# Optimal Negotiation Decision Functions in Time-Sensitive Domains

Tim Baarslag*, Enrico H. Gerding*, Reyhan Aydoğan†‡, m.c. schraefel*
*Agents, Interaction and Complexity Group
University of Southampton, UK
Email: T.Baarslag@soton.ac.uk, E.H.Gerding@soton.ac.uk, mc@ecs.soton.ac.uk
†Interactive Intelligence Group
Delft University of Technology, The Netherlands
Email: R.Aydogan@tudelft.nl
‡Computer Science Department
Özyeğin University, Istanbul, Turkey

***Abstract*—The last two decades have seen a growing interest in automated agents that are able to negotiate on behalf of human negotiators in a wide variety of negotiation domains. One key aspect of a successful negotiating agent is its ability to make appropriate concessions at the right time, especially when there are costs associated with the duration of the negotiation. However, so far, there is no fundamental approach on how much to concede at every stage of the negotiation in such time-sensitive domains. We introduce an efficient solution based on simultaneous search, which is able to select the optimal sequence of offers that maximizes expected payoff, given the agent's beliefs about the opponent. To this end, we show that our approach is consistent with known theoretical results and we demonstrate both its effectiveness and natural properties by applying it to a number of typical negotiation scenarios. Finally, we show in a number of experiments that our solution outperforms other state of the art strategy benchmarks.**

## I. Introduction

Negotiation is an important process for coordinating our actions and reaching agreements, and a crucial component of many important decisions, such as job negotiations or acquiring a house. We can also observe it in various everyday situations, such as setting a calendar date with a friend, or agreeing on a deadline. As a result, the last two decades have seen a growing interest in the automation of negotiation and e-negotiation systems [1], [2], and automated agents are becoming increasingly adept at negotiating on behalf of human negotiators [3].

A key ingredient of a successful negotiation is that both parties make concessions during the negotiation [4]. Concessions are used to elicit cooperation from the other, to convey information to the opponent about the negotiator's preferences and, perhaps most importantly, to accommodate the pressure of time [5], [6]. The fact that time is *costly* (typically in the form a deadline or a perceived maximum number of bidding rounds) puts important pressure on the parties to reduce their aspirations and operates as a force on the parties to concede. For example, a house searcher whose own house is getting sold might have ample time for a few negotiation exchanges, but may require some guarantees on the worst-time duration of the negotiation. Another important example is that it might be costly to *generate* suitable offers; e.g., when negotiating a highly complex domain that requires significant computational and cognitive resources.

Current automated negotiation research on concession strategies is mainly heuristic-based. One of the most well-known concession strategies is the family of time dependent tactics (TDT's) [7], [8], such as *Boulware* and *Conceder*, which are characterized by the fact that they consistently concede throughout the negotiation process as a function of time. Similar concession-based behavior can also be observed in practice in the Automated Negotiating Agents Competition (ANAC) [9], [10]. A variety of such state-of-the-art agents, such *Agent K* [11] (winner of ANAC 2010) and *HardHeaded* [12] (winner of ANAC 2011), as well as the time dependent tactics, select their concessions in a heuristic-based manner rather than being informed by fundamental theoretical insights; therefore, they make largely unfounded choices on how much to concede at every stage of the negotiation. In addition, there is a surprising lack of automated negotiation strategies that can proficiently take time-sensitive domains into account when deciding on an offer.

Against this background, we introduce a fundamental, decision-theoretic approach that optimally solves how to concede in time-sensitive domains, given certain beliefs about the opponent. Moreover, we present an algorithm that does so in a computationally efficient manner. This is not an easy task as there is a significant amount of planning involved in the optimal selection of bids: what should be offered at one moment naturally depends on the consequences of having it rejected and relying on subsequent bids. Conversely, reasoning backward poses a similar challenge, as what can be offered later depends on time costs dispensed earlier in the negotiation.

By adapting a technique from search theory called *simultaneous search* [13], our solution looks ahead and selects, given the agent's current beliefs, the optimal sequence of offers that maximizes expected payoff, while incurring the least amount of costs. Specifically, we calculate *how many* concessions we should make to optimize expected utility, in *what order* to make them, and *how much* we should concede at every negotiation step, given a model of the opponent's likelihood of accepting different offers.

Our solution exhibits the same properties as known theoretical results for a classic resource division problem of splitting a pie. Furthermore, we demonstrate its use in both distributive (i.e. single-issue, win–lose) and integrative (i.e. multi-issue, win–win) bargaining scenarios. We show that our algorithm exhibits desirable negotiation behavior such as compromising over time and aiming for fair and win-win outcomes. Moreover, we show that although the brute force method is intractable, the optimal strategy can be calculated in $O(n^2 \log n)$ time as a function of the domain size.

We begin with a discussion of related work in Section II and then we formalize our problem setting in Section III. We formulate our optimal concession strategy in Section IV, and we apply it to two negotiation cases in Section V. We subsequently compare our solution with state of the art bidding strategies in a series of tests (Section VI). We conclude our paper in Section VII, which summarizes the contributions of this work and outlines its implications.

## II. Related Work

The problem of finding an optimal bidding strategy under uncertainty is also studied by Fatima et al. in [8]. They determine optimal negotiation strategies for different information states and environments (e.g. symmetrical or asymmetrical deadlines) and conditions for their convergence. In this paper, the strategy set of both agents is restricted to time-dependent strategies as defined in [7], and only single-issue bargaining is studied. This is later extended in [14] to a multi-issue setting, but their focus is on finding optimal negotiation agendas and procedures, rather than negotiation strategies. Similarly, Hao et al. [15] focus on concession curves in discounted domains by introducing a parameter to balance exploiting and compromising. Much like the state-of-the-art agents mentioned in the introduction ([11], [12]), they build upon the heuristic approach of the time-dependent tactics.

Work that presents optimal choices of how much to concede also includes game theoretic work, for example in single-shot bargaining and the ultimatum game. For instance, Rubinstein [16] studies negotiation strategies in which each players bears a fixed cost for each period. Rubinstein finds equilibrium strategies in several different scenarios relating to differences between bargaining costs of the two players. This approach assumes a complete information setting, in which the deal is struck immediately, which we cannot easily apply to a typical concession-based negotiation of multiple rounds. Furthermore, this type of work typically revolves around equilibrium strategies, which assumes full rationality on the part of both agents. Our approach uses methods from decision theory, focusing on optimal solutions for one negotiating party, given certain beliefs about the opponent.

