AhBuNe Agent

Winner of the Eleventh International Automated Negotiating Agent Competition (ANAC 2020)

Yildirim, Ahmet Burak; Sunman, Nezih; Aydoğan, Reyhan

DOI
10.1007/978-981-99-0561-4_6

Publication date
2023

Document Version
Final published version

Published in
Recent Advances in Agent-Based Negotiation

Citation (APA)
Yildirim, A. B., Sunman, N., & Aydoğan, R. (2023). AhBuNe Agent: Winner of the Eleventh International Automated Negotiating Agent Competition (ANAC 2020). In R. Hadfi, T. Ito, R. Arisaka, & R. Aydoğan (Eds.), *Recent Advances in Agent-Based Negotiation: Applications and Competition Challenges* (pp. 102-118). (Studies in Computational Intelligence; Vol. 1092). Springer. https://doi.org/10.1007/978-981-99-0561-4_6

AhBuNe Agent: Winner of the Eleventh International Automated Negotiating Agent Competition (ANAC 2020)

Ahmet Burak Yildirim[1(✉)], Nezih Sunman[1,3], and Reyhan Aydoğan[1,2]

[1] Computer Science, Özyeğin University, Istanbul, Turkey
{burak.yildirim,nezih.sunman}@ozu.edu.tr, reyhan.aydogan@ozyegin.edu.tr
[2] Delft University of Technology, Interactive Intelligence Group, Delft, The Netherlands
[3] Corporate Technology, Siemens A.S., Istanbul, Turkey

Abstract. The International Automated Negotiating Agent Competition introduces a new challenge each year to facilitate the research on agent-based negotiation and provide a test benchmark. ANAC 2020 addressed the problem of designing effective agents that do not know their users' complete preferences in addition to their opponent's negotiation strategy. Accordingly, this paper presents the negotiation strategy of the winner agent called "AhBuNe Agent". The proposed heuristic-based bidding strategy checks whether it has sufficient orderings to reason about its complete preferences and accordingly decides whether to sacrifice some utility in return for preference elicitation. While making an offer, it uses the most-desired known outcome as a reference and modifies the content of the bid by adopting a concession-based strategy. By analyzing the content of the given ordered bids, the importance ranking of the issues is estimated. As our agent adopts a fixed time-based concession strategy and takes the estimated issue importance ranks into account, it determines to what extent the issues are to be modified. The evaluation results of the ANAC 2020 show that our agent beats the other participating agents in terms of the received individual score.

Keywords: Automated negotiation · Agent competition · Partial preference ordering · Negotiation strategy

1 Introduction

Up to this point, various research challenges have been addressed in agent-based negotiation, where intelligent autonomous agents negotiate with each other or their human counterpart on behalf of their users [9,10,17,22]. The main challenges can be listed as generating bids under uncertainty about their opponent [7,14], learning the opponent's preferences and strategies during the negotiation [6,25], and determining when to accept the opponent's offer [8,21]. Researchers

R. Hadfi et al. (Eds.): IJCAI 2022, SCI 1092, pp. 102–118, 2023.
https://doi.org/10.1007/978-981-99-0561-4_6

aim to design effective negotiation strategies to beat opponents and maximize their received utility.

In this well-established research field, various negotiation strategies have been proposed so far. With the intention of providing a public benchmark to rigorously evaluate and compare those strategies, an international competition called Automated Negotiating Agent Competition (ANAC) has been organized since 2010 [13]. Initially, the competition focused on bilateral multi-issue closed negotiation where the agents have linear additive utility functions and negotiate with their opponents under a time-based deadline. Over the years, organizers have introduced various research topics such as reasoning non-linear utility functions in large-scaled negotiation domains [5,18], multilateral negotiations (i.e., having more than one opponent) [11], repeated negotiations [2], human-agent negotiations [19], diplomacy game challenges [12] and supply chain management [20].

Since it may not be trivial to elicit the user's complete preferences in terms of a linear utility function, the challenge of designing a negotiating agent having only its user's partial qualitative preference information came out in ANAC 2019 [4]. The following year, the organizers extended this challenge by introducing a variant of the Stacked Alternating Offers Protocol (SAOP) in which agents are not only able to generate offers or accept their opponent's counter-offers but also ask some preference elicitation questions to their users to reduce the uncertainty about their preferences.

There are a few studies regarding the design of effective agents with partial preferences in the literature. Aydoğan and Yolum present some heuristic approaches for partial preferences represented in terms of Conditional Preference Networks (CP-Nets) by exploiting the structure of the induced preferences graphs [3]. Furthermore, Tsimpoukis et al. propose to use a linear programming approach to estimate the agent's utility function given a set of pairwise comparisons of outcomes [24]. This work is mainly based on the approach to estimating the weights for multiple attributes in a composite criterion using pairwise comparisons [23].

