" and "AR".

Hou [13] employs non-linear regression to predict the opponent's tactic (though in single-issue negotiation), thereby supposing that the opponent uses a pure tactic as introduced in [8] and that the types of tactics are fixed. In [4] artificial neural networks (ANNs) are applied and explored in offline competition against human negotiators. Another interesting work in this area is [21]. There Williams et al. apply Gaussian processes to predict the future opponent concession before the deadline of negotiation is reached and to set the agent's "optimum" concession rate accordingly. This approach performed better than the best negotiating agents of ANAC 2010 and made the 3rd place in ANAC 2011 (ANAC is the International Automated Negotiation Agents Competition). (Our approach, **EMAR**, is experimentally evaluated against this and other agents, details are given in Section 5.) A disadvantage of available approaches exploiting learning of an opponent's negotiation strategy are the strong and often unrealistic assumptions on which they are based (as described above). In contrast, **EMAR** – which belongs to the "negotiation strategy learning" class – is designed to avoid such assumptions; in particular, it does not require any prior knowledge about the opponents and the negotiation domain.

3 Negotiation Environment

We adopt a basic bilateral multi-issue negotiation setting which is widely used in the agents field (e.g., [6, 8, 9]) and the negotiation protocol we use is based on a variant of the alternating offers protocol proposed in [18].[2] Let $I = \{a, b\}$ be a pair of negotiating agents, i represent a specific agent ($i \in I$), J be the set of issues under negotiation, and j be a particular issue ($j \in \{1, ..., n\}$ where n is the number of issues). The goal of a and b is to establish a contract for a product or service. Thereby a contract consists of a package of issues such as price, quality and quantity. Each agent has a lowest expectation for the outcome of a negotiation; this expectation is called reserved utility u_{res}. w_j^i ($j \in \{1, \ldots, n\}$) denotes the weighting preference which agent i assigns to issue j, where the weights of an agent are normalized (i.e., $\sum_{j=1}^{n}(w_j^i) = 1$ for each agent i). During negotiation agents a and b act in conflictive roles which are specified by their preference profiles. In order to reach an agreement they exchange offers O in each round to express their demands. Thereby an offer is a vector of values, with one value for each issue. The utility of an offer for agent i is obtained by the utility function defined as:

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

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

Following Rubinstein's alternating bargaining model [19], each agent makes, in turn, an offer in form of a contract proposal. Negotiation is time-limited

[2] The description of the environment in this section is taken from our previous publication on automated negotiation, see [5].

instead of being restricted by a fixed number of exchanged offers; specifically, negotiators have a shared hard deadline by when they must have completed or withdraw the negotiation. The negotiation deadline of agents is denoted by t_{max}. In this form of real-time constraints, the number of remaining rounds are not known and the outcome of a negotiation depends crucially on the time sensitivity of the agents' negotiation strategies. This holds, in particular, for discounting domains, that is, domains in which the utility is discounted with time. As usual for discounting domains, we define a so-called discounting factor δ ($\delta \in [0, 1]$) and use this factor to calculate the discounted utility as follows:

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

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

After receiving an offer from the opponent, O_{opp}, an agent decides on acceptance and rejection according to its interpretation $I(t, O_{opp})$ of the current negotiation situation. For instance, this decision can be made in dependence on a certain threshold Thres^i: agent i accepts if $U^i(O_{opp}) \geq \mathit{Thres}^i$, and rejects otherwise. As another example, the decision can be based on utility differences. Negotiation continues until one of the negotiating agents accepts or withdraws due to timeout.[3]

4 EMAR

EMAR includes two core stages – opponent modeling and adaptive concession making – as described in detail in 4.1 and 4.2, respectively. A third important stage of our strategy, its response mechanism to counter-offers, is described in 4.3. An overview of **EMAR** is given in *Algorithm 1* (the individual steps are explained in the text).

4.1 Opponent modeling

Opponent modeling realized by **EMAR** aims at predicting the future behavior of the negotiating opponents. It is mainly based on the combination of Empirical Mode Decomposition (EMD, [7, 10, 14]) and Autoregressive Moving Average (ARMA, [2]), which applies the "divide-and-conquer" principle to construct a reasonable forecasting methodology. More specifically, first EMD is employed to decompose the time series given by the utilities of past counter-offers into a finite number of components and then ARMA is applied to predict future values of these sub-components. EMD, which is based on the Hilbert-Huang transform (HHT), is a decomposition technique which relies on time-local characteristics

[3] If the agents know each other's utility functions, they can compute the Pareto-optimal contract [18]. However, in most applications a negotiator will not make this information available to its opponent.