Another principled approach in formulating a concession strategy is given by Williams at al. [17]. They use Gaussian processes to predict the opponent's future behavior and to set the agent's concession rate dynamically during the negotiation by calculating the optimal time for an agreement. This approach only works in a real-time setting with many possible exchanges and when time pressure is induced by discount factor. We cannot meaningfully employ this technique in domains with different costs and a discrete time line.

An alternative approach is to study behavior dependent tactics that reciprocate the opponent's concessions (e.g. by employing tit for tat [7], [18]). Such adaptive strategies base the decision to make concessions on the actions of the other negotiating party, but they do not give us any information on how to concede based on time and costs.

## III. Problem Setting

Our negotiation setting consists of a *negotiation domain* $\Omega = \{\omega_1, \ldots, \omega_n\}$, which contains all possible *negotiation outcomes* or *agreements*. In this work, we focus on bilateral automated negotiations, in which agents take turns in exchanging offers using the *alternating offers protocol* [16]: at every point in time, the agent can make a bid to the opponent, and the opponent responds by indicating whether or not the offer is acceptable; if the offer is rejected, the opponent can respond with a counter-offer, and the process continues.

While the domain is common knowledge to the negotiating parties, the *preferences* of each player is private information. The agent and the opponent both have a utility function, denoted $U$ and $U_{\text{opp}}$ respectively, which maps each outcome $\omega \in \Omega$ to a utility in the range $[0, 1]$. As the players do not have access to the utility function of the opponent, they will need to learn about the other during the encounter and/or from previous encounters. To this end, the agent has an *opponent model*, in the form of independent acceptance probabilities $p_\omega$ for every outcome $\omega$, and which can be updated after each negotiation exchange.

The agent models its time-sensitivity through a *cost function* $c(k)$, assigning a cost to the maximum length $k$ (i.e., the maximum number of offers made by the agent) of the negotiation. Note that this implies that the costs depend on the *number* of selected offers and hence not on the particular *order* in which they are offered. Such cost functions can arise in a number of different circumstances. For example, the bids can be costly for the agent to generate; e.g., because they require computational resources or costly user feedback. Another reason may be the cost associated with the uncertainty about the duration of the negotiation; e.g., the longer the negotiation is allowed to go on for, the more uncertainty a seller has with respect to planning, stocking, etc. We do not restrict the specific type of cost function, but a special case we focus on in the paper is a *deadline*; by setting $c(k) = 0$ for $k < D$ and $c(k) = \infty$ for $k \geq D$, the costs impose a deadline in the form of a specified maximum number of rounds $D \in \mathbb{N}$.

The agents seek to reach an agreement while at the same time aiming to maximize their own utility. Note that simply maximizing the expected utility in the *current* round, i.e., sending the bid $\omega$ that maximizes $p_\omega \cdot U(\omega)$ may not be optimal, as it could be more beneficial, costs permitting, to take a chance with more 'risky' bids; i.e., offers with higher utility and a lower probability of being accepted.

Therefore, the agent needs to plan ahead when deciding on an offer. To this end, the agent needs to decide not only on a *number* $k$ of bids to make, but also, at the same time, *which* bids $x_1, \ldots, x_k \in \Omega$ to make. That is, if we denote by

$$\Omega^{\leq n} = \bigcup_{1 \leq k \leq n} \Omega^k$$

the set of all possible bid sequences, the agent's goal is to determine a sequence of bids $(x_1, \ldots, x_k) \in \Omega^{\leq n}$ (not containing duplicates) that optimizes the expected *negotiation payoff*; i.e., the expected utility of sending the first bid, or, when this is rejected, the expected utility of the second bid, and so on, subtracted by the costs:

$$p_{x_1} U(x_1) + (1 - p_{x_1}) \left[ p_{x_2} U(x_2) + (1 - p_{x_2}) \left[ \cdots \right] \right] - c(k). \quad (1)$$

In principle, one could calculate the optimal sequence by sampling every element of $\Omega^{\leq n}$ and selecting the one with the highest expected payoff. However, this would take factorial time to accomplish, which is infeasible even for a moderately sized negotiation space. In the next section, we will provide an algorithm that finds the optimal sequence in $O(n^2 \log n)$ time.

## IV. Optimal Bidding Strategy

Given a sequence of bids $\pi \in \Omega^k$, rearranging the terms in Eq. (1) yields the following expression for the expected utility of $\pi$:

$$EU(\pi) = \sum_{i=1}^{k} p_{x_i} U(x_i) \prod_{j=1}^{i-1} (1 - p_{x_j}) - c(k).$$

Our goal is to find the optimal bid sequence of *arbitrary* length $k \in \{1, \ldots, n\}$:

$$\pi^* = \underset{\pi \in \Omega^{\leq n}}{\arg\max}\, EU(\pi).$$

Note that there are three distinct aspects to solving this equation for $\pi^*$: we need to find the *number*, the *set*, and the *order* of the bids. We start with a crucial lemma stating that, for every sequence of bids, it is best to offer them in decreasing order of utility, regardless of the accompanying acceptance probabilities.