We propose a heuristic-based negotiation strategy that can work under the desired protocol allowing agents to ask preference elicitation questions as well as make offers. Given the partial information about preferences, it first checks whether it has sufficient orderings to make reasoning about its complete preferences. Accordingly, it decides whether to sacrifice some utility in return for preference elicitation. In the case of making an offer, it uses the best-desired known outcome as a reference and modifies the content of the bid by adopting a concession-based strategy. Our agent analyzes the content of the given ordered bids to estimate the importance ranking of the issues. By adopting a fixed time-based concession strategy and taking the estimated issue importance ranks into account, it determines to what extent the issues are to be modified. The evaluation results of ANAC 2020 show that our agent beats the other participating agents in terms of the received individual score.

The rest of this paper is organized as follows: Sect. 2 provides the necessary background for ANAC 2020 negotiation setting while Sect. 3 explains the proposed heuristic-based negotiation strategy. Section 4 describes the experimental setup and reports the achieved results in the competition. Finally, Sect. 5 concludes the paper and discusses ideas and plans for future work.

2 ANAC 2020 Negotiation Setting

In ANAC 2020, GeniusWeb [16] framework is used to run the negotiation simulations in which agents negotiate with each other by following the Stacked Human Alternating Offers Protocol (SHAOP), which is an extension of SAOP [1]. In line with the underlying research challenge, a partial preference profile is given to each agent instead of their utility functions directly. That is, agents can compare some pair of outcomes according to the given profile under the given partial information about their own user preferences. However, the system has complete utility functions for its users and allows agents to query unknown preference orderings of some outcomes with a cost of a certain utility. As Fig. 1 summarizes the interaction among negotiating agents according to the SHAOP, one of the agents initiates the negotiation with an offer, and the negotiation is held in a turn-taking fashion. In each round, an agent can perform one of the following actions:

1. Requesting elicitation
2. Accepting the offer
3. Making a counteroffer (i.e., rejecting and overriding the previous offer)
4. Walking away (i.e., ending the negotiation without any agreement)

An oracle user represents an abstract agent having access to its user's complete preferences in terms of a linear additive utility function. It can compare the bids according to the given utility function and respond to the agent's preference queries. In other words, the oracle informs its agent by providing the order of the bids without exposing their utility values so that it expands the knowledge of its agent.

This process continues until an agreement or a deadline (e.g. 100 rounds) is reached. If the agents cannot reach an agreement by the given deadline, the negotiation fails. In such a case, both negotiating parties receive the utility of their reservation bid given by the GeniusWeb framework. Note that the agents know their own reservation bid. As seen in Fig. 1, agents can elicit more information about their user's preferences but each elicitation request penalizes the score of the agents with an elicitation cost. In elicitation queries, agents aim to learn the bids in a given list that are less preferred over a certain bid. For example, if there are five bids in the list and a particular bid μ, the system will return which bids out of those five bids are less preferred over μ according to

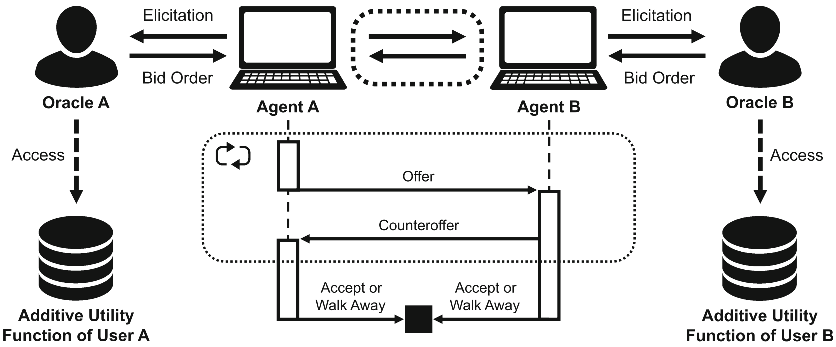

Fig. 1. Negotiation setting of ANAC 2020.

the complete preferences. At the end of the negotiation, the scores of the agents are calculated as the user utilities of the final agreement lowered with their total penalty. Winners are determined based on the average of the scores they earned in the tournament.

In this setting, a negotiation scenario consists of $I = \{1, 2, \ldots, n\}$ negotiation issues (or attributes) whose domain values are represented by $\mathcal{D} = \{D_1, \ldots, D_n\}$. An outcome is represented by o, while Ω represents the set of all possible outcomes in the negotiation domain (i.e., $D_1 \times D_2 \times ... \times D_n$). The agents' preferences are represented by means of linear additive utility functions in the form of:

$$\mathcal{U}(o) = \sum_{k \in I} w_k \times V_k(o[k]) \tag{1}$$

where w_k represents the importance of the negotiation issue k for the agent, $o[k]$ represents the value for issue k in outcomes o, and $V_k(.)$ is the valuation function for issue k, which returns the desirability of the issue value. Without losing generality, it is assumed that $\sum_{k \in I} w_k = 1$ and the domain of $V_k(.)$ is in the range of $[0, 1]$ for any k.