Algorithm 1 The **EMAR** approach. 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 **EMAR**. χ is the time series comprised of the maximum utilities over intervals. ξ is the lead time for prediction and ω is the estimated central tendency of χ. E is predicted received utility series. u_{res} is the reservation utility, specifying the lowest expectation to negotiation benefit, and e_{min} is the conservative estimation of opponent concession. R is the dynamic conservative expectation function. u' is the expected utility at time t_c.

```
Require: R, δ, ξ, t_max
while t_c <= t_max do
    O_opp ⇐ receiveMessage;
    recordBids(t_c, O_opp);
    if TimeToUpdate(t_c) then
        χ ⇐ preprocessData(t_c)
        (ω, E) ⇐ getForecast(χ, ξ);
        (u_res, e_min) ⇐ updateParas(ω, χ, t_c);
        R ⇐ (u_res, e_min);
    end if
    u' = getExpUtility(t_c, E, δ, R);
    if isAcceptable(u', O_opp, t_c, δ) then
        accept(O_opp);
    else
        O_own ⇐ constructOffer(u') ;
        proposeNewBid(O_own);
    end if
end while
```

of data and can deal with nonlinear and non-stationary time series in a adaptive manner. EMD has been widely applied as a powerful data analysis tool in a broad scope of fields such as finance, image processing, ocean engineering and solar studies.

A main advantage of EMD as the decomposition method is that it is very suitable for analyzing complicated data and is fully data driven (thus requiring no additional decomposition information) – this makes EMD adaptive and very efficient. Compared to traditional Fourier and wavelet decompositions, EMD has several distinct advantages [15, 22]. First of all, fluctuations within a time series are automatically selected from the time series. Second, EMD can adaptively decompose a time series into several independent components called Intrinsic Mode Functions (IMFs). With the help of the IMFs a residue can be calculated which easily captures the main trend of the time series. Lastly, unlike wavelet decomposition, no filter base function (e.g. scaling and wavelet functions) need to be determined beforehand – which is particularly helpful when there is no prior knowledge about which filters work properly.

The IMFs satisfy the following conditions:

1. In the whole data set (time series), the number of extrema and the number of zero crossings must either equal or differ at most by one.

2. At any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

Any data series can be decomposed into IMFs according to the following sifting procedure (let $k \geq 1$, k indicates the iterative decomposition level):

1. Take signal r_{k-1} as input, with r_0 representing the original signal $\chi(t)$.
   (a) Identify all local extrema of the signal r_{k-1}.
   (b) Construct the upper envelop $Upp(r_{k-1})$ and the lower envelop $Low(r_{k-1})$ by interpolating via cubic spline the maximum and minimum values, respectively.
   (c) Approximate the local average based upon the envelop mean as $Mean(r_{k-1}) = \frac{Upp(r_{k-1})+Low(r_{k-1})}{2}$
   (d) Compute the candidate implicit mode $h_{kn} = r_{k-1} - Mean(r_{k-1})$.
   (e) If h_{kn} is an IMF, then calculate r_k as $r_k = r_{k-1} - h_{kn}$. Otherwise replace r_{k-1} with h_{kn} and repeat sifting.
2. If r_k has an implicit oscillation mode, set r_k as input signal and repeat step 1.

This sifting process serves two purposes: to eliminate riding waves and to make the wave profiles symmetric.

The decomposition procedure can be repeated on all subsequent components r_j, and the result is

$$r_0 - c_1 = r_1, r_1 - c_2 = r_2, \ldots, r_{n-1} - c_n = r_n. \quad (3)$$

This procedure terminates when (1) the latest residue r_k becomes a monotonic function (from which no more IMFs can be extracted) or (2) the IMF component c_k or the residue becomes less than the predetermined value of substantial consequence. Overall, c_1 contains the signal at a fine-grained time scale and subsequent IMFs include information at increasingly longer time periods. Eventually, the data series $\chi(t)$ can be expressed by

$$\chi(t) = \sum_{i=1}^{n} c_i + r_n \quad (4)$$

where n is the total decomposition layer (i.e., the number of IMFs), c_i is the i-th IMF component and r_n is the final residue (which represents the main trend of the data series). With that, we are able to achieve a decomposition of the data into n empirical modes and one residue. The IMFs contained in each frequency band are independent and nearly orthogonal to each other (with all having zero means) and they change with variation of the data series $\chi(t)$, whilst the residue part captures the central tendency.