**Concession Lemma.** *Let $k \in \mathbb{N}$ and $x_1, \ldots, x_k \in \Omega$ be such that*

$$(x_1, \ldots, x_k) = \underset{\pi \in \Omega^{\leq n}}{\arg\max}\, EU(\pi).$$

*Then, $U(x_1) \geq U(x_2) \geq \cdots \geq U(x_k)$.*

*Proof:* The proof runs as a bubble sort on the sequence of bids, similar to the exchange argument in [19], but this time involving a costly selection mechanism. Suppose, on the contrary, that there exists an $s < k$ such that $U(x_s) < U(x_{s+1})$. We will show that $\pi = (x_1, \ldots, x_k)$ cannot possibly be optimal. Let $\overline{\pi}$ be defined by swapping $x_s$ and $x_{s+1}$ in $\pi$. We have:

$$\begin{aligned} EU(\overline{\pi}) = & \sum_{i=1}^{s-1} p_{x_i} U(x_i) \prod_{j=1}^{i-1} (1 - p_{x_j}) \\ & + U(x_{s+1}) p_{x_{s+1}} \prod_{j=1}^{s-1} (1 - p_{x_j}) \\ & + U(x_s) p_{x_s} (1 - p_{x_{s+1}}) \prod_{j=1}^{s-1} (1 - p_{x_j}) \\ & + \sum_{i=s+2}^{k} p_{x_i} U(x_i) \prod_{j=1}^{i-1} (1 - p_{x_j}) - c(\overline{\pi}). \end{aligned}$$

Therefore,

$$\begin{aligned} EU(\overline{\pi}) - EU(\pi) &= U(x_{s+1}) p_{x_{s+1}} + U(x_s) p_{x_s} (1 - p_{x_{s+1}}) \\ &\quad - U(x_s) p_{x_s} - U(x_{s+1}) p_{x_{s+1}} (1 - p_{x_s}) \\ &= U(x_{s+1}) p_{x_{s+1}} p_{x_s} - U(x_s) p_{x_s} p_{x_{s+1}} \\ &> 0. \end{aligned}$$

■

The Concession Lemma drastically reduces the number of sequences we need to inspect to find the optimal bid sequence. Since we know a concession-based strategy is optimal, it suffices to focus on the number of concessions, and how much to concede at every time step. Note that the Concession Lemma does not state that we should simply make bids from highest to lowest utility; it merely expresses that, once we know the optimal sequence of bids, they need to be sent out in decreasing order. In particular, it does not tell us how to select the right bids to optimize $EU(\pi)$.

Interestingly, a similar situation occurs in the cascade model studied in sponsored search literature [20], in which ads can be placed in ranked slots which are traversed linearly by a user. Each ad $a_i$ in slot $i$ is either clicked with probability $q_{a_i}$, or skipped with probability $c_{a_i}$, after which the scanning process continues with slot $i + 1$. A classic result by Kempe and Mahdian [19] is that the optimal placement of the ads is characterized by the fact that their ratios $\frac{U(a_i) q_{a_i}}{1 - c_{a_i}}$ are sorted in decreasing order. Neither model subsumes the other, as our model incorporates cost, while [19] has a general skipping probability $c_{a_i}$. However, comparing the two methods is possible when setting costs to zero and $c_{a_i} = 1 - q_{a_i}$, in which case Kempe and Mahdian's solution implies the ads (or bids, in our case) need to be tried in decreasing order of utility. As we will see below, this is consistent with the behavior of our algorithm, which does exactly this for the special case of no costs, in an arguably more straightforward way the dynamic programming solution proposed in [20].

Before we formulate a greedy algorithm for finding $\pi^*$ and prove that it is optimal in Theorem IV.1, we first provide some definitions. For any set of bids $S \subseteq \Omega$, define $\mathrm{incr}(S)$ as its corresponding sequence sorted by descending utility in terms of $U$. That is, $\mathrm{incr}(S) = (x_1, \ldots, x_{|S|})$, such that $\forall_i (x_i \in S \wedge U(x_i) \geq U(x_{i+1}))$. The *marginal improvement* of an offer $\omega$ with respect to an existing sequence $(x_1, \ldots, x_k)$ is simply the added negotiation payoff of placing $\omega$ in the right place of the sequence; i.e., $EU(\mathrm{incr}(\{x_1, \ldots, x_k, \omega\})) - EU(x_1, \ldots, x_k)$.

**Greedy Concession Algorithm (GCA).** *Iteratively select, in a greedy manner, bids that maximize marginal improvement in expected negotiation payoff, until this becomes negative. Send these bids out in order of decreasing utility. (See Algorithm 1.)*

The algorithm is greedy in the sense that once it has selected a set of bids $\{x_1, \ldots, x_{k-1}\}$, it chooses the best *addition* $x_k$ to this set by maximizing marginal improvement. Note that in line 5 of the algorithm, we can disregard the costs when maximizing $EU$, as the cost function does not depend on the order. The costs come into play, however, in the termination condition on line 6. We can prove that, in principle, every additional bid increases expected utility if we disregard the costs. Therefore, the GCA continues to include bids until

**Algorithm 1:** Greedy Concession Algorithm (GCA)

**Input**: The current negotiation state.
**Output**: A sequence $\pi^*$ of optimal bids.

1 **begin**
  `// Update using opponent model`
2   **for** $\omega \in \Omega$ **do**
3     `update`$(p_\omega)$;
4   **for** $k \in \{1, \ldots, |\Omega|\}$ **do**
    `// Note this invokes cost` $c(k)$
5     $x_k \longleftarrow \underset{\omega \in \Omega \setminus \{x_1, \ldots, x_{k-1}\}}{\arg\max} EU(\text{incr}(\{x_1, \ldots, x_{k-1}, \omega\}))$;
6     **if** $EU(\text{incr}(\{x_1, \ldots, x_k\})) < EU(\text{incr}(\{x_1, \ldots, x_{k-1}\}))$ **then**
7       $k \longleftarrow k - 1$;
8       **break**;
  `// Sort result in decreasing order`
9   $\pi^* \longleftarrow \text{incr}(\{x_1, \ldots, x_k\})$;
10   **return** $\pi^*$

even the best addition lowers overall expected utility due to marginal costs. For instance, disregarding costs, for $k = 1$, the GCA simply selects the offer with the highest myopic payoff:

$$x_1 = \underset{\omega \in \Omega}{\arg\max}\, EU((\omega)) = \underset{\omega \in \Omega}{\arg\max}\, p_\omega U(x_\omega).$$

For $x_2$, it selects the best bid, *given* that it has already selected $x_1$. The GCA will offer $x_2$ either before or after $x_1$, depending on the utility order:

$$\begin{aligned} x_2 &= \underset{\omega \in \Omega \setminus \{x_1\}}{\arg\max}\, EU(\text{incr}(\{x_1, \omega\})) \\ &= \underset{\omega \in \Omega \setminus \{x_1\}}{\arg\max} \begin{cases} p_\omega U(\omega) + (1 - p_\omega) p_{x_1} U(x_1), & \text{if } U(\omega) \geq U(x_1) \\ p_{x_1} U(x_1) + (1 - p_{x_1}) p_\omega U(\omega) & \text{otherwise.} \end{cases} \end{aligned}$$

We can prove, using techniques from simultaneous search theory [13], that remarkably, a greedy strategy is actually optimal for particular forms of the cost function.