The total ordered profile is a set of outcome pairs $\mathcal{P}$ such that $\forall_{i \neq j} o_i, o_j \in \Omega \wedge o_i \succ o_j$, $(o_i, o_j) \in \mathcal{P}$ where $o_i \succ o_j$ denotes that the outcome o_i is strictly preferred over o_j. In such a profile, every outcome pair is comparable. $\mathcal{P}' \subset \mathcal{P}$ denotes the partial ordered profile. The set of unique bids, outcomes, inside a total ordered profile $\mathcal{P}$ and a partially ordered profile $\mathcal{P}'$ are represented by B and B', respectively.

In the following sections, the partially ordered profiles of the agent and the opponent are denoted by B'_A and B'_O, respectively. Also, the most preferred bid in a partially ordered profile B' is represented by B'^*.

3 Proposed Negotiation Strategy

Various strategies have been proposed to estimate the agent's precise utility information from the given partially ordered profile as mentioned in Sect. 1. Instead of predicting the precise utility information, our agent called AhBuNe Agent takes the most preferred bid from the provided partially ordered profile and changes the issue values of the bid regarding the time-based utility value lower bound and the counteroffers of the opponent (i.e., opponent's offer history). The strategy of selecting the number of issues to replace and the issue values for the replacement are explained in the following sections.

3.1 Preference Elicitation

Agents are allowed to know only the preference order $\mathcal{P}'$ of a proper subset B' of all possible bids B. Here, the main challenge is to design a strategy for preference elicitation, which allows the agents to acquire the unknown preference order of a given bid among partially ordered bids. Using an elicitation strategy, the agents sacrifice some utility (i.e., elicitation cost e_c) to perceive their preferences better. AhBuNe Agent utilizes an elicitation strategy applied in two different phases of the negotiation session. In this strategy, our agent calculates the maximum number of elicitation n_e to prevent being penalized significantly by their costs, see Eq. 2. For this purpose, the maximum elicitation penalization is determined as 0.05. By dividing this constant value by the elicitation cost defined in the competition setting e_c, the allowed maximum number of elicitation n_e is found. Note that all constant values used in our strategy are determined empirically.

$$n_e = \frac{0.05}{e_c} \tag{2}$$

Before the Negotiation Session Begins. Before starting the negotiation, AhBuNe Agent elicits n_b random bids to increase the number of ordered bids in B'_A so as to converge to the total ordering of all bids with respect to the agent's preferences. n_b is determined by the function given in Eq. 3, where $|\Omega|$ denotes the number of all possible bids in the given negotiation scenario.

$$n_b = \begin{cases} \max(\min(|\Omega| * 0.1 - |B'_A|, n_e), 0) & |\Omega| \leq 100 \\ \max(\min(10 - |B'_A|, n_e), 0) & \textit{otherwise} \end{cases} \tag{3}$$

For the domains containing less than 100 bids, we observed that the agent should know the order of at least 10% of all possible bids. If there are more than 100 bids in the domain, the minimum number of bids to be known is set to 10. If the agent initially knows only the order of less than the minimum number of bids determined above, it elicits preferences of the randomly selected n_b bids. Note that it does not exceed the number of allowed elicitation n_e. Accordingly, the agent updates B'_A after performing n_b elicitation and uses the updated partially ordered profile in the offering strategy.

In the Last Rounds. In the last rounds of the negotiation, where time t is over 0.98, our agent aims to find the most preferred bid among the opponent's previous offers. To find the target bid, the agent keeps the history of the bids offered by the opponent during the negotiation session B_{O_H}. In the last rounds, it calculates the number of conceded issue values of each bid offered by the opponent $B^i_{O_H} \in B_{O_H}$. The number of conceded issue values is calculated as the Levenshtein distance [15] L^i, between $B^i_{O_H}$ and the first offered bid by the opponent, which is assumed to be the opponent's most preferred bid B'^*_O as shown in Eq. 4 and Eq. 5 where D denotes Hamming distance.

$$D(a,b) = \begin{cases} 1 & a \neq b \\ 0 & otherwise \end{cases} \tag{4}$$

$$L^i = \sum_{k \in I} D(B'^*_O[k], B^i_{O_H}[k]) \tag{5}$$

Our agent finds the opponent's most conceded bids by sorting them according to their L^i distances. By taking n_e into account, it asks for the preference ordering of those opponent's most conceded bids one by one in descending order. Consequently, it learns the order of those bids according to its own preferences. Iterating from the most preferred one to the least, if the selected bid is acceptable for the agent (see Sect. 3.5), the agent offers that bid towards the end of the negotiation.

As a result, our agent offers an acceptable bid from the opponent's offer history, which means that the bid was acceptable for the opponent in a part of the negotiation session, in order to reach an agreement. If none of these bids are acceptable, the agent follows its default offering strategy explained in Sect. 3.4.

