Acceptance Conditions in Automated Negotiation

Tim Baarslag
Man-Machine Interaction
Group
Delft University of Technology
Mekelweg 4, Delft, The
Netherlands
T.Baarslag@tudelft.nl

Koen Hindriks
Man-Machine Interaction
Group
Delft University of Technology
Mekelweg 4, Delft, The
Netherlands
K.V.Hindriks@tudelft.nl

Catholijn Jonker
Man-Machine Interaction
Group
Delft University of Technology
Mekelweg 4, Delft, The
Netherlands
C.M.Jonker@tudelft.nl

ABSTRACT

In every negotiation with a deadline, one of the negotiating parties has to accept an offer to avoid a break off. A break off is usually an undesirable outcome for both parties, therefore it is important that a negotiator employs a proficient mechanism to decide under which conditions to accept. When designing such conditions one is faced with the acceptance dilemma: accepting the current offer may be suboptimal, as better offers may still be presented. On the other hand, accepting too late may prevent an agreement from being reached, resulting in a break off with no gain for either party.

Motivated by the challenges of bilateral negotiations between automated agents and by the results and insights of the automated negotiating agents competition (ANAC), we classify and compare state-of-the-art generic acceptance conditions. We focus on *decoupled* acceptance conditions, i.e. conditions that do not depend on the bidding strategy that is used. We performed extensive experiments to compare the performance of acceptance conditions in combination with a broad range of bidding strategies and negotiation domains. Furthermore we propose new acceptance conditions and we demonstrate that they outperform the other conditions that we study. In particular, it is shown that they outperform the standard acceptance condition of comparing the current offer with the offer the agent is ready to send out. We also provide insight in to why some conditions work better than others and investigate correlations between the properties of the negotiation environment and the efficacy of acceptance conditions.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*intelligent agents, multi-agent systems*

General Terms

Algorithms, Bargaining, Experimentation, Negotiation

Keywords

Automated bilateral negotiation, acceptance criteria, acceptance conditions

1. INTRODUCTION

Negotiation is an important process to reach trade agreements, and to form alliances or resolve conflicts. The field of negotiation originates from various disciplines including artificial intelligence, economics, social science, and game theory (e.g., [2, 16, 20]). The strategic–negotiation model has a wide range of applications, such as resource and task allocation mechanisms, conflict resolution mechanisms, and decentralized information services [16].

A number of successful negotiation strategies have already been established both in literature and in implementations [6, 7, 12, 13, 19]. And more recently, in 2010 seven new negotiation strategies were created to participate in the first automated negotiating agents competition (ANAC 2010) [3] in conjunction with the Ninth International Conference on Autonomous Agents and Multiagent Systems (AAMAS-10). During post tournament analysis of the results, it became apparent that different agent implementations use various conditions to decide when to accept an offer. In every negotiation with a deadline, one of the negotiating parties has to accept an offer to avoid a break off. Therefore, it is important for every negotiator to employ a mechanism to decide under which conditions to accept. However, designing a proper acceptance condition is a difficult task: accepting too late may result in the break off of a negotiation, while accepting too early may result in suboptimal agreements.

The importance of choosing an appropriate acceptance condition is confirmed by the results of ANAC 2010 (see Table 1). Agents with simple acceptance criteria were ranked at the bottom, while the more sophisticated time- and utility-based criteria obtained a higher score. For instance, the low ranking of *Agent Smith* was due to a mistake in the implementation of the acceptance condition [27].

Despite its importance, the theory and practice of acceptance conditions has not yet received much attention. The goal of this paper is to classify current approaches and to compare acceptance conditions in an experimental setting. Thus in this paper we will concentrate on the final part of the negotiation process: the acceptation of an offer. We focus on decoupled acceptance conditions: i.e., generic acceptance conditions that can be used in conjunction with an arbitrary bidding strategy.

Our contribution is fourfold:

1. We give an overview and provide a categorization of current decoupled acceptance conditions.

2. We introduce a formal negotiation model that supports the use of arbitrary acceptance conditions.

3. We compare a selection of current generic acceptance conditions and evaluate them in an experimental setting.

Rank	Agent	Acceptance condition
1	*Agent K*	Time and utility based
2	*Yushu*	Time and utility based
3	*Nozomi*	Time and utility based
4	*IAMhaggler*	Utility based only
5	*FSEGA*	Utility based only
6	*IAMcrazyHaggler*	Utility based only
7	*Agent Smith*	Time and utility based

Table 1: An overview of the rank and acceptance conditions of every agent in ANAC 2010.

4. We propose new acceptance conditions and test them against established acceptance conditions, using varying types of bidding techniques.

The remainder of this paper is organized as follows. Section 2 defines the model of negotiation that we employ and provides an overview of current acceptance conditions. In Section 3, we also consider combinations of acceptance conditions. Section 4 discusses our experimental setup and results, which demonstrate that some combinations outperform traditional acceptance conditions. Finally, Section 5 outlines our conclusions and our plans for further research on acceptance strategies.

2. ACCEPTANCE CONDITIONS IN NEGOTIATION

This paper focuses on acceptance conditions (also called acceptance criteria) that are decoupled: i.e. generic acceptance conditions that are not tied to a specific agent implementation and hence can be used in conjunction with an arbitrary bidding strategy. We first describe a general negotiation model which fits current decoupled acceptance conditions. We have surveyed existing negotiation agents to examine the acceptance criteria that they employ. We then categorize them according to the input that they use in their decision making process.