**Theorem IV.1.** *If $c(k)$ is a convex increasing function, then the Greedy Concession Algorithm selects the optimal sequence of bids $\pi^* = \underset{\pi \in \Omega^{\leq n}}{\arg\max}\, EU(\pi)$.*

*Proof:* Let $k \in \mathbb{N}$ be such that $\pi^* = (x_1, \ldots, x_k)$. From the Concession Lemma we know that $U(x_1) \geq U(x_2) \geq \cdots \geq U(x_k)$. Now, instead of sending out offers $x_1$ to $x_k$ one by one, we imagine sending them all simultaneously to the opponent, who accepts a subset of these bids and communicates this back to the agent. It is easy to verify this is equivalent given the agent's current beliefs, as long as the agent always elects the highest bid in the subset of accepted bids:

$$\pi^* = \underset{S \subseteq \Omega, |S| = k}{\arg\max}\, EU(\text{incr}(S)).$$

In this form, the problem is about finding the optimal *set* of bids, which can be solved by simultaneous search techniques [13]. Since the function $S \mapsto \sum_{i=1}^{k} p_{\text{incr}(S)_i} U(\text{incr}(S)_i) \prod_{j=1}^{i-1} (1 - p_{\text{incr}(S)_j})$ is downward recursive and $c$ is a convex increasing function defined in terms of $|S|$, the greedy marginal improvement algorithm is optimal [13]. ■

Note that the optimality of the GCA holds for the agent's *current* beliefs, namely after the updating has been performed on line 3. These beliefs may change as a result of negotiation exchanges with the opponent at later stages; therefore, a new optimal solution needs to be calculated at the start of every round. Fortunately, the complexity of the algorithm greatly improves over the naïve approach of an exhaustive search over $\Omega^{\leq n}$:

**Proposition IV.1.** *The complexity of GCA is $O(n^2 \log n)$.*

*Proof:* The bottleneck of the GCA is in line 5, where the set $\{x_1, \ldots, x_{k-1}, \omega\}$ is sorted $n - k + 1$ times to compute the expected utility. However, $\{x_1, \ldots, x_{k-1}\}$ can be sorted once beforehand, so interleaving $\omega$ can be achieved in $O(\log k)$. In the worst case (i.e., $c(k) \equiv 0$), the outer loop on line 4 never breaks, and therefore, $n + (n-1) + \cdots + 1 = O(n^2)$ outcomes are tested to maximize $EU$, resulting in $O(n^2 \log n)$ overall complexity. ■

Observe that one can improve the complexity bound in case the costs induce a deadline $D$: the breaking condition of Algorithm 1 on line 6 is then guaranteed to hold after $D$ steps, leading to $O(n^2)$ complexity.

## V. Applications

In this section, we show how to apply the GCA to a number of negotiation scenarios. We first focus on a well-studied case involving a single negotiation issue with strictly opposite preferences in V-A, which is an example of *distributive bargaining*. We also study an *integrative bargaining* case concerning a multi-issue negotiation scenario in V-B.

### A. Dividing a Single Resource

As a concrete example of the conceding behavior of the GCA, we will consider a classical resource division scenario in which two players need to divide a unit-sized pie [16], [21]. This scenario is often studied in a game theoretic setting (e.g. [16]); in single-shot bargaining, it is also known as the *ultimatum game* [22]. These approaches usually assume a complete information setting with fully rational players, in which the deal is struck immediately. Here, we are interested in an optimal concession-based solution for one negotiating party, given certain beliefs about the opponent.

We discretize a pie $\Omega$ into $n$ slices of size $1/n$, after which the agents need to reach an agreement on who gets what part of the cake. That is, we instantiate $\Omega = \{0, 1/n, \ldots, 1\}$, where every outcome $x \in \Omega$ represents the part $x$ that the agent receives, while the opponent receives $1 - x$. We assume the players have linear preferences over the pie, so that $U(x) = x$, and the acceptance probability of the opponent is the opposite: $p_x = 1 - x$.

Given a deadline $D$, which bids should the agent select to optimize expected utility? With one bid remaining ($D = 1$) and even $n$, the answer the GCA gives is

$$x_1 = \underset{x \in \Omega}{\arg\max}\, x(1 - x) = 1/2.$$

The greedy algorithm also tells us which bids to consider with more time left. Figure 1 shows the results for any $D \leq n = 100$. The figure should be read as follows: for any given $D$ on the horizontal axis, all bids (with their utility shown on the vertical axis) to the left of $k$ are selected by the GCA, and then offered after first being sorted in decreasing order as per the Concession Lemma. For instance, we can see that the bid with utility $1/2$ is indeed the first one to be selected, followed by a bid with utility $0.625$. For the extreme case of $D = |\Omega| = n+1$, all offers are selected, which means that the agent should simply offer all slices in decreasing order.

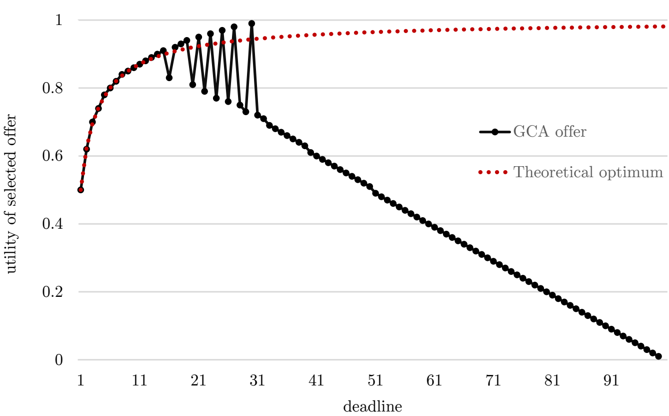



Fig. 1. The order in which offer are selected by the *GCA* (for the discrete variant of the resource division problem) and the *theoretical optimum* (for the continuous variant). Agent utility of the offers are indicated on the vertical axis.