3.2 Estimating the Importance Order of the Issues

AhBuNe Agent extracts information from the partially ordered profile by focusing on the importance order of the issues to be able to decide whether a given bid is acceptable or not. In the given partial ordering, the ordered bids are considered as a list with indexes starting from 0 to $|B'_A|$. The importance of the bids is represented by their indexes in the list. For instance, assume that our agent has the partial ordering of the bids shown in Table 1. In the given example, the first row of the table indicates the least preferred bid.

Table 1. An example partial ordering list of an agent. Each column represents an issue and each row represents a bid.

Index	Music	Invitation	Drinks	Cleanup	Food	Location
0	DJ	Custom, Handmade	Catering	Special Equipment	Chips and Nuts	Ballroom
1	Band	Custom, Handmade	Non-Alcoholic	Special Equipment	Finger-Food	Ballroom
2	MP3	Photo	Catering	Hired Help	Finger-Food	Your Dorm
3	Band	Photo	Catering	Hired Help	Catering	Your Dorm
4	MP3	Photo	Non-Alcoholic	Hired Help	Catering	Party Tent
5	MP3	Custom, Handmade	Non-Alcoholic	Specialized Materials	Catering	Your Dorm
6	DJ	Custom, Handmade	Handmade Cocktails	Special Equipment	Catering	Ballroom
7	Band	Plain	Beer Only	Special Equipment	Handmade Food	Your Dorm
8	Band	Plain	Non-Alcoholic	Special Equipment	Finger-Food	Party Tent
9	MP3	Plain	Non-Alcoholic	Special Equipment	Catering	Party Tent
10	MP3	Custom, Printed	Beer Only	Special Equipment	Chips and Nuts	Party Room
11	Band	Photo	Catering	Water And Soup	Finger-Food	Party Room
12	MP3	Custom, Printed	Beer Only	Specialized Materials	Chips and Nuts	Party Room
13	Band	Custom, Printed	Handmade Cocktails	Special Equipment	Chips and Nuts	Party Tent
14	DJ	Custom, Printed	Beer Only	Water And Soup	Finger-Food	Party Tent
15	DJ	Custom, Printed	Handmade Cocktails	Hired Help	Handmade Food	Party Tent
16	DJ	Custom, Printed	Non-Alcoholic	Water And Soup	Finger-Food	Party Tent
17	MP3	Custom, Printed	Handmade Cocktails	Special Equipment	Handmade Food	Party Tent
18	Band	Custom, Printed	Beer Only	Water And Soup	Handmade Food	Party Tent
19	MP3	Custom, Printed	Handmade Cocktails	Water And Soup	Catering	Party Room

As explained in Sect. 3.1, the elicited bids, if they exist, are also incorporated into this list according to their learned preference order and the indices are updated accordingly. Our agent aims to estimate the importance order of each issue $k \in I$ by making inferences from the indexes. Consequently, it groups the indices of the bids by issue values for each issue. While finding the importance score of an issue, the mean index value of each issue value and their standard deviation is calculated as in Table 2.

Table 2. Example importance calculation of the music issue. The standard deviation of the mean index value of each issue value corresponds to the importance of the issue.

Issue values			
DJ	Band	MP3	
0	1	2	
6	3	4	
14	7	5	
15	8	9	
16	11	10	
-	13	12	
-	18	17	
-	-	19	**Standard deviation**
10.2	8.714	9.75	0.7618

For simplicity, our intuition is that if the standard deviation of the average indices for possible issue values is higher, it is considered a more important issue for the user. Here, the primary assumption is that all issue values are equally distributed in the given partial ordering of the bids. Without a doubt, it may not hold in all cases. It depends on the distribution of the values in the given partially ordered bids. By following the example above, the estimated importance order of the issues is listed in Table 3.

Table 3. Estimated importance order of the issues according to the standard deviations of the average indexes of their issue values.

Importance order	Issue	Standard deviation
1	Location	5.5658
2	Invitation	5.2009
3	Drinks	4.5874
4	Cleanup	4.2054
5	Food	2.9853
6	Music	0.7618

3.3 Opponent Modeling

When we analyze the strategies used by the state-of-the-art negotiating agents, we can see that they mostly offer their most preferred bid in the first round not to leave money on the table. Based on this observation, we use a heuristic in our opponent modeling, which assumes that the opponent's first offer is his/her most preferred bid. This bid is used to estimate the lower boundary of the opponent's target utility u_o^*. It is estimated by using the Levenshtein distance with an additional penalization value, which is explained in the following sections.

Estimating Opponent's Partial Ordering. As the opponent's partially ordered profile B'_O is not known by our agent, we adopt the same indexing approach explained in Sect. 3.2 to estimate it. Note that B'^{i}_O represents the bid at index i. Our agent keeps the history of its opponent's previous offers in a list similar to our partial ordering structure to estimate the opponent's preference list. Here, we assume that the first offer made by the opponent is the most preferred outcome, and it concedes over time. The opponent may not constantly concede during the negotiation and offer the same bid in different time slots. In such a case, we consider the time slot of its first appearance in the bid history. Using the estimated partial order, our agent can determine whether or not its opponent concedes enough.