The process of opponent modeling corresponds to the lines 2 to 10 in *Algorithm 1*. When receiving a new bid from the negotiation opponent at the time t_c, the agent records the time stamp t_c and the utility $U(O_{opp})$ this bid offers to it according to its utility function. The maximum utilities in consecutive equal

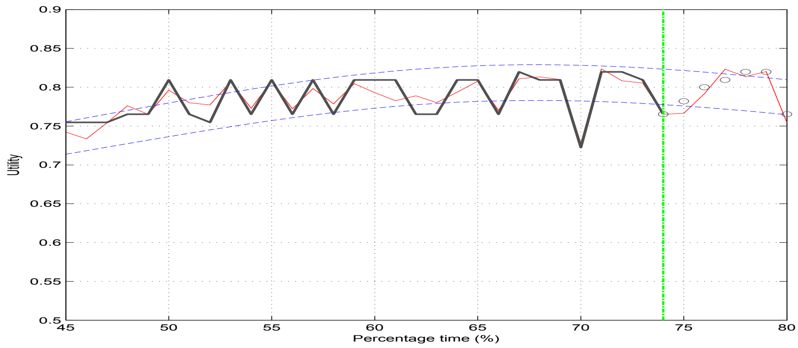


Fig. 1: Illustrating the prediction power of our model. The original time series χ, represented by the thick solid line, is received from negotiation with agent Agent_K2 in domain *Camera*. The prediction is depicted by the thin solid line, and the two dash lines show the estimated upper and lower bounds of χ. The vertical thick dash-dot line indicates the time point at which **EMAR** calculated the prediction, and the circles right to this line are the utilities actually received in the subsequent negotiation phase.

time intervals and the corresponding time stamps are used periodically as input for predicting the opponent's behavior (line 5 and 6). The reasons for *periodical* updating are similar to those mentioned in [21]. First, this reduces the computation complexity of **EMAR** so that the response speed is improved. Assume *all* observed counter-offers were taken as input, then it would be necessary deal with perhaps many thousands of data points at once. This computational load would have a clear negative impact on the quality of negotiation in a real-time setting. Second, the effect of noise can be reduced.This is important because in multi-issue negotiations a small change in utility of the opponent can result in a large utility change for the other agent – and this can easily result in a fatal misinterpretation of the opponent's behavior.

In the next stage, ARMA is used to extend all resulting components, and then ensemble them to predict opponent behaviors (shown in line 7). ARMA is a common analysis regression model which is widely used in many fields, with the formal expression as follows:

$$(1+\sum_{i=1}^{p}\phi_i L^i)X_t = (1+\sum_{i=1}^{q}\theta_i L^i)\epsilon_i \tag{5}$$

where L is the lag operator, the ϕ_i are parameters for the p-order autoregressive term, the θ are parameters for the moving average term with q order, and ϵ is a parameter capturing white noise.

Equation 5 is applied with appropriate parameters for each component extracted by EMD (i.e. c_i and the residue) for the purpose of making accurate forecasting, and then **EMAR** ensembles them to predict the future counter-offers of the opponent. Fig. 1 exemplifies this methodology, depicting the prediction power of our model with a lead time of 6 intervals. Further details of usage are given in section 4.2.

4.2 Adaptive Concession Making

EMAR adjusts the concession on the basis of the generated opponent model. Thereby a dynamic conservative expectation $R(t)$ is used to avoid "irrational concession" caused by inaccurate or over-pessimistic predictions. This makes sense in the case of negotiation opponents that are "sophisticated and tough" and always avoid making any concession in bargaining: in this case the prediction results could lead to a misleading, very low expectation about the utility offered by the opponent and this, in turn, could result in an adverse concession behavior. Furthermore, using global prediction could make this situation even worse. (This phenomenon is also considered in 5.2.)