The bids generated by the GCA are in line with theoretical results known for the *continuous case* of this problem, in which any real-valued division of the pie is a valid outcome. Specifically, consider a pie represented by the unit interval, $\Omega = [0,1] \subseteq \mathbb{R}$, in which every bid $x \in [0,1]$ is allowed. It is shown in [23] that the optimal bids are given by solutions of the following recurrence relation $v_j$ for $j \in \mathbb{N}$:

$$\begin{cases} v_0 = 0, \\ v_j = \frac{1}{2} + \frac{1}{2} v_{j-1}^2. \end{cases}$$

Ordering $v_j$ ($1 \leq j \leq k$) in decreasing order yields the required solutions $x_1, \ldots, x_k$. The graph of $v_j$ is plotted as the gray dotted line in Figure 1. The results are exactly aligned for small $k$. The discrepancies between the two can be attributed entirely to the fact that the GCA is applied to the discretized version of the domain, while the theoretical optimum is defined for the continuous case. In more detail, the GCA is unable to locate shares of exactly size $v_j$, leading to small discretization 'errors' that accumulate for higher $k$, which explains the erratic behavior near the end of the graph. This is easily remedied by increasing the number of partitions $n$ or, alternatively, by choosing an appropriate cut-off point for $k$, since the marginal improvement of adding another offer decreases quickly for higher $k$. For instance, in the case of $n = 100$, the marginal improvement is already smaller than $0.01$ for $k > 8$.

### *B. The Multi-Issue Negotiation Case*

We can also readily apply the GCA to the situation in which players negotiate *multiple* issues simultaneously, so that there may exist win-win outcomes that benefit both players.

We illustrate the GCA's behavior on a negotiation scenario called Itex–Cypress [24], which is used in the Automated Negotiating Agents Competition of 2010 [25]. It involves *Itex Manufacturing*, a producer of bicycle components, and *Cypress Cycles*, a builder of bicycles. Both sides negotiate multiple issues at once, such as the price, delivery times, and payment arrangements. There are 180 potential agreements in total, which denote all combinations of values for the issues. As before, the agent needs to take into account a deadline $D$.

Before we can apply the GCA to this scenario, the agent needs to establish an estimate of the opponent's acceptance probabilities $p_\omega$ for every offer $\omega \in \Omega$. One straightforward way of estimating this probability is through an estimate of the opponent's utility, for which we might employ a range of state of the art preference modeling techniques [26], [27], [28], [29]. Similar to the distributive case, suppose that the opponent's acceptance probability $p_\omega$ equals the opponent's utility $U_{\text{op}}(\omega)$.[1] This assumption has a number of attractive properties, which we now outline. Call an outcome dominated if there exists another outcome that is preferred by both. Then the following holds:

**Proposition V.1.** *Suppose $p_x = U_{op}(x)$ for all $x \in \Omega$. If any $\omega \in \Omega$ gets selected by the greedy concession algorithm, then all outcomes that dominate $\omega$ are also selected.*

*Proof:* If $\omega$ is dominated by $\omega'$, then it follows that $U(\omega)p_\omega > U(\omega')p_{\omega'}$. It is shown in [13] that this implies the marginal benefit of $\omega'$ is higher than of $\omega$. This holds irrespective of any previously selected offers (as long as they are different from $\omega$ and $\omega'$), and therefore, $\omega'$ will be selected first, in line 5 of Algorithm 1. ■

Note that it follows that Pareto efficient outcomes are always part of the offers selected by the algorithm. In particular, note that the *Nash point*, defined as $\omega_{\text{Nash}} = \max_{\omega \in \Omega} U(\omega) U_{\text{op}}(\omega)$, is the first to be selected by the algorithm and hence always part of the bidding sequence if $c(1) \leq U(\omega_{\text{Nash}}) p_{\omega_{\text{Nash}}}$. It is important to stress this does not imply that the Nash point is offered first (or last), as the bids are still sorted in decreasing order after the algorithm has finished selecting all bids. However, it does mean that the Nash point will always be among the offered bids, and in such a way that it is optimal to include it.

Figure 2 provides a visualization of the offers selected by the GCA as we increase the deadline $D$ when the agent plays as *Itex Manufacturing* in the Itex–Cypress scenario. From this, we can get a clear idea of the bidding strategy emerging from the algorithm: in general, it tends to propose offers that are grouped around the same iso-utility of the agent [30], with a bias towards Pareto efficient outcomes.

The algorithm favors win-win outcomes over selfish bids at the start of the selection phase but, as the marginal improvement of later bids decreases, the agent's own utility gains importance. As a result, the iso-curves tend to become more vertically oriented towards the left of the figure. For instance, the outcome with utility $(0.58, 0.78)$ (pictured in Figure 2 as the top-most outcome in the darkest region) is the 9th of bids to

[1]We expand upon alternative techniques for estimating the opponent's acceptance probabilities in Section VII

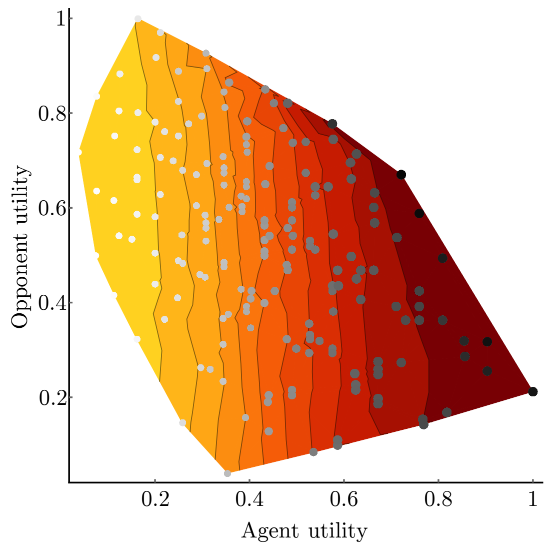



Fig. 2. The bids that the GCA selects in the Itex–Cypress scenario as we increase the deadline $D$. Every point represents one of the 180 possible outcome of the scenarios. The outcomes are divided in 10 different regions, all containing an equal amount of outcomes. Every region represents a set of offers that is selected around the same deadline. Bids are offered from dark to light; e.g., the dark region on the far right contains the bids selected by the algorithm for $D \leq 18$.

be selected by the algorithm, although it is only $47^{\text{th}}$ in terms of utility. This shows how desirable negotiation behavior such as locating highly efficient outcomes for *both* players naturally arises from the GCA.