2.1 Negotiation Model

We consider *bilateral* negotiations, i.e. a negotiation between two parties or agents A and B. The agents negotiate over *issues* that are part of a negotiation *domain*, and every issue has an associated range of alternatives or *values*. A negotiation outcome consists of a mapping of every issue to a value, and the set Ω of all possible outcomes is called the *outcome space*. The outcome space is common knowledge to the negotiating parties and stays fixed during a single negotiation session.

We further assume that both parties have certain preferences prescribed by a *preference profile* over Ω. These preferences can be modeled by means of a utility function U, which maps a possible outcome $\omega \in \Omega$ to a real-valued number in the range $[0, 1]$. In contrast to the outcome space, the preference profile of the agents is private information.

Finally, the interaction between negotiating parties is regulated by a *negotiation protocol* that defines the rules of how and when proposals can be exchanged. We use the alternating-offers protocol [23] for bilateral negotiation, in which the negotiating parties exchange offers in turns.

As in [26], we assume a common global time, represented here by $\mathcal{T} = [0, 1]$. We supplement the alternating-offers protocol with a deadline $t = 1$, at which moment both agent receive utility 0. This is the same setup as [8], with the exception that issues are not necessarily real-valued and both agents have the same deadline equal to $t = 1$. We represent by $x_{A \to B}^t$ the negotiation outcome proposed by agent A to agent B at time t. A *negotiation thread* (cf. [6, 26]) between two agents A and B at time $t \in \mathcal{T}$ is defined as a finite sequence

$$H_{A \leftrightarrow B}^t := \left(x_{p_1 \to p_2}^{t_1}, x_{p_2 \to p_3}^{t_2}, x_{p_3 \to p_4}^{t_3}, \ldots, x_{p_n \to p_{n+1}}^{t_n} \right),$$

where

1. $t_k \leq t_l$ for $k \leq l$, the offers are ordered over time $\mathcal{T}$,

2. $p_k = p_{k+2} \in \{A, B\}$ for all k, the offers are alternating between the agents,

3. All t_i represent instances of time $\mathcal{T}$, with $t_n \leq t$,

4. $x_{p_k \to p_{k+1}}^{t_k} \in \Omega$ for $k \in \{1, \ldots, n\}$, the agents exchange complete offers.

Additionally, the last element of $H_{A \leftrightarrow B}^t$ may be equal to one of the particles $\{Accept, End\}$. We will say a negotiation thread is *active* if this is not the case.

When agent A receives an offer $x_{B \to A}^t$ from agent B sent at time t, it has to decide at a later time $t' > t$ whether to accept the offer, or to send a counter-offer $x_{A \to B}^{t'}$. Given a negotiation thread $H_{A \leftrightarrow B}^t$ between agents A and B, we can formally express the action performed by A with an *action function* X_A:

$$X_A(t', x_{B \to A}^t) = \begin{cases} End & \text{if } t' \geq 1 \\ Accept & \text{if } \mathbf{AC}_A(t', x_{A \to B}^{t'}, H_{A \leftrightarrow B}^t) \\ x_{A \to B}^{t'} & \text{otherwise} \end{cases}$$

Note that we extend the setting of [8, 26] by introducing the *acceptance condition* $\mathbf{AC}_A$ of an agent A. This model enables us to study arbitrary decoupled acceptance conditions. $\mathbf{AC}_A$ that takes as input

$$\mathcal{I} = (t', x_{A \to B}^{t'}, H_{A \leftrightarrow B}^t),$$

the tuple containing the current time t', the offer $x_{A \to B}^{t'}$ that the agent considers as a bid (in line with the bidding strategy the agent uses), and the ongoing negotiation thread $H_{B \leftrightarrow A}^t$.

The resulting action given by the function $X_A(t', x_{B \to A}^t)$ is used to extend the current negotiation thread between the two agents. If the agent does not accept the current offer, and the deadline has not been reached, it will prepare a counter-offer $x_{A \to B}^{t'}$ by using a bidding strategy or *tactic* to generate new values for the negotiable issues. Tactics can take many forms, e.g. time-dependent, resource dependent, imitative, and so on [26]. In our setup we will consider the tactics as given and try to optimize the accompanying acceptance conditions.

2.2 Acceptance Criteria

Let an active negotiation thread

$$H_{A \leftrightarrow B}^t = \left(x_{p_1 \to p_2}^{t_1}, x_{p_2 \to p_3}^{t_2}, \ldots, x_{A \to B}^{t_{n-1}}, x_{B \to A}^{t_n} \right),$$

be given at time $t' > t = t_n$, so that it is agent A's turn to perform an action.

As outlined in our negotiation model, the action function X_A of an agent A uses an acceptance condition $\mathbf{AC}_A(\mathcal{I})$ to

decide whether to accept. In practice, most agents do not use the full negotiation thread to decide whether it is time to accept. For instance many agent implementations, such as [9, 8, 26], use the following implementation of $\mathbf{AC}_A(\mathcal{I})$:

$$\mathbf{AC}_A(t', x_{A\to B}^{t'}, H_{A\leftrightarrow B}^t) \iff U_A(x_{B\to A}^t) \geq U_A(x_{A\to B}^{t'}).$$

That is, A will accept when the utility U_A for the opponent's last offer at time t is greater than the value of the offer agent A is ready to send out at time t'. The acceptance condition above depends on the agent's upcoming offer $x_{A\to B}^{t'}$. For $\alpha, \beta \in \mathbb{R}$ this may be generalized as follows:

$$\mathbf{AC}_{\text{next}}^{\mathcal{I}}(\alpha, \beta) \stackrel{\text{def}}{\iff} \alpha \cdot U_A(x_{B\to A}^t) + \beta \geq U_A(x_{A\to B}^{t'}).$$

We can view α as the scale factor by which we multiply the opponent's bid, while β specifies the minimal 'utility gap' [13] that is sufficient to accept.