$R(t)$ guarantees the desired minimum utility at each step, yielding values which are a lower bound of the agent's expected utilities. For the purpose of adaptation to complex negotiation sessions, $R(t)$ requires two parameters e_{min} and u_{res}. They are both periodically updated depending on the forecast of the opponent concession (line 8). e_{min} is defined as the minimum expectation of the compromise suggested by the opponent. Specifically, e_{min} is set it to the maximum value from $\psi^{low}(t)$, which is the estimated lower bound of the extended χ given by the central trend. Formally:

$$\psi^{low}(t) = \omega(t) \cdot (1 - Stdev(r_{[0,t_l]})) \tag{6}$$

where ω is the extended main tendency of χ, $r_{[0,t_l]}$ is the ratio between ω over χ within $[0,t_l]$ and $Stdev$ is the standard deviation. Having obtained ψ^{low}, e_{min} can be defined as follows:

$$e_{min} = \begin{cases} \vartheta & if \;\; \vartheta > Max(\psi^{low}) \\ Max(\psi^{low}) & otherwise \end{cases} \tag{7}$$

where $Max(x)$ gives the maximum value of input vector x. Because counter-offers with utilities indicated by ψ^{low} have already been received or can be expected during the lead time with high probability, using the maximum value assures an increase of the agent's potential profit even without significant concession.

The variable u_{res} is the reservation utility specifying the lowest expectation about the eventual benefit from a specific negotiation session. Formally this is captured by:

$$u_{res} = \begin{cases} \vartheta & if \;\; \vartheta > Max(\psi^{low}_{0,t_l}) \\ \frac{1}{2}(Max(\psi^{low}_{0,t_l}) + \vartheta) & otherwise \end{cases} \tag{8}$$

Because the final negotiation outcome (failure or agreement) is more sensitive to u_{res} than e_{min}, **EMAR** adopts a cautious and conservative way to specify it, where only ψ^{low}_{0,t_l} is considered.

Based on the above specifications, $R(t)$ is defined as follows:

$$R(t) = e_{min} + \frac{e_{min} - u_{res}}{2}(1 - t^{5\delta}) + \cos(\frac{1-\delta}{1.1}t^{\lambda})(1 - t^{1/\beta})(getMaxU(P) \cdot \delta^{\eta} - u_{res}) \tag{9}$$

where β and λ are concession factors affecting the concession rate, $getMaxU(P)$ is the function specifying the maximum utility dependent on a given preference P, δ is the discounting factor, and η is the risk factor which reflects the agent's optimal expectation about the maximum utility it can achieve. $R(t)$ can be characterized as a "dynamic conservative expectation function which carefully suggests utilities".

The subsequent process is then to decide the target utility **EMAR** expects to achieve, represented by line 11. The ensemble of all predicted components provide useful information about the opponent behavior in the lead time. This is essential because the observation of ω (and its estimated bound ψ) only gives the ambiguous area where opponent would make a compromise (rather than how the compromise might look like). Let the predicted utility series be $E(t)$, given as follows:

$$E(t) = \sum_{i=1}^{n} f_i(c_i(t), \xi) + f_{n+1}(r_n(t), \xi) \tag{10}$$

where $f_i(x)$ is the corresponding prediction model for components c_i (the IMFs) and r_n (the residue). Assume that the future expectation we have obtained from $E(t)$ is optimistic (i.e., there exists an interval $\{T | T \neq \varnothing, T \subseteq [t_c, t_s]\}$), that is,

$$E(t) \geq R(t)), \quad t \in T \tag{11}$$

where t_s is the end point of the predicated series and $t_s \leq t_{max}$. In this case the time $\hat{t}$ at which the maximal expectation $\hat{u}$ is reached is set as follows:

$$\hat{t} = argmax_{t \in T} E(t) \quad . \tag{12}$$

Moreover, in this case $\hat{u}$ is defined as

$$\hat{u} = E(\hat{t}) \quad . \tag{13}$$

On the other hand, now assume that the estimated opponent concession is below the agent's expectations (according to $R(t)$), that is, there exists no such time interval T as in the "optimistic case". In this case it is necessary to define the probability of accepting the best possible utility that can be achieved under this pessimistic expectation. This probability is given by