## VI. Experimental Evaluation

In order to analyze the performance of the GCA, we developed a negotiating agent adopting this algorithm and implemented it within the GENIUS negotiation platform [31]. GENIUS is an environment for designing and evaluating automated negotiators' strategies, and has been used for the Automated Negotiating Agents Competition (ANAC) [9], [25]. We compared the performance of the GCA agent to the top performing negotiation agents from the past ANACs. For our experiments, we used a round-based alternating offers protocol with a deadline.

Specifically, we selected the following top-performing agents as benchmarks: *Agent K* (the winner of ANAC 2010 [25]), *HardHeaded* (the winner of ANAC 2011 [9]), *CUHKAgent* (the winner of ANAC 2012 [32]). In addition, we included the *Linear Conceder* agent [7], which is a well-known time-dependent concession strategy. We did not include the winners from ANAC 2013 and 2014, since these competitions had a different negotiation setup from preceding years.

All negotiation strategies were tested in four different negotiation scenarios with a varying domain size in order to show the scalability of the GCA strategy. Specifically, the scenarios we used are: *Laptop* (27 outcomes – ANAC 2011), *Itex-Cypress* (180 outcomes – ANAC 2010), *England-Zimbabwe* (576 possible outcomes – ANAC 2010), and *Grocery* (1600 outcomes – ANAC 2011).

For the opponent, we selected a strategy that simply accepts any offer with probability equal to the utility of the offer, as previously described in Section V. The set of acceptable offers is sampled before the start of each negotiation, and does not change during the negotiation. The probabilities are common information and hence are known to the agent. However, note that the agent does not know which offers will be accepted; it only has a faithful model of the opponent's acceptance probabilities. Furthermore, to avoid the effect of the acceptance strategies of the agents on the negotiation outcome, the opponent's offers are discarded. This enables us to have the performance depend solely on the concession strategies of the agents.

In our experiments, all agents negotiated against the same opponent on all negotiation scenarios with different negotiation deadlines (5, 10, 15, 20 and 25). Because of the non-deterministic behavior of the opponent (i.e. the set of acceptable offers was sampled anew for each negotiation), we repeated each negotiation 1000 times and we report average utilities gained by each agent, together with the standard error.

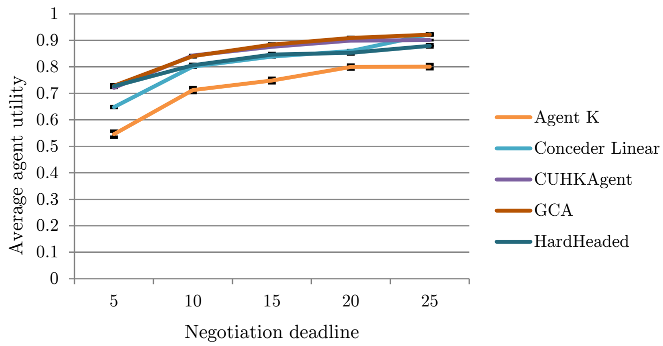
Figure 3 shows the average utility obtained by the agents for different deadlines and negotiation scenarios. It can be seen that the GCA agent outperforms the other agents.[2] Also, the average utility difference between the GCA agent and the other strategies is higher for an earlier deadline. This supports our hypothesis that the GCA agent is able to select the optimal sequence of offers even when the agents are allowed to make a limited number of offers during the negotiation. Another observation is that the average utility of the agents is higher for later deadlines in most cases. This shows that, if an agent has more time to explore, it is able to find the acceptable bids with a high utility for itself.

We now consider how the performance of the GCA agent compares across different scenarios. To this end, Figure 4 shows the same results as before but focuses on one particular deadline, namely 15 rounds. From this figure we can see that the performance of the GCA agent is fairly consistent across domains, and outperforms the state-of-art negotiating agents especially in the larger ones. For instance, when the agents negotiate on *Grocery* domain, the average utility gained by GCA agent is significantly higher than the others (0.95 versus 0.89, 0.87, 0.85 and 0.86). However, when they negotiate on the *Laptop* domain, the performance difference between GCA agent and other agent strategies is less significant (0.88 versus 0.75, 0.84, 0.88, 0.85). The reason for this is that, in small domains such as *Laptop*, a large fraction of the outcome space can be easily explored without the need for any optimization, whereas in large domains the agents need to be much more selective. The same trend is observed in settings with different deadlines.

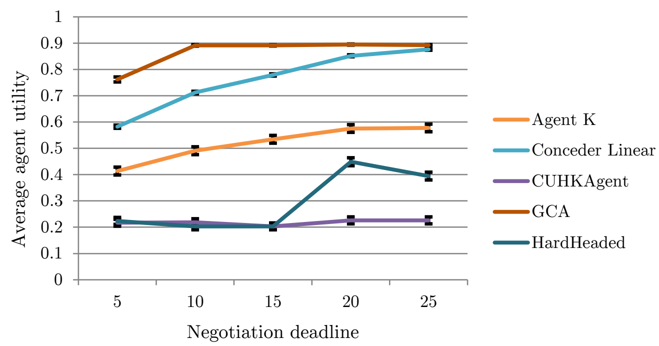
## VII. Conclusion and Discussion

In this work, we deal with informed concession strategies in time-sensitive domains, in which there are costs associated with the duration of the negotiation. Our solution provides an optimal sequence of offers to propose to the opponent, given how likely it is that an offer is accepted. Our method is optimal

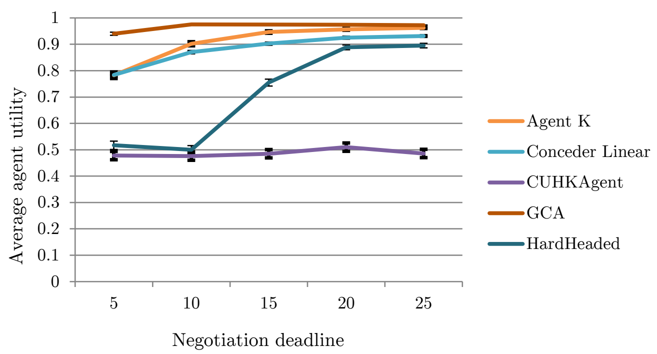
[2]This is to be expected as the GCA agent is the only one to optimally use knowledge about the opponent model. The point here is to show that, by using the opponent model in an optimal manner, this leads to better decision functions, and to see how the performance varies depending on the specific setting.