Analogously, we have acceptance conditions that rely on the agent's *previous* offer $x_{A\to B}^{t_{n-1}}$:

$$\mathbf{AC}_{\text{prev}}^{\mathcal{I}}(\alpha, \beta) \stackrel{\text{def}}{\iff} \alpha \cdot U_A(x_{B\to A}^t) + \beta \geq U_A(x_{A\to B}^{t_{n-1}}).$$

Note that this acceptance condition does not take into account the time that is left in the negotiation, nor any offers made previous to time t. Other acceptance conditions may rely on other measures, such as the remaining negotiation time or the utility of our previous offer. For example, there is a very simple acceptance criterion that only compares the opponent's offer with a constant α:

$$\mathbf{AC}_{\text{const}}^{\mathcal{I}}(\alpha) \stackrel{\text{def}}{\iff} U_A(x_{B\to A}^t) \geq \alpha.$$

Last but not least, instead of considering utility agents may employ a time-based condition to accept after a certain amount of time $T \in \mathcal{T}$ has passed:

$$\mathbf{AC}_{\text{time}}^{\mathcal{I}}(T) \stackrel{\text{def}}{\iff} t' \geq T.$$

We will omit the superscript $\mathcal{I}$ when it is clear from the context. We will use these general acceptance conditions to classify existing acceptance mechanisms in the next section.

2.3 Existing Acceptance Conditions

We give a short overview of decoupled acceptance conditions used in literature and current agent implementations. We are primarily interested in acceptance conditions that are not specifically designed for a single agent. We do not claim the list below is complete; however it serves as a good starting point to categorize current decoupled acceptance conditions. We surveyed the entire pool of agents of ANAC 2010, including *Agent K* and *Nozomi* [25], *Yushu* [1], *IAM(crazy)Haggler* [5], *FSEGA* [24] and *Agent Smith* [27]. We also examined well-known agents from literature, such as the Trade-off agent [7], the Bayesian learning agent [11], *ABMP* [13], equilibrium strategies of [9], and time dependent negotiation strategies as defined in [22], i.e. the Boulware and Conceder tactics.

Listed in Table 2 is a selection of generic acceptance conditions found.

Some agents also use logical combinations of different acceptance conditions at the same time. This explains why some agents are listed multiple times. For example, both *IAMHaggler* and *IAMcrazyHaggler* [4] accept precisely when

$$\mathbf{AC}_{\text{const}}(0.88) \vee \mathbf{AC}_{\text{next}}(1.02, 0) \vee \mathbf{AC}_{\text{prev}}(1.02, 0).$$

Table 2: A selection of existing decoupled acceptance conditions.

AC	α	β	Agent
$\mathbf{AC}_{\text{prev}}(\alpha, \beta)$	1.03	0	*FSEGA, Bayesian Agent*
	1	0	*Agent Smith*
	1.02	0	*IAM(crazy)Haggler*
	1	0.02	*ABMP*
$\mathbf{AC}_{\text{next}}(\alpha, \beta)$	1	0	*FSEGA,* Boulware, Conceder, Trade-off, Equilibrium strategies
	1.02	0	*IAM(crazy)Haggler*
	1.03	0	*Bayesian Agent*
$\mathbf{AC}_{\text{const}}(\alpha)$	1	-	*FSEGA*
	0.9	-	*Agent Smith*
	0.88	-	*IAM(crazy)Haggler*
	T		
$\mathbf{AC}_{\text{time}}(T)$	0.92	-	*Agent Smith*

We will not focus on the many possible combinations of all acceptance conditions that may thus be obtained; we will study the basic acceptance conditions in isolation with varying parameters. However in addition to this we study a small selection of combinations in Section 3. We leave further combinations for future research.

As can be seen from Table 2, in our sample the most commonly used acceptance condition is $\mathbf{AC}_{\text{next}} = \mathbf{AC}_{\text{next}}(1, 0)$, which is the familiar condition of accepting when the opponent's last offer is better than the planned offer of the agent. The function $\beta \mapsto \mathbf{AC}_{\text{prev}}(1, \beta)$ can be viewed as an acceptance condition that accepts when the utility gap [13] between the parties is smaller than β. We denote this condition by $\mathbf{AC}_{\text{gap}}(\beta)$.

3. COMBINED ACCEPTANCE CONDITIONS

We define three acceptance conditions that are designed to perform well in conjunction with an arbitrary bidding strategy. This will incorporate all ideas behind the traditional acceptance conditions we have described so far. We will show in Section 4 that they work better than the majority of simple generic conditions listed in Table 2.

From a negotiation point of view, it makes sense to alter the behavior of the acceptance condition when time is running short. Many ANAC agents such as Yushu, Nozomi and FSEGA [1, 24, 25] split the negotiation into different intervals of time and apply different sub-strategies to each interval.

The basic idea behind combined acceptance conditions $\mathbf{AC}_{\text{combi}}$ is as follows. In case the bidding strategy plans to propose a deal that is worse than the opponent's offer, we have reached a consensus with our opponent and we accept the offer. But if there still exists a gap between our offer and time is short, the acceptance condition should wait for an offer that is not expected to improve in the remaining time. Thus $\mathbf{AC}_{\text{combi}}$ is designed to be a proper extension of $\mathbf{AC}_{\text{next}}$, with adaptive behavior based on recent bidding behavior near the deadline.