Estimating the Most Preferred Issue Values. The opponent's most preferred issue values are determined by considering its bid history. It is assumed that the opponent offers his/her most preferred bids in the first rounds; therefore, the issue values that appeared in those rounds are considered the most preferred issue values. The target concession ratio of the opponent $1 - u_o^*$ is calculated by taking the difference between the maximum utility value that can be obtained (i.e., one) and the target utility value of the opponent u_o^*. Considering the target concession ratio of the opponent, we calculate the value of n_{mp} as in Eq. 6, which denotes the number of the most preferred bids consisting of the most preferred issue values. The set of most preferred issue values K_{mp}, see Eq. 7, is found by taking the unique issue values occurring in the most preferred n_{mp} bids.

$$n_{mp} = floor(|B'_O| * (1 - u_o^*)) \tag{6}$$

$$K_{mp} = \{B'^{i}_O[k] \mid k \in I, i \in [|B'_O| - n_{mp} - 1, |B'_O| - 1]\} \tag{7}$$

Estimating to What Extent the Opponent Concedes. To decide whether a given bid satisfies the opponent's target utility, it is assumed that each issue has a similar effect on the utility value, which is $1/|I|$. Equation 8 denotes to what extent the opponent concedes. If the issue values v in the given bids appear in the most preferred bid, then there is no concession (i.e., zero). If it does not exist in K_{mp}, that corresponds to a big concession. Otherwise, it corresponds to a small concession.

$$C(v) = \begin{cases} 0 & v \in B'^*_O \\ 2 & v \notin K_{mp} \wedge v \notin B'^*_O \\ 1 & otherwise \end{cases} \tag{8}$$

We estimate the approximate utility value $\hat{u}$ of a given bid o for the opponent as shown in Eq. 9.

$$\hat{u}(o) = \frac{(|I| - 1) - \sum_{k \in I} C(o[k])}{|I|} \tag{9}$$

Our agent desires that the opponent concedes at least as we do. Therefore, it generates an offer whose estimated utility for the opponent is less than our estimated utility lower bound.

3.4 Offering Strategy

Our agent follows a basic offering strategy changing the issue values of the most preferred bid considering a time-based lower boundary for the target utility. We randomly generate a bid meeting the lower utility boundary condition. In the following part, we explain how we calculate this boundary.

Lower Target Utility Boundary Curve. We adopt a time-based concession strategy where we calculate the lower target utility (TU) boundary. It represents the minimum target utility $TU_{min}(t)$ that the agent can concede to at a specific time t during the negotiation. The lower utility boundary calculation is given in Eq. 10. In the equation, t and $p_e(t)$ represent the time between $[0, 1]$ and the total elicitation penalization at time t, respectively. $p_e(t)$ is used to take the penalization cost into account when concerning the lower boundary of TU. The plotted version of the curve is shown in Fig. 2, in which the penalized elicitation cost variable is neglected because it is a dynamic variable that can vary during the negotiation.

$$TU_{min}(t) = \begin{cases} -(t-0.25)^2 + 0.9 + p_e(t) & 0 \le t < 0.5 \\ -(1.5 * (t-0.7))^2 + 0.9 + p_e(t) & 0.5 \le t < 0.7 \\ 3.25 * t^2 - 6.155 * t + 3.6105 + p_e(t) & otherwise \end{cases} \tag{10}$$

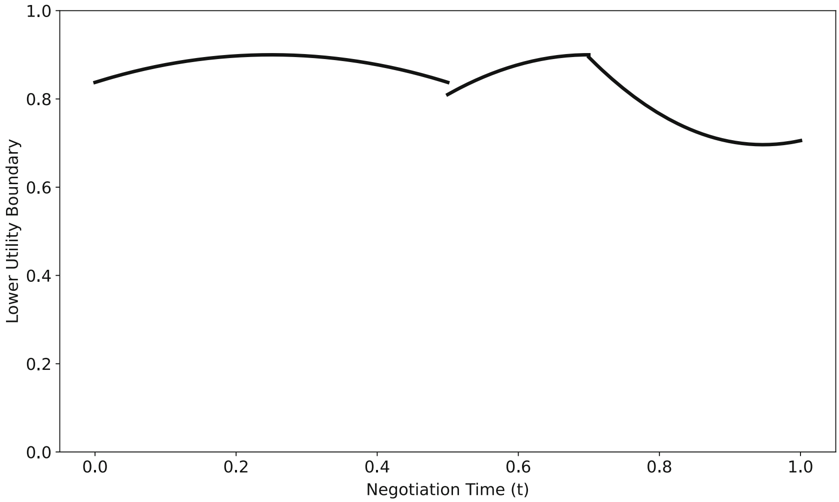

Fig. 2. Lower target utility boundary curve.