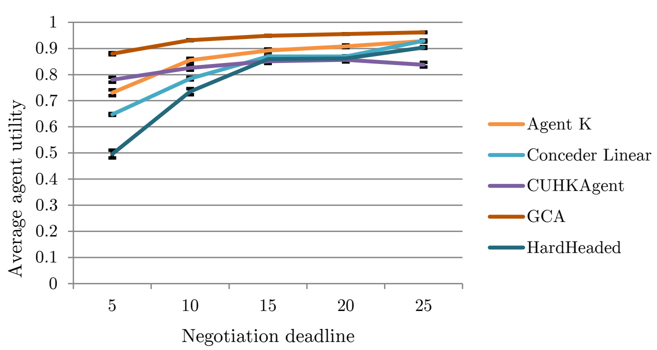

(a) Laptop Domain




(b) Itex-Cypress Domain




(c) England-Zimbabwe Domain




(d) Grocery Domain

Fig. 3. The average utility obtained by the agents for different negotiation scenarios and deadlines. The error bars denote the standard error to the mean.

in terms of the agent's own utility, and yet, it also takes the opponent into account when deciding on an offer. Through the use of a generic opponent model based on acceptance

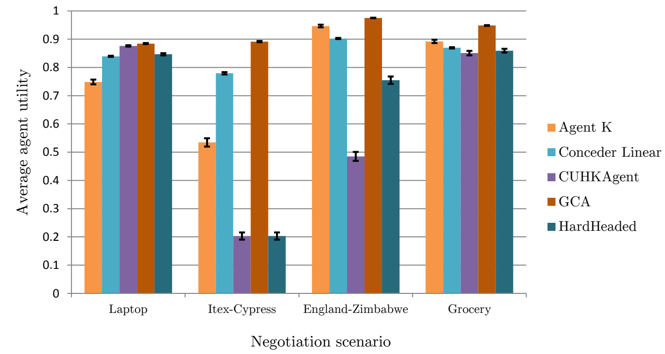



Fig. 4. The average utility gained by the agents in different negotiation scenarios. The error bars denote the standard error to the mean.

probabilities, the agent naturally aims for win-win outcomes on the Pareto frontier. Our algorithm scales with the negotiation domain size and performs well in a variety of circumstances, ranging from integrative to distributive negotiation scenarios.

We should stress that a negotiation strategy is a complex combination of different components (e.g. deciding when to accept, predicting the opponent's strategy). All such components are important and must work together to comprise an effective negotiation agent. Eventually, we envision a design of an automated negotiator that incorporates our optimal decision function with regard to time costs, while other types of concessions (e.g., to convey and withhold information, or to elicit cooperation) are handled separately by other decision components.

For future work, it would be interesting to combine our method with a number of different opponent modeling techniques. For example, estimates of the opponent's likelihood of acceptance could be inferred from previous interactions using transfer learning techniques or by adapting preference modeling techniques that are widely available for our setting. The most promising approach is to estimate the opponent's entire decision model. For example, by modeling the opponent's decision function based on the history of exchanged offers (i.e., rejected and proposed offers), as proposed in [33].

Finally, an interesting future application of our approach lies in multi-objective negotiation. For such cases, we would need to maximize a sum of weighted utilities associated with reaching the objectives, minus the sum of costs of acquiring them. This type of negotiation settings could be mapped to simultaneous *selection* problems [34], which could provide an interesting extension to our work.

## ACKNOWLEDGMENTS

This work was supported by the EPSRC, grant number EP/K039989/1 and the ITEA M2MGrids Project, grant number ITEA141011.

## REFERENCES

[1] G. E. Kersten and H. Lai, "Negotiation support and e-negotiation systems: An overview," *Group Decision and Negotiation*, vol. 16, no. 6, pp. 553–586, 2007. [Online]. Available: http://dx.doi.org/10.1007/s10726-007-9095-5

[2] S. Kraus, *Strategic Negotiation in Multiagent Environments*. MIT Press, Oct 2001.

[3] R. Lin and S. Kraus, "Can automated agents proficiently negotiate with humans?" *Communications of the ACM*, vol. 53, no. 1, pp. 78–88, Jan 2010. [Online]. Available: http://doi.acm.org/10.1145/1629175.1629199

[4] J. Z. Rubin and B. R. Brown, *The social psychology of bargaining and negotiation*. Academic press, 1975.

[5] P. J. Carnevale and E. J. Lawler, "Time pressure and the development of integrative agreements in bilateral negotiations," *The Journal of Conflict Resolution*, vol. 30, no. 4, pp. 636–659, 1986. [Online]. Available: http://www.jstor.org/stable/174079

[6] D. G. Pruitt, *Negotiation Behavior*. Academic Press, 1981.

[7] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Robotics and Autonomous Systems*, vol. 24, no. 3-4, pp. 159 – 182, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889098000293

[8] S. S. Fatima, M. J. Wooldridge, and N. R. Jennings, "Optimal negotiation strategies for agents with incomplete information," in *Revised Papers from the 8th International Workshop on Intelligent Agents VIII*, ser. ATAL '01. London, UK: Springer-Verlag, 2002, pp. 377–392. [Online]. Available: http://dl.acm.org/citation.cfm?id=648208.757345

[9] T. Baarslag, K. Fujita, E. H. Gerding, K. V. Hindriks, T. Ito, N. R. Jennings, C. M. Jonker, S. Kraus, R. Lin, V. Robu, and C. R. Williams, "Evaluating practical negotiating agents: Results and analysis of the 2011 international competition," *Artificial Intelligence*, vol. 198, pp. 73 – 103, May 2013. [Online]. Available: http://dx.doi.org/10.1016/j.artint.2012.09.004