$$\varphi = 1 - \frac{D(R, t_{\nu}) - D(E, t_{\nu})}{\rho \cdot \sqrt{1-\delta} D(getMaxU(P)\delta^{\eta}, t_{\nu})}, \quad t_{\nu} \in [t_c, t_s] \tag{14}$$

where ρ indicates the acceptance tolerance for the pessimistic forecast and t_{ν} is given by

$$t_{\nu} = argmin_{t \in [t_c, t_s]}(|D(E, t) - D(R, t)|) \tag{15}$$

φ is compared to a random variable x with uniform distribution from the interval $[0,1]$, and the best possible outcome in the "pessimistic" scenario is chosen as the target utility if $\delta \geq x$. The rationale behind it is that if the agent rejects the "locally optimal" counter-offer (which is not too negative according to ρ), it probably loses the opportunity to reach a fairly good agreement. In the acceptance case, $\hat{u}$ and $\hat{t}$ are defined as $E(t_\nu)$ and t_ν, respectively. Otherwise, $\hat{u}$ is defined as -1, meaning it does not have an effect, and $R(t_c)$ is used to set the expected utility u'. When the agent expects to achieve a better outcome (see Equation 11), it chooses the optimal estimated utility $\hat{u}$ as its target utility (see Equations 12 and 13).

It is apparently not rational and smart to concede immediately to $\hat{u}$ when $u_l \geq \hat{u}$, and it is not appropriate for an agent to concede $\hat{u}$ without delay if $u_l < \hat{u}$ (especially because the predication may be not very accurate). To deal with this, **EMAR** simply concedes linearly. More precisely, the concession rate is dynamically adjusted in order to be able to "grasp" every chance to maximize profit. Overall, u' is calculated as follows:

$$u' = \begin{cases} R(t_c) & if \quad \hat{u} = -1 \\ \hat{u} + (u_l - \hat{u})\frac{t_c - \hat{t}}{t_l - \hat{t}} & otherwise \end{cases} \tag{16}$$

where u_l is the utility of last bid before **EMAR** performs prediction process at time t_l.

4.3 Response to counter-offers

This stage corresponds to lines 12 to 17 in Algorithm 1. When the expected utility u' has been determined, the agent needs to examine whether the utility of the counter-offer $U(O_{opp})$ is better than u' or whether it has already proposed this offer earlier in the negotiation process. If either of these two conditions is satisfied, the agent accepts this counter-offer and finishes the current negotiation session. Otherwise, the agent constructs a new offer which has an utility within some range around u'. There are two main reasons for this kind of construction. First, in multi-issue negotiations it is possible to generate a number of offers whose utilities are the same or very similar for the offering agent, but have very different utilities the opposing negotiator. (Note that in real-time constraints environment there are no limits for the number of negotiation rounds, which means that an agent in principle can construct a large amount of offers having a utility close to u' and, thus, has the opportunity to explore the utility space with the purpose of improving the acceptance chance of its offers.) Second, it is sometimes not possible to make an offer whose utility is exactly equivalent to u'. It is thus reasonable that an agent selects any offer whose utility is in the range $[(1-0.005)u', (1+0.005)u']$. If no such solution can be constructed, the agent makes its latest bid again in the next round. Moreover, with respect to negotiation efficiency, if u' drops below the value of the best counter offer in terms of its utility based on our own utility function, the agent chooses that best counter offer as its next offer. This makes much sense because this counter offer can well satisfy the expected utility of the opponent who then will be inclined to accept it.

5 Experimental Analysis

In order to evaluate the performance of **EMAR**, the General Environment for Negotiation with Intelligent multipurpose Usage Simulation (GENIUS) [12] is used as the testing platform. GENIUS is the standard platform for the annual International Automated Negotiating Agents Competition (ANAC) [1]. In this environment an agent can negotiate with other agents in a variety of domains, where every is defined by the utility function of each negotiating party. The performance of an agent (its negotiation strategy) can be evaluated via its utility achievements in negotiation tournaments which include a possibly large number of negotiation sessions for a variety of negotiation domains. Subsection 5.1 describes the overall experimental and Subsection 5.2 then presents the experimental results.