To define $\mathbf{AC}_{\mathrm{combi}}$, suppose an active negotiation thread

$$H_{A \leftrightarrow B}^t = \left(x_{p_1 \to p_2}^{t_1}, x_{p_2 \to p_3}^{t_2}, \ldots, x_{A \to B}^{t_{n-1}}, x_{B \to A}^{t_n} \right),$$

is given at time $t' > t = t_n > \frac{1}{2}$ near the deadline, when it is agent A's turn. Note that there is $r = 1 - t'$ time remaining in the negotiation, which we will call the *remaining time window*. A good sample of what might be expected in the remaining time window consists of the bids that were exchanged during the previous time window $W = [t' - r, t'] \subseteq \mathcal{T}$ of the same size.

Let

$$H_{B \to A}^W = \left\{ x_{B \to A}^s \in H_{A \leftrightarrow B}^t \mid s \in W \right\}$$

denote all bids offered by B to A in time window W. We can now formulate the average and maximum utility that was offered during the previous time window in the negotiation thread $H = H_{B \to A}^W$:

$$\mathrm{MAX}^W = \max_{x \in H} U_A(x).$$

and

$$\mathrm{AVG}^W = \frac{1}{|H|} \sum_{x \in H} U_A(x).$$

We let $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$ accept at time t' exactly when the following holds: $\mathbf{AC}_{\mathrm{next}}$ indicates that we have to accept, *or* we have almost reached the deadline ($t' \geq T$) and the current offer suffices (i.e. better than α) given the remaining time:

$$\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$$
$$\stackrel{\mathrm{def}}{\Longleftrightarrow}$$
$$\mathbf{AC}_{\mathrm{next}} \vee \mathbf{AC}_{\mathrm{time}}(T) \wedge \left(U_A(x_{B \to A}^t) \geq \alpha \right).$$

Note that we have defined $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$ in such a way that it splits the negotiation time into two phases: $[0, T)$ and $[T, 1]$, with different behavior in both cases.

We will consider three different combined acceptance conditions:

1. $\mathbf{AC}_{\mathrm{combi}}(T, \mathrm{MAX}^W)$, the current offer is good enough when it is better than all offers seen in the previous time window W,

2. $\mathbf{AC}_{\mathrm{combi}}(T, \mathrm{AVG}^W)$, the offer is better than the average utility of offers during the previous time window W,

3. $\mathbf{AC}_{\mathrm{combi}}(T, \mathrm{MAX}^{\mathcal{T}})$, the offer should be better than any bid seen before.

4. EXPERIMENTS

In order to experimentally test the efficacy of an acceptance condition, we considered a negotiation setup with the following characteristics. We equipped a set of agents (as defined later) with an acceptance condition, and measured its result against other agents in the following way. Suppose agent A is equipped with acceptance condition $\mathbf{AC}_A$ and negotiates with agent B. The two parties may reach a certain outcome $\omega \in \Omega$, for which A receives the associated utility $U_A(\omega)$. The score for A is averaged over all trials on various domains (see Section 4.1.2), alternating between the two preference profiles defined on that domain. E.g., on the negotiation scenario between England and Zimbabwe, A will play both as England and as Zimbabwe against all others.

For our experimental setup we employed GENIUS (General Environment for Negotiation with Intelligent multi-purpose Usage Simulation) [17]. This environment, which is also used in ANAC, helps to facilitate the design and evaluation of automated negotiators' strategies. It can be used to simulate tournaments between negotiating agents in various negotiation scenarios, such as the setup described in this section. It supports the alternating offer protocol with a real-time deadline as outlined in our negotiation model. The default negotiation time in GENIUS and in the setup of ANAC is 3 minutes per negotiation session; therefore we use the same value in our experiments.

4.1 Detailed Experimental Setup

4.1.1 Agents

We use the negotiation tactics that were submitted to The Automated Negotiating Agents Competition (ANAC 2010) [3]. ANAC is a negotiation competition aiming to facilitate and coordinate the research into proficient negotiation strategies for bilateral multi-issue negotiation, similar to what the Trading Agent Competition (TAC) has achieved for the trading agent problem [28].

The seven agents that participated in ANAC 2010 have been implemented by various international research groups of negotiation experts. We used these strategies in our experiments as they are representative of the current state-of-the-art in automated negotiation. Firstly, we removed the built-in acceptance mechanism from this representative group of agents; this left us with its pure bidding tactics. As outlined in our negotiation model, this procedure allowed us to test arbitrary acceptance conditions in tandem with any ANAC tactic.

We aimed to tune our acceptance conditions to the top performing ANAC 2010 agents. Therefore we have selected the top 3 of ANAC agents that were submitted by different research groups, namely *Agent K*, *Yushu* and *IAMhaggler* (we omitted *Nozomi* as the designing group also implemented *Agent K*, cf. Table 1). For the set of opponents, we selected all agents from ANAC 2010, for the acceptance conditions should be tested against a wide array of strategies. The opponents also had their built-in acceptance conditions removed (and hence were not able to accept), so that differences in results would depend entirely on the acceptance condition under consideration. To test the efficacy of an acceptance condition, we equipped the top 3 tactics with this condition and compared the average utility obtained by the three agents when negotiating against their opponents.

4.1.2 Domains

The specifics of a negotiation domain can be of great influence on the negotiation outcome [10]. Acceptance conditions have to be assessed on negotiation domains of different size and complexity. Negotiation results also depend on the *opposition* of the parties' preferences. The notion of weak and strong opposition can be formally defined [14]. Strong opposition is typical of competitive domains, when a gain for one party can be achieved only at a loss for the other party. Conversely, weak opposition means that both parties achieve either losses or gains simultaneously.