[10] C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings, "An overview of the results and insights from the third automated negotiating agents competition (ANAC 2012)," in *Novel Insights in Agent-based Complex Automated Negotiation*, ser. Studies in Computational Intelligence. Springer, Japan, 2014, vol. 535, pp. 151–162. [Online]. Available: http://dx.doi.org/10.1007/978-4-431-54758-7_9

[11] S. Kawaguchi, K. Fujita, and T. Ito, "Compromising strategy based on estimated maximum utility for automated negotiating agents," in *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence*, T. Ito, M. Zhang, V. Robu, S. Fatima, and T. Matsuo, Eds. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 137–144. [Online]. Available: http://link.springer.com/content/pdf/10.1007%2F978-3-642-24696-8_8

[12] T. van Krimpen, D. Looije, and S. Hajizadeh, "Hardheaded," in *Complex Automated Negotiations: Theories, Models, and Software Competitions*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2013, vol. 435, pp. 223–227. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30737-9_17

[13] H. Chade and L. Smith, "Simultaneous search," *Econometrica*, vol. 74, no. 5, pp. 1293–1307, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1468-0262.2006.00705.x

[14] S. Fatima, M. J. Wooldridge, and N. R. Jennings, "Optimal negotiation of multiple issues in incomplete information settings," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, vol. 3. IEEE Computer Society, 2004, pp. 1080–1087.

[15] J. Hao, S. Song, H. Leung, and Z. Ming, "An efficient and robust negotiating strategy in bilateral negotiations over multiple items," *Engineering Applications of Artificial Intelligence*, vol. 34, pp. 45 – 57, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0952197614001067

[16] A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica*, vol. 50, no. 1, pp. 97–109, 1982. [Online]. Available: http://www.jstor.org/stable/1912531

[17] C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings, "Using gaussian processes to optimise concession in complex negotiations against unknown opponents," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, ser. IJCAI'11, vol. 1. AAAI Press, 2011, pp. 432–438. [Online]. Available: http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-080

[18] T. Baarslag, K. V. Hindriks, and C. M. Jonker, "A tit for tat negotiation strategy for real-time bilateral negotiations," in *Complex Automated Negotiations: Theories, Models, and Software Competitions*, ser. Studies in Computational Intelligence, T. Ito, M. Zhang, V. Robu, and T. Matsuo, Eds. Springer Berlin Heidelberg, 2013, vol. 435, pp. 229–233. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30737-9_18

[19] D. Kempe and M. Mahdian, "A cascade model for externalities in sponsored search," in *Proceedings of the 4th International Workshop on Internet and Network Economics*, ser. WINE '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 585–596. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-92185-1_65

[20] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey, "An experimental comparison of click position-bias models," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, ser. WSDM '08. New York, NY, USA: ACM, 2008, pp. 87–94. [Online]. Available: http://doi.acm.org/10.1145/1341531.1341545

[21] I. Stahl, *Bargaining theory*. Economic Research Institute, Stockholm, 1972.

[22] H. Raiffa, *The art and science of negotiation: How to resolve conflicts and get the best out of bargaining*. Harvard University Press, 1982.

[23] T. Baarslag, R. Hadfi, K. V. Hindriks, T. Ito, and C. M. Jonker, "Optimal non-adaptive concession strategies with incomplete information," in *Proceedings of The Seventh International Workshop on Agent-based Complex Automated Negotiations (ACAN 2014)*, 2014. [Online]. Available: http://ii.tudelft.nl/sites/default/files/optimalbidding_pub.pdf

[24] G. E. Kersten and G. Zhang, "Mining inspire data for the determinants of successful internet negotiations," *Central European Journal of Operational Research*, vol. 11, no. 3, pp. 297–316, 2003.

[25] T. Baarslag, K. V. 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*, ser. Studies in Computational Intelligence, T. Ito, M. Zhang, V. Robu, S. Fatima, and T. Matsuo, Eds., vol. 383. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 113–135. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-24696-8_7

[26] C. M. Jonker, V. Robu, and J. Treur, "An agent architecture for multi-attribute negotiation using incomplete preference information," *Autonomous Agents and Multi-Agent Systems*, vol. 15, pp. 221–252, 2007. [Online]. Available: http://dx.doi.org/10.1007/s10458-006-9009-y

[27] J. Zhang, F. Ren, and M. Zhang, "Bayesian-based preference prediction in bilateral multi-issue negotiation between intelligent agents," *Knowledge-Based Systems*, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705115001446

[28] F. Zafari, F. Nassiri-Mofakham, and A. Z. Hamadani, "Dopponent: A socially efficient preference model of opponent in bilateral multi issue negotiations," *Journal of Computing and Security*, vol. 1, no. 4, 2015. [Online]. Available: http://www.jcomsec.org/index.php/JCS/article/view/130

[29] D. Zeng and K. P. Sycara, "Benefits of learning in negotiation," in *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence*, ser. AAAI'97/IAAI'97. AAAI Press, 1997, pp. 36–41. [Online]. Available: http://dl.acm.org/citation.cfm?id=1867406.1867412

[30] P. Faratin, C. Sierra, and N. R. Jennings, "Using similarity criteria to make issue trade-offs in automated negotiations," *Artificial Intelligence*, vol. 142, no. 2, pp. 205 – 237, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0004370202002904

[31] R. Lin, S. Kraus, T. Baarslag, D. Tykhonov, K. V. Hindriks, and C. M. Jonker, "Genius: An integrated environment for supporting the design of generic automated negotiators," *Computational Intelligence*, vol. 30, no. 1, pp. 48–70, 2014. [Online]. Available: http://dx.doi.org/10.1111/j.1467-8640.2012.00463.x

[32] J. Hao and H. Leung, "Cuhkagent: An adaptive negotiation strategy for bilateral negotiations over multiple items," in *Novel Insights in Agent-based Complex Automated Negotiation*, ser. Studies in Computational Intelligence. Springer Japan, 2014, vol. 535, pp. 171–179.

[33] 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*, ser. AAMAS '05. New York, NY, USA: ACM, 2005, pp. 397–403. [Online]. Available: http://doi.acm.org/10.1145/1082473.1082534

[34] W. Olszewski and R. Vohra, "Simultaneous selection," 2014. [Online]. Available: http://faculty.wcas.northwestern.edu/~wol737/Selc.pdf