5.1 Environmental setting

EMAR is compared against the best winners (i.e., the top five agents) of ANAC2011; these are HardHeaded, Gabhoninho, IAMhaggler2011, BRAMAgent and Agent_K2 (descending order in ANAC2011). Moreover, we use five standard domains created for ANAC. All but one of them – the "*Camera*" domain – were originally used in ANAC as non-discounting domains. Three of these domains were used in ANAC2010 and two were used in ANAC2011. This choice of domains from ANAC2010+2011 makes the overall setting balanced and fair and avoids any advantageous bias for **EMAR** (note that the creators of the 2011 winners knew the ANAC2010 domains and could optimize their agents accordingly). Agent_K2, for instance, would like to accept an offer early in the domain *Energy* designed for ANAC2011, which is completely meaningless for the agent as this proposal grants the maximum profit to the opponent while no benefit is given to itself. For convenience, we refer to the non-discounting domain "*Travel*" as U_1, "*Itex vs Cypress*" as U_2, "*England vs Zimbabwe*" as U_3, "*Amsterdam party*" as U_4, and "*Camera*" as U_5. The corresponding versions with time-dependent discounting are referred to as $D_1, D_2, \ldots, D_5$, respectively. The application domains we choose cover a wide range of domain characteristics with respect to four key aspects, as overviewed in Table 1.

Domain features	U1(D1)	U2(D2)	U3(D3)	U4(D4)	U5(D5)
Domain issues	7	4	5	6	6
Domain size	188,160	180	576	3024	3600
Opposition	weak	strong	medium	medium	weak
Discounting factor	1(0.4)	1(0.5)	1(0.6)	1(0.7)	1(0.89)

Table 1: Overview of application domains

We, for each domain, ran a tournament consisting of 6 agents (i.e., the five 2011 winners and our **EMAR** agent) 10 times to get results with high statistical confidence, where each agent negotiates against all other agents in different roles. (These roles are predefined in ANAC and correspond to conflictive "buyer" and "seller" roles.) The agents do not have any information about their opponents' strategies and they are prohibited to take advantage of knowledge they might have acquired in previous negotiation sessions about the behavior of their opponents. The duration of a negotiation session is 180 seconds.

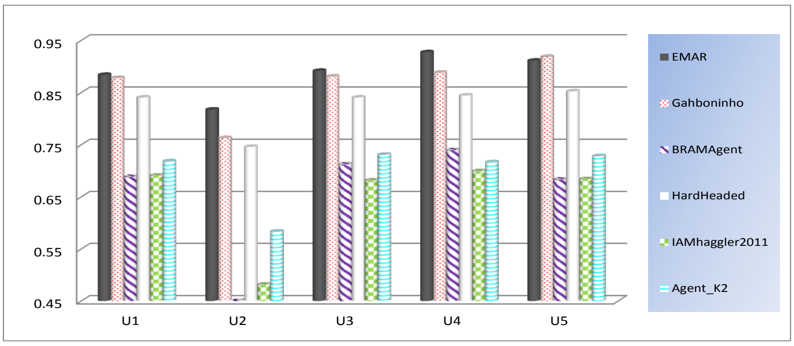
The **EMAR** agent divides the overall duration of a session into 100 consecutive intervals of 1.8 seconds each. The lead time ξ is 6, the threshold θ is 0.6, the pair concession coefficients of (β, λ) is (0.04,3) and the risk factor η is 0.2, the tolerance coefficient ρ is 0.05. These values work well in practice, but we have not intended to tweak them to stay away the issues of over-fitting and unfair competition.

5.2 Experimental results

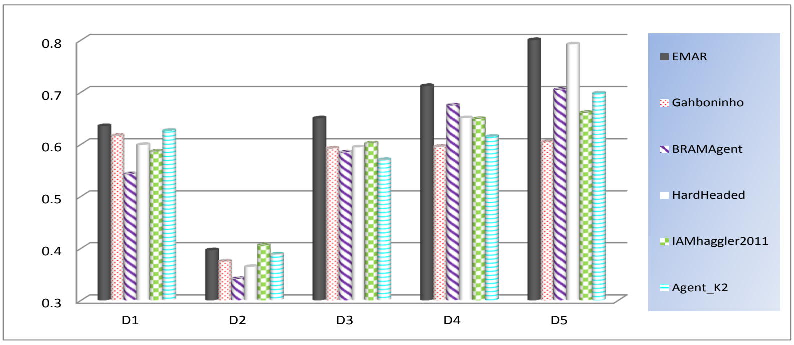
We show the experimental results achieved by each agent in terms of raw score based on non-discounting domains in Fig. 2(a), and discounting domains in Fig. 2(b). As shown in figures, the agent using **EMAR** demonstrates excellent bargaining skills. More precisely, **EMAR** wins in eight domains with 17.4% above the mean score achieved in these domains by the five competing agents. Moreover, **EMAR** made the second place for the other two domains (i.e., U5 and D2), where the performance of **EMAR** for these domains is only marginally (namely, 0.82% and 2.34%) below the score achieved by the best performer. Most notably and impressive, **EMAR** outperforms the other agents in the most competitive domain, U2, by 35.2% (compared to the mean score achieved by the other agents) and in the domain with the largest outcome space, U1, by 15.8% (compared to mean score).