With this in mind, we aimed for two domains (with two preference profiles each) with a good spread of negotiation characteristics. We picked two domains from the three that

	Itex–Cyp	Zim–Eng
Size	180	576
Opposition	Strong	Medium

Table 3: The four preference profiles used in experiments.

were used in ANAC 2010 (cf. [3]). Some agents participating in ANAC 2010 did not scale well and could not deal with a large bid space. We omitted the Travel domain as the agents had too many difficulties with it to make it a reliable testing domain.

Our first scenario is taken from [15], which describes a buyer–seller business negotiation. It involves representatives of two companies: Itex Manufacturing, a producer of bicycle components and Cypress Cycles, a builder of bicycles. There are four issues that both sides have to discuss: the price of the components, delivery times, payment arrangements and terms for the return of possibly defective parts. The opposition between the parties is strong in this domain, as the manufacturer and consumer have naturally opposing requirements. Altogether, there are 180 potential offers that contain all combinations of values for the four issues.

The second domain taken from [17, 18] involves a case where England and Zimbabwe negotiate an agreement on tobacco control. The leaders of both countries must reach an agreement on five issues. England and Zimbabwe have contradictory preferences for the first two issues, but the other issues have options that are jointly preferred by both sides. The domain has a total of 576 possible agreements.

To compensate for any utility differences in the preference profiles, the agents play both sides of every scenario.

4.1.3 Acceptance Conditions

For each acceptance condition we tested all $3 \times 7 = 21$ pairings of agents, playing with each of the 4 different preference profiles. We ran every experiment twice, so that altogether each acceptance condition was tested 168 times. We selected the following acceptance conditions for experimental testing. The different values of parameters will be discussed in the section below.

- $\mathbf{AC}_{\text{next}}(\alpha, 0)$ and $\mathbf{AC}_{\text{prev}}(\alpha, 0)$ for $\alpha \in \{1, 1.02\}$,

- $\mathbf{AC}_{\text{gap}}(\alpha)$ for $\alpha \in \{0.02, 0.05, 0.1, 0.2\}$,

- $\mathbf{AC}_{\text{const}}(\alpha)$ for $\alpha \in \{0.8, 0.9\}$,

- $\mathbf{AC}_{\text{time}}(T)$, $\mathbf{AC}_{\text{combi}}(T, \text{MAX}^W)$, $\mathbf{AC}_{\text{combi}}(T, \text{AVG}^W)$ and $\mathbf{AC}_{\text{combi}}(T, \text{MAX}^T)$, where W is the previous time window with respect to the current time t', and $T = 0.99$ (this particular value of T is discussed below).

Additionally, we ran the experiments with agents having their built-in acceptance mechanism in place. That is, we also tested the original agents' *coupled* acceptance mechanism. As we cannot for example, equip *Agent K* with the coupled acceptance condition of *Yushu*, we tested the built-in mechanism by having each agent employ its own mechanism.

4.2 Hypotheses and Experimental Results

The experiments considered here are designed to discuss the main properties and drawbacks of the acceptance conditions listed above. We formulate several hypotheses with respect to the acceptance conditions we have discussed.

Our hypothesis about $\mathbf{AC}_{\text{const}}(\alpha)$ is the following:

Hypothesis 1. For α close to one, $\mathbf{AC}_{\text{const}}(\alpha)$ performs worse than all other conditions.

To evaluate this hypothesis and others below, we have carried out a large number of experiments. The results are summarized in Table 4. The table shows the average utility obtained by the agents when equipped with several acceptance conditions. The "average utility of agreements" column represents the average utility obtained by the agent given the fact that they have reached an agreement. When they do not reach an agreement (due to the deadline), they get zero utility. Thus the following holds:

(The acceptance dilemma)

$$\text{Total average utility} = \begin{array}{c} \text{Agreement percentage} \\ \times \\ \text{Average utility of agreements.} \end{array}$$

This formula captures the essence of the acceptance dilemma: accepting bad to mediocre offers yields more agreements of relatively low utility. While accepting only the best offers produces less agreements, but of higher utility.

Now consider $\mathbf{AC}_{\text{const}}(0.9)$ and $\mathbf{AC}_{\text{const}}(0.8)$. When it reaches an agreement, it receives a very high utility (at least 0.9 or 0.8 respectively), but this happens so infrequently (resp. 26% and 38% of all negotiations), that it is ranked at the bottom when we consider total average utility.

We can conclude that our hypothesis is confirmed: in isolation, $\mathbf{AC}_{\text{const}}(\alpha)$ is not very advantageous to use. The main reason is that the choice of the constant α is highly domain-dependent. A very cooperative domain may have multiple win–win outcomes with utilities above α. $\mathbf{AC}_{\text{const}}(\alpha)$ would then accept an offer which is *relatively* bad, i.e. it could have done much better. On the other hand, in highly competitive domains, it may simply 'ask for too much' and may rarely obtain an agreement. Its value lies mostly in using it in combination with other acceptance conditions such as $\mathbf{AC}_{\text{next}}$. It can then benefit the agent by accepting an unexpectedly good offer or a mistake by the opponent.

As we discussed earlier in Section 2.3, the acceptance conditions $\mathbf{AC}_{\text{prev}}(\alpha, 0)$ and $\mathbf{AC}_{\text{next}}(\alpha, 0)$ are standard in literature for $\alpha \in \{1, 1.02\}$. Many agents tend to use these acceptance conditions, as they are well-known and easy to implement. We have formed the following hypothesis:

Hypothesis 2. $\mathbf{AC}_{\text{next}}(\alpha, 0)$ will outperform $\mathbf{AC}_{\text{prev}}(\alpha, 0)$ for $\alpha \in \{1, 1.02\}$. However, both conditions will perform worse than conditions that take the remaining time into account.