First, our agent is reluctant to reveal its most preferred offer. Therefore, instead of starting with the most preferred offer, as usual, our agent initially makes a random offers whose estimated utility is above $\sim$ 0.8 and slightly increases this boundary to 0.9 by hoping that its moves can be considered as a concession by the opponent (i.e., misleading the opponent about its own preferences). Then, it slightly decreases the lower boundary in order to make the opponent think that our agent is insisting on its most preferred bids because we sent similar bids at the beginning of the session.

By adopting a sarcastic movement again, the agent increases its lower boundary to make the opponent think that it is decreasing its target utility value the second time. We hope that the opponent perceives it as a concession and accepts one of our offers made in this phase. While approaching the deadline, our agent starts conceding to reach an agreement.

In the last moments (i.e., after reaching 98% of the negotiation deadline), instead of randomly generating bids, our agent elicits a subset of the opponent's bid history as explained in Sect. 3.1 to offer the most suitable one with the aim of increasing the chance of agreement.

Determining the Number of Issues to Be Replaced. While making an offer, our agent calculates the maximum number of issues to be changed, n_{ch}, based on the lower boundary of TU as seen in Eq. 11.

$$n_{ch}(t) = floor(TU_{min}(t) * |I|) \tag{11}$$

Then, it estimates the number of the most important issues (n_{mi}) and the least important issues (n_{li}) to be changed in the most preferred bid by following Eq. 12 and Eq. 13, respectively. Accordingly, it makes random issue value changes to generate an offer. Note that the upper half of the issue importance order found in Sect. 3.2 denotes the most important issues, while the bottom half denotes the least important issues.

$$n_{mi} = floor(n_{ch}/2) \tag{12}$$

$$n_{li} = n_{ch} \mod 2 + floor(n_{ch}/2) \tag{13}$$

Determining Which Issue Values to Replace. In order not to fall below the lower utility boundary, some of the issue values are excluded while some of them are given priority to be used for the replacement. Similar to the opponent modeling idea explained in Sect. 3.3, a set of desired issue values K_d is determined using the most important n_d bids. Note that n_d is calculated with a similar approach explained in the opponent modeling strategy, see Eq. 14.

$$n_d(t) = floor(|B'_A| * (1 - TU_{min}(t))) + 1 \tag{14}$$

Similar to the desired issue value set, the set of undesired issue values K_u is formed by considering the least important n_u bids. The issue values inside K_d are ignored in this process. The intuition behind this idea is that the issue values in the most preferred bids are more likely to be the most desired ones even if they exist in the least preferred bids. It is worth noting that n_u is determined by the distance between the current utility lower bound $TU_{min}(t)$ and the lowest value of the utility lower bound curve (i.e., $TU_{min}(1)$) as shown in Eq. 15.

$$n_u(t) = floor(|B'_A| * (TU_{min}(t) - TU_{min}(1))) + 1 \tag{15}$$

When determining a value of an issue for replacement, the values in K_u are not allowed to be used if there exists an alternative issue value. On the other hand, if the selected value is in K_d, then n_{mi} or n_{li} are decreased by one concerning the importance of the issue changed. Otherwise, both n_{mi} and n_{li} are decreased by one regardless of whether the issue is less or more important not to fall below the lower utility boundary.

Furthermore, the most crucial part of the issue value selection process is that the changed issue values are desired to be taken, if possible, from the values of the opponent's most preferred bid so that the generated bid becomes more acceptable for the opponent. After replacing the issue values, if the randomly generated bid satisfies all the conditions explained above and our lower boundary of target utility is higher than the opponent's one, then the generated bid is sent as an offer. Otherwise, we generate different randomly generated bids repeatedly until one of them satisfies the conditions.

3.5 Acceptance Strategy

If the opponent's offer has an estimated utility value greater than 0.9, AhBuNe Agent accepts this offer regardless of the lower utility boundary of the opponent. If this condition is not satisfied, AhBuNe Agent uses the strategy explained in Sect. 3.4 to estimate the lower utility boundary of a given bid. Then, using the opponent modeling strategy explained in Sect. 3.3, it also estimates the utility value of the bid for the opponent as well. As the last step, it compares these utility values. If the lower boundary of our utility value is greater than the utility of the opponent, and the utility lower bound $TU_{min}(t)$ at time t is satisfied, it accepts the offer. Otherwise, it makes a counteroffer using the algorithm explained in Sect. 3.4.

4 Evaluation

In ANAC 2020, 13 agents are submitted by eight institutions from seven countries. Scores of the agents are calculated by subtracting the average penalty values from the average received utility values of the tournament results. The finalists, the best-performing five agents, and the winner are determined with respect to their calculated scores. The tournament setup and the overall results are reported below.

4.1 Setup of the Tournament

The submitted agents are evaluated by organizing a tournament on the GeniusWeb 1.4.4 platform. The deadline in terms of rounds is set to 100 rounds. Four different negotiation domains were used in the competition where two partial preference profiles exist per domain. In the tournament, each negotiation session is run 10 times which results in 1560 negotiations per scenario. Each negotiation scenario is run with two different elicitation costs 0.01 and 0.001. The details of the negotiation scenarios used in the competition are explained in Table 4.