Table 2 shows the overall score of all agents in terms of raw and normalized results averaged over the ten domains. Normalization is done in the standard way, using the maximum and minimum utility obtained by all agents. According to the overall performance depicted in this table, **EMAR** is the best agent, with an average normalized score of 0.772. This is 14% above the second best agent – Hardheaded –, and 30.1% above the mean score of all five opponents. Moreover, the performance of **EMAR** is very stable – compared to the other agents it shows the smallest variance values. **EMAR** is followed by HardHeaded and Gahboninho; these two agents made the first and the second place in ANAC2011. Agent_K2, which is a refined version of the champion (named Agent_K) in ANAC2010, made the fourth place. To summarize, these results show that **EMAR** is pretty efficient and significantly outperforms in a variety of negotiation scenarios the state-of-the-art automated negotiators (resp. negotiation strategies) currently available.

An interesting observation is that there is the noticeable gap between **EMAR** and IAMhaggler2011. More specifically, this agent only reaches 68.5% of the performance of **EMAR** in terms of normalized utility. As described in [21], similar



(a) Average raw scores of all agents in non-discounting domains



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

Fig. 2: Average raw scores of all agents in the ten domains. The vertical axis represents utility and horizontal axis represents domain.

to **EMAR** IAMhaggler2011 aims at predicting an opponent's future in order to be able to adjust its own behavior appropriately. Unlike **EMAR**, IAMhaggler (i) applies Gaussian process as prediction tool and (ii) adapts its concession rate on the basis of a global prediction view (i.e., on the basis of the whole preceding negotiation process). Our experimental studies suggest that a main reason for this performance gap lies in the global prediction view: this view seems to be vulnerable to "irrational concession making" induced by pessimistic predictions (see also 4.2). The phenomenon of irrational concession becomes increasingly apparent when IAMhaggler2011 bargains with "sophisticated and tough" opponents like HardHeaded, Gahboninho, and **EMAR**. For instance, when competing against these opponents in the most conflictive domain (U_2) then IAMhaggler2011 achieves only a mean utility of 0.313 while the three opponents achieve 0.903 on average. The situation furthermore is similar for other domains in our experiments.

Agent	Raw Score		Normalized Score	
	mean	variance	mean	variance
EMAR	**0.768**	**0.0011**	**0.772**	**0.0035**
HardHeaded	0.712	0.0012	0.677	0.0031
Gahboninho	0.711	0.0021	0.675	0.0058
Agent_K2	0.637	0.0022	0.559	0.0057
IAMhaggler2011	0.614	0.0026	0.528	0.0070
BRAMAgent	0.612	0.0091	0.526	0.0219

Table 2: Overall performance of every agent, represented by mean and variance.

6 Conclusion

This work introduced an effective strategy for automated bilateral negotiation in complex scenarios (multi-issue, time-constrained, unknown opponents, no prior domain knowledge, computationally feasible, etc.). The strategy, **EMAR**, outperforms the five best agents of the International Automated Negotiation Agents Competition (ANAC) 2011. We think the exceptional results justify to invest further research efforts into this approach.

Research described in this paper opens several interesting research avenues. First, is the **EMAR** strategy robust and flexible enough if the opponent has an incentive to deviate? (Independent of **EMAR**, this question has not yet been treated sufficiently in the field of automated negotiation.) An idea we pursue at the moment is to address this question with the help of empirical game analysis. Second, are there opponent modeling techniques which are even more efficient than the one used by **EMAR**? Techniques that could be considered here are e.g. GMDH networks or artificial neural networks. And, last but not least, is it possible to extend opponent modeling of **EMAR**, which focuses on modeling the opponents' negotiation strategies, toward modeling the opponents' preferences as well? We believe such an extension could lead to a significant increase in negotiation power (not only for **EMAR** but in general), though at the cost of assuming the availability of certain domain knowledge.