To test this hypothesis, we consult Table 4 where we have considered the two values for α. The first observation is that $\mathbf{AC}_{\text{prev}}(\alpha, 0)$ and $\mathbf{AC}_{\text{next}}(\alpha, 0)$ already perform much better than $\mathbf{AC}_{\text{const}}$. The higher value for α yields a better result and $\mathbf{AC}_{\text{next}}(\alpha, 0)$ does indeed outperform $\mathbf{AC}_{\text{prev}}(\alpha, 0)$. It makes sense that comparing the opponent's offer to our up-

Table 4: Utility scores of agents equipped with an acceptance condition

Acceptance Condition	Agent K	IAMhaggler	Yushu	Agreement %	Average utility of agreements	Total avg
$\mathbf{AC}_{\mathrm{combi}}(\mathrm{MAX}^W)$	0.691	0.639	0.695	99%	0.679	0.675
$\mathbf{AC}_{\mathrm{combi}}(\mathrm{AVG}^W)$	0.684	0.634	0.691	99%	0.678	0.670
$\mathbf{AC}_{\mathrm{gap}}(0.1)$	0.636	0.562	0.693	83%	0.761	0.630
Built-in mechanism	0.641	0.547	0.692	82%	0.768	0.627
$\mathbf{AC}_{\mathrm{combi}}(\mathrm{MAX}^{\mathcal{T}})$	0.691	0.577	0.596	89%	0.696	0.621
$\mathbf{AC}_{\mathrm{time}}(0.99)$	0.612	0.580	0.663	99%	0.622	0.618
$\mathbf{AC}_{\mathrm{gap}}(0.2)$	0.626	0.579	0.650	86%	0.721	0.618
$\mathbf{AC}_{\mathrm{gap}}(0.05)$	0.629	0.550	0.672	78%	0.791	0.617
$\mathbf{AC}_{\mathrm{next}}(1.02, 0)$	0.616	0.517	0.696	77%	0.788	0.610
$\mathbf{AC}_{\mathrm{gap}}(0.02)$	0.618	0.491	0.638	73%	0.802	0.582
$\mathbf{AC}_{\mathrm{prev}}(1.02, 0)$	0.618	0.491	0.629	72%	0.805	0.579
$\mathbf{AC}_{\mathrm{next}}(1, 0)$	0.586	0.517	0.597	72%	0.787	0.567
$\mathbf{AC}_{\mathrm{prev}}(1, 0)$	0.588	0.491	0.589	69%	0.805	0.556
$\mathbf{AC}_{\mathrm{const}}(0.8)$	0.286	0.374	0.313	38%	0.851	0.324
$\mathbf{AC}_{\mathrm{const}}(0.9)$	0.215	0.272	0.231	26%	0.935	0.239

coming offer is more beneficial than comparing it to our previous offer, as $\mathbf{AC}_{\mathrm{next}}$ is always 'one step ahead' of $\mathbf{AC}_{\mathrm{prev}}$. However, all time-dependent acceptance conditions outperform both of them, even for $\alpha = 1.02$. This also settles the second part of the hypothesis. The reason for this bad performance is that many bidding strategies focus on the 'negotiation dance' [21]. That is, modeling the opponent, trying to make equal concessions and so on. When a strategy does not explicitly take time considerations into account when making an offer, this poses a problem for the two standard acceptance conditions: they rely completely on the bidding strategy to concede to the opponent before the deadline occurs. When the agent or the opponent does not concede enough near the deadline, the standard conditions lead to poor performance.

Our third hypothesis with respect to the time-dependent condition is as follows:

Hypothesis 3. $\mathbf{AC}_{\mathrm{time}}(T)$ always reaches an agreement, but of relatively low utility.

To evaluate this hypothesis we needed to provide a concrete value for the experimental variable T. We have set $T = 0.99$ for every acceptance condition depending on T. As we have found during preliminary experiments, this value is sufficiently close to the deadline, while it still allows enough time to reach a win-win outcome. From observing the acceptance probability of $\mathbf{AC}_{\mathrm{time}}(0.99)$ in the experimental results, we see that in 1 out of 168 negotiations ($\approx 1\%$) this criterion did not reach an agreement due to agent crashes and protocol errors, in which case both agents received utility zero. But except for these particular events, $\mathbf{AC}_{\mathrm{time}}(T)$ will always reach an agreement, therefore we consider this part of the hypothesis confirmed.

$\mathbf{AC}_{\mathrm{time}}(T)$, with T close to 1 is a sensible criterion to avoid a break off at all cost. It is rational to prefer any outcome over a break off of zero utility. However, the resulting deal can be anything. As we can see from the table, this is the reverse situation of $\mathbf{AC}_{\mathrm{gap}}$: $\mathbf{AC}_{\mathrm{time}}(T)$ yields the lowest agreement score (0.622) of all conditions. This can be explained by the acceptance dilemma: by accepting any offer near the deadline, it reaches more agreements but of relatively low utility. Still the overall score is almost the same (0.618) and thus reasonable. It is interesting to note that $\mathbf{AC}_{\mathrm{time}}(T)$ outperforms both $\mathbf{AC}_{\mathrm{prev}}$ and $\mathbf{AC}_{\mathrm{next}}$ in average overall score.

This insight led us to believe that more consideration has to be given to the remaining time when deciding to accept an offer. The combined acceptance conditions evaluated in the next chapter expand upon this idea to get better deals near the deadline.