Table 4. Negotiation scenarios used in the ANAC 2020 tournament.

Domain name	Domain size	# Issues	# Partial bids	Reservation value
Flight Booking	$4 \times 3 \times 3 = 36$	3	10	~ 0.6
Japan Trip	$4 \times 4 \times 4 \times 3 = 192$	4	50	~ 0.2
Fitness	$5 \times 4 \times 4 \times 4 \times 4 = 1280$	5	50	~ 0.2
Party	$4 \times 4 \times 4 \times 4 \times 3 \times 4 = 3072$	6	75	~ 0.6

4.2 Results

In order to analyze the tournament results elaborately, we used ANAC 2020 tournament logs to compare the characteristics of the agents in terms of three metrics as follows:

- **Acceptance Ratio:** It is calculated by dividing the number of agreements by the number of total negotiation sessions.
- **Average Social Welfare:** It corresponds to the average of the social welfare of all agreements. Note that social welfare is the sum of the utilities of the negotiation outcome for both parties.
- **Average Individual Acceptance Utility:** It is the average of the utilities received by each agent and their opponents.

The tournament results of the finalist agents are provided in Table 5. It can be observed that AhBuNe Agent and AgentKT outperformed other agents in terms of average social welfare and average acceptance utility even though their acceptance ratios are lower than the Hamming Agent and the Shine Agent.

Table 5. ANAC 2020 tournament results.

Agent name	Acceptance ratio	Average social welfare	Average acceptance utility	
			Agent	Opponent
AhBuNe agent	0.5249	**1.4781**	**0.8611**	**0.6169**
Hamming agent	**0.6966**	1.4369	0.7510	0.6858
Shine agent	**0.6751**	1.4540	0.7428	0.7112
AgentKT	0.5354	**1.4740**	**0.8649**	**0.6091**
ANGELparty	0.4217	1.4579	0.8237	0.6342

For a better understanding of the tournament results, the performances of the finalist agents are analyzed in each domain as shown in Fig. 3. In all negotiation scenarios, our agent took either first or second place. Besides, there is a significant performance difference between our agent and Hamming Agent in the Flight Booking domain, which is the smallest domain in the tournament.

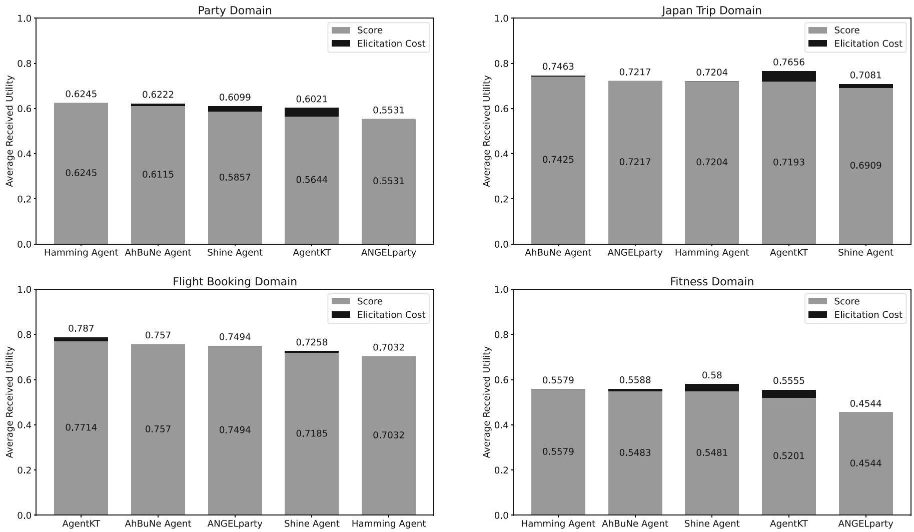

Fig. 3. ANAC 2020 tournament results per each domain.

Table 6 shows the overall results of the tournament per finalist agents in terms of received average utility, penalty, and score, respectively. As is seen, effective usage of preference elicitation plays a crucial role in the agents' success in terms of their final scores. The average utilities received by AhBuNe Agent and AgentKT are almost the same. However, AgentKT ranked in fourth place due to receiving a higher penalty score for the elicitation even though its acceptance rate is higher than ours. As a result, our agent received a higher score because of the low elicitation cost.

Furthermore, it is also seen that Hamming Agent and ANGEL Party did not perform any elicitation during their negotiation and could not outperform AhBuNe Agent. This may stem from having less information about their user's preferences. To sum up, the negotiation strategy used by AhBuNe Agent succeeded to balance the utility and the penalty scores. It is worth mentioning that there is a trade-off between preference elicitation and the cost of elicitation. As the elicitation number increases, agents get more insight into their users' preferences. However, it also causes a decrease in the received final score, which determines the winner.

Table 6. Overall ranking of ANAC 2020.