References

1. T. Baarslag, K. Hindriks, C. M. Jonker, S. Kraus, and R. Lin. The first automated negotiating agents competition (ANAC 2010). In *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence*. Springer-Verlag, 2010.
2. G. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control, third edition.* Prentice-Hall, 1994.
3. J. Brzostowski and R. Kowalczyk. Predicting partner's behaviour in agent negotiation. In *Proceedings of the Fifth Int. Joint Conf. on Autonomous Agents and Multiagent Systems*, AAMAS '06, pages 355–361, New York, NY, USA, 2006. ACM.

4. R. Carbonneau, G. E. Kersten, and R. Vahidov. Predicting opponent's moves in electronic negotiations using neural networks. *Expert Syst. Appl.*, 34:1266–1273, February 2008.
5. S. Chen and G. Weiss. An efficient and adaptive approach to negotiation in complex environments. In *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI2012)*, Montpellier, France, 2012. IOS Press.
6. R. M. Coehoorn and N. R. Jennings. Learning on opponent's preferences to make effective multi-issue negotiation trade-offs. In *Proceedings of the 6th Int. conf. on Electronic commerce*, ICEC '04, pages 59–68, New York, NY, USA, 2004. ACM.
7. N. E.Hunag, Z. Shen, and S. R.Long. The empirical mode decomposition and the hilbert spectrum for nonlinear and nonstationary time series analysis. *Proc. R. Soc. Lond. A*, pages 903–995, 1998.
8. P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(4):159–182, 1998.
9. P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *Artif. Intell.*, 142(2):205–237, 2002.
10. P. Flandrin, G. Rilling, P. Gonalvs, and I. E. Basics. Empirical mode decomposition as a filter bank. *IEEE Signal Proc Lett. 11 : 112-114.*, 2004.
11. M. Hendrikx. A survey of oppnent models in automated negotiation. Technical report, Delft University of Technology, The Netherlands, September 2011.
12. K. Hindriks, C. Jonker, S. Kraus, R. Lin, and D. Tykhonov. Genius: negotiation environment for heterogeneous agents. In *Proceedings of AAMAS 2009*, pages 1397–1398, 2009.
13. C. Hou. Predicting agents tactics in automated negotiation. In *Intelligent Agent Technology, IEEE / WIC / ACM International Conference on*, volume 0, pages 127–133, Los Alamitos, CA, USA, 2004. IEEE Computer Society.
14. N. E. Huang and S. S. Shen. *Hilbert-Huang transform and its applications.* World Scientific, 2005.
15. N. E. Huang, M.-L. Wu, W. Qu, S. R. Long, and S. S. Shen. Applications of hilbert-huang transform to non-stationary financial time series analysis. *Appl. Stoch. Models Bus. Ind.*, 19(3):245–268, 2003.
16. G. Lai, C. Li, K. Sycara, and J. Giampapa. Literature review on multi-attribute negotiations. Technical Report CMU-RI-TR-04-66, Robotics Institute, Pittsburgh, PA, December 2004.
17. R. Lin, S. Kraus, J. Wilkenfeld, and J. Barry. Negotiating with bounded rational agents in environments with incomplete information using an automated agent. *Artif. Intell.*, 172:823–851, April 2008.
18. H. Raiffa. *The art and science of negotiation.* Harvard University Press Cambridge, Mass, 1982.
19. A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.
20. S. Saha, A. Biswas, and S. Sen. Modeling opponent decision in repeated one-shot negotiations. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, AAMAS '05, pages 397–403, New York, NY, USA, 2005. ACM.
21. C. Williams, V. Robu, E. Gerding, and N. Jennings. Using gaussian processes to optimise concession in complex negotiations against unknown opponents. In *In Proceedings of the 22nd Internatioanl Joint Conference on Artificial Intelligence.* AAAI Press, 2011.
22. L. Yu, S. Wang, and K. K. Lai. Forecasting crude oil price with an emd-based neural network ensemble learning paradigm. *Energy Economics*, 30(5):2623 – 2635, 2008.