4.2.1 Evaluating $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$

When evaluating $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$, we expect the following characteristics. $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$ is an extension of $\mathbf{AC}_{\mathrm{next}}$ in the sense that it will accept under broader circumstances. It alleviates some of the mentioned drawbacks of $\mathbf{AC}_{\mathrm{next}}$ by also accepting when the utility gap between the parties is positive. Also note, that in addition to the parameters that current acceptance conditions use, such as my previous bid $x_{A \to B}^{t_{n-1}}$, my next bid $x_{A \to B}^{t'}$, the remaining time, and the opponent's bid $x_{B \to A}^t$, this condition employs the entire bidding history $H_{A \leftrightarrow B}^t$ to compute the acceptability of an offer. Therefore we expect better results than with $\mathbf{AC}_{\mathrm{next}}$, with more agreements, and when it agrees, we expect a better deal than by using $\mathbf{AC}_{\mathrm{time}}(T)$.

We capture this last statement in our final hypothesis:

Hypothesis 4. The combination $\mathbf{AC}_{\mathrm{combi}}(T, \alpha)$ outperform other acceptance conditions, such as $\mathbf{AC}_{\mathrm{time}}(T)$ and $\mathbf{AC}_{\mathrm{next}}$ primarily by getting deals of higher utility.

As is evident from the experimental results, $\mathbf{AC}_{\mathrm{combi}}(\mathrm{MAX}^W)$ and $\mathbf{AC}_{\mathrm{combi}}(\mathrm{AVG}^W)$ dominate the other acceptance conditions. They even perform 7% better than the built-in mechanisms of the agent, and 18% better than $\mathbf{AC}_{\mathrm{next}}$. Similar to $\mathbf{AC}_{\mathrm{time}}$, both conditions still get a deal almost every time, but with a higher utility. However, the average utility of an agreement is not the highest: the $\mathbf{AC}_{\mathrm{gap}}$ conditions and the built-in mechanisms get better agreements. But again, we can observe that their agreement rate is also lower, resulting in a higher overall score for the combined criteria.

Aiming for the highest utility that has been offered so far (i.e. $\mathbf{AC}_{\mathrm{combi}}(\mathrm{MAX}^{\mathcal{T}})$) is a less successful criterion, mostly due to a big decrease in agreements. The higher utility that is obtained with this condition does not compensate for the loss of utility that is caused by a break off.

4.3 Related Work

All existing negotiation agent implementations deal with the problem of when to accept. In many cases, the agent accepts a proposal when the value of the offered contract is higher than the offer it is ready to send out at that moment in time. Examples include the time dependent negotiation strategies defined in [22] (e.g. the Boulware and Conceder tactics). The same principle is used in the equilibrium strategies of [9] and for the Trade-off agent [7]. In the setting of [7] however, the deadline can be different for both agents. In this paper, we consider strategies that do not always reach an agreement, and hence we have concentrated on acceptance conditions that yield better results in such cases.

Of all ANAC 2010 participants, we shortly discuss *Agent K* [25] as it employs the most sophisticated method to decide when to accept. Its acceptance mechanism is based on the mean and variance of all received offers. It then tries to determine the best offer it might receive in the future and sets its proposal target accordingly. In contrast to our approach, this mechanism is not fully decoupled from the bidding strategy as it directly influences its bid target. Furthermore, it does not restrict its scope to the remaining or previous time window. Finally, we note that *Agent K* performs better in our experimental setup (cf. Table 4) when equipped with our combined acceptance conditions than with its built-in mechanism.

Although we do not focus on negotiation tactics and convergence results, our negotiation model builds upon the model of [26]. However, in this model, the action function of an agent only takes into account the offer it is ready to send out at that moment in time. Moreover, the focus of the paper is not on comparing acceptance conditions as only one specific instance is studied. We take a more general approach in which the agent utilizes a generic acceptance mechanism, in which the current time and the entire bidding history is considered.

5. CONCLUSION AND FUTURE WORK

In this paper, we aimed to classify current approaches to generic acceptance conditions and to compare a selection of acceptance conditions in an experimental setting. We presented the challenges and proposed new solutions for accepting offers in current state-of-the-art automated negotiations. The focus of this paper is on decoupled acceptance conditions, i.e. general conditions that do not depend on a particular bidding strategy.

Designing an effective acceptance condition is challenging because of the acceptance dilemma: better offers may arrive in the future, but waiting for too long can result in a break off of the negotiation, which is undesirable for both parties.

We have seen that the standard acceptance criterion $\mathbf{AC}_{\mathrm{next}}$ is often used by negotiating agents. From our results, it is apparent that $\mathbf{AC}_{\mathrm{next}}$ does not always yield optimal agreements. We established that it performs worse than more sophisticated acceptance conditions.

In addition to classifying and comparing existing acceptance conditions, we have devised three new acceptance conditions by combining existing ones. This included two acceptance conditions that estimate whether a better offer might occur in the future based on recent bidding behavior. These conditions obtained the highest utility in our experiments and hence performed better than the other conditions we have investigated.

A suggestion for future research would be to explore the many possible combinations of acceptance conditions that may be obtained using conjunction and disjunction (and possibly negation). Some agents already use a logical combination of different acceptance conditions at the same time. For example, the *IAM(crazy)Haggler* agents accept when

$$\mathbf{AC}_{\mathrm{const}}(0.88) \vee \mathbf{AC}_{\mathrm{next}}(1.02, 0) \vee \mathbf{AC}_{\mathrm{prev}}(1.02, 0).$$

A suitable combination of acceptance conditions could provide a considerable improvement over current acceptance conditions.

Secondly, we plan to test acceptance conditions with more agents and on larger domains, using the resources that will be available after the upcoming ANAC 2011 event.