Rank	Agent name	Utility	Penalty	Score
1	AhBuNe Agent	**0.6623**	**0.0070**	**0.6554**
2	Hamming Agent	0.6484	0	0.6484
3	Shine Agent	0.6591	0.0187	0.6404
4	AgentKT	**0.6640**	0.0304	0.6336
5	ANGELparty	0.6096	0	0.6096

5 Conclusion

This paper describes our negotiation strategy designed for the research challenge addressed in ANAC 2020 where the agents are supposed to negotiate with their opponents by reasoning on their partial preference orderings. In the framework, they can pay elicitation costs in return for additional preference ordering. Therefore, one of the challenges is to determine when to elicit the user's preferences during the negotiation. The proposed strategy in this paper received the highest score in ANAC 2020 and became the winner of the competition. Instead of predicting the complete preference structure, our agent tries to predict the order of issue importance and the most preferred values by following a simple heuristic-based approach. Based on the assumptions regarding issue value changes in the opponent's bids during the negotiation, the agent infers the concession level of the opponent and accordingly generates its next offer. In future work, we plan to exploit the relationship between the structure of the utility function (i.e., additive utility function) and the given partial ordering so to get more insights into the preference ordering of the issue values and importance. It would be interesting to design an elicitation strategy not only at the beginning/end of the negotiation but also in the middle of the negotiation.

Acknowledgement. We would like to thank Prof. Dr. Catholijn Jonker, Assoc. Prof. Dr. Katsuhide Fujita, Assist. Prof. Dr. Reyhan Aydoğan, and Dr. Tim Baarslag for sharing the ANAC 2020 tournament results.

References

1. Aydoğan, R., Festen, D., Hindriks, K.V., Jonker, C.M.: Alternating offers protocols for multilateral negotiation. In: Fujita, K., et al. (eds.) Modern Approaches to Agent-based Complex Automated Negotiation. SCI, vol. 674, pp. 153–167. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-51563-2_10
2. Aydogan, R., Fujita, K., Baarslag, T., Jonker, C.M., Ito, T.: ANAC 2018: repeated multilateral negotiation league. In: The 33rd Annual Conference of the Japanese Society for Artificial Intelligence, Japan (2019)
3. Aydoğan, R., Yolum, P.: Learning disjunctive preferences for negotiating effectively. In: Eighth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 1201–1202 (2009). https://doi.org/10.1145/1558109.1558212

4. Aydoğan, R., et al.: Challenges and main results of the automated negotiating agents competition (ANAC) 2019. In: Bassiliades, N., Chalkiadakis, G., de Jonge, D. (eds.) EUMAS/AT -2020. LNCS (LNAI), vol. 12520, pp. 366–381. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-66412-1_23
5. Aydoğan, R., et al.: A baseline for non-linear bilateral negotiations: the full results of the agents competing in ANAC 2014. In: Frontiers in Artificial Intelligence: Intelligent Computational Systems, pp. 96–122. Bentham Science (2017)
6. Baarslag, T., Hendrikx, M.J.C., Hindriks, K., Jonker, C.: Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques. Auton. Agent. Multi-Agent Syst. **30**, 849–898 (2015)
7. Baarslag, T., Gerding, E.H., Aydoğan, R., Schraefel, M.: Optimal negotiation decision functions in time-sensitive domains. In: 2015 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 2, pp. 190–197 (2015). https://doi.org/10.1109/WI-IAT.2015.161
8. Baarslag, T., Hindriks, K.V., Jonker, C.M.: Effective acceptance conditions in real-time automated negotiation. Decis. Support Syst. **60**, 68–77 (2014). https://doi.org/10.1016/j.dss.2013.05.021
9. Baarslag, T., Kaisers, M., Gerding, E.H., Jonker, C.M., Gratch, J.: Self-sufficient, self-directed, and interdependent negotiation systems: a roadmap toward autonomous negotiation agents. In: Karagözoğlu, E., Hyndman, K.B. (eds.) Bargaining: Current Research and Future Directions, pp. 387–406. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-76666-5_18
10. Fatima, S., Kraus, S., Wooldridge, M.: Principles of Automated Negotiation. Cambridge University Press, Cambridge (2014)
11. Fujita, K., Aydoğan, R., Baarslag, T., Hindriks, K., Ito, T., Jonker, C.: The sixth automated negotiating agents competition (ANAC 2015). In: Fujita, K., et al. (eds.) Modern Approaches to Agent-based Complex Automated Negotiation. SCI, vol. 674, pp. 139–151. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-51563-2_9
12. de Jonge, D., Baarslag, T., Aydoğan, R., Jonker, C., Fujita, K., Ito, T.: The challenge of negotiation in the game of diplomacy. In: Lujak, M. (ed.) AT 2018. LNCS (LNAI), vol. 11327, pp. 100–114. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17294-7_8
13. Jonker, C.M., Aydoğan, R., Baarslag, T., Fujita, K., Ito, T