Finally, we did not consider negotiation domains with discount factors, which devaluate utility with the passing of time. Adding discount factors will require new acceptance conditions that give more consideration to the negotiation timeline. We plan to examine such extensions in future work.

Acknowledgements

6. REFERENCES

[1] Bo An and Victor Lesser. Yushu: a heuristic-based agent for automated negotiating competition. In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear)*. Springer-Verlag, 2010.

[2] R.J. Aumann and S. Hart, editors. *Handbook of Game Theory with Economic Applications*, volume 1. Elsevier, March 1992.

[3] Tim Baarslag, Koen Hindriks, Catholijn M. Jonker, Sarit Kraus, and Raz Lin. The first automated negotiating agents competition (ANAC 2010). In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in*

Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear). Springer-Verlag, 2010.

[4] Enrico H. Gerding Colin R. Williams, Valentin Robu and Nicholas R. Jennings. Iamhaggler: A negotiation agent for complex environments. *This volume.*

[5] Enrico H. Gerding Colin R. Williams, Valentin Robu and Nicholas R. Jennings. Iamhaggler: A negotiation agent for complex environments. In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear).* Springer-Verlag, 2010.

[6] P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Int. Journal of Robotics and Autonomous Systems,* 24(3-4):159–182, 1998.

[7] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make negotiation trade-offs. *Journal of Artificial Intelligence,* 142(2):205–237, 2003.

[8] S. S. Fatima, M. Wooldridge, and N. R. Jennings. Optimal negotiation strategies for agents with incomplete information. *Intelligent Agents Viii: Agent Theories, Architectures, and Languages,* 2333:377–392, 2002.

[9] Shaheen S. Fatima, Michael Wooldridge, and Nicholas R. Jennings. Multi-issue negotiation under time constraints. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems,* pages 143–150, New York, NY, USA, 2002. ACM.

[10] Koen Hindriks and Dmytro Tykhonov. Towards a quality assessment method for learning preference profiles in negotiation, 2008.

[11] Koen V. Hindriks and Dmytro Tykhonov. Opponent modelling in automated multi-issue negotiation using bayesian learning, 2008.

[12] Takayuki Ito, Hiromitsu Hattori, and Mark Klein. Multi-issue negotiation protocol for agents : Exploring nonlinear utility spaces, 2007.

[13] Catholijn Jonker, Valentin Robu, and Jan Treur. An agent architecture for multi-attribute negotiation using incomplete preference information. *Autonomous Agents and Multi-Agent Systems,* 15:221–252, 2007. 10.1007/s10458-006-9009-y.

[14] G.E. Kersten and S.J. Noronha. Rational agents, contract curves, and inefficient compromises report. Working papers, International Institute for Applied Systems Analysis, 1997.

[15] Gregory E. Kersten and Grant Zhang. Mining inspire data for the determinants of successful internet negotiations. *InterNeg Research Papers INR 04/01 Central European Journal of Operational Research,* 2003.

[16] Sarit Kraus. *Strategic Negotiation in Multiagent Environments.* Mit Press, October 2001.

[17] R. Lin, S. Kraus, D. Tykhonov, K. Hindriks, and C. M. Jonker. Supporting the design of general automated negotiators. In *Proceedings of the Second International Workshop on Agent-based Complex Automated Negotiations (ACAN'09),* 2009.

[18] Raz Lin, Sarit Kraus, Jonathan Wilkenfeld, and James Barry. Negotiating with bounded rational agents in environments with incomplete information using an automated agent. *Artificial Intelligence,* 172(6-7):823 – 851, 2008.

[19] Raz Lin, Yinon Oshrat, and Sarit Kraus. Investigating the benefits of automated negotiations in enhancing people's negotiation skills. In *AAMAS '09: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems,* pages 345–352, 2009.

[20] Martin J. Osborne and Ariel Rubinstein. *Bargaining and Markets (Economic Theory, Econometrics, and Mathematical Economics).* Academic Press, April 1990.

[21] H. Raiffa. *The Art and Science of Negotiation.* Harvard University Press, 1982.

[22] R. Ros and C. Sierra. A negotiation meta strategy combining trade-off and concession moves. *Autonomous Agents and Multi-Agent Systems,* pages 163–181, 2006.

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

[24] Liviu Dan Serban, Gheorghe Cosmin Silaghi, and Cristian Marius Litan. Agent fsega - time constrained reasoning model for bilateral multi-issue negotiations. In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear).* Springer-Verlag, 2010.

[25] Katsuhide Fujita Shogo Kawaguchi and Takayuki Ito. Compromising strategy based on estimated maximum utility for automated negotiating agents. In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear).* Springer-Verlag, 2010.

[26] C. Sierra, P. Faratin, and N.R. Jennings. A service-oriented negotiation model between autonomous agents. In M. Boman and W. van de Velde, editors, *Proceedings of the 8th European Workshop on Modelling Autonomous Agents in Multi-Agent World, MAAMAWŠ97,* volume 1237 of *Lecture Notes in Artificial Intelligence,* pages 17–35. Springer-Verlag, 1997.

[27] Niels van Galen Last. Opponent model estimation in bilateral multi-issue negotiation. In Takayuki Ito, Minjie Zhang, Valentin Robu, Shaheen Fatima, and Tokuro Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence (to appear).* Springer-Verlag, 2010.

[28] Michael P. Wellman, Peter R. Wurman, Kevin O'Malley, Roshan Bangera, Shou de Lin, Daniel Reeves, and William E. Walsh. Designing the market game for a trading agent competition. *IEEE Internet Computing,* 5(2):43–51, 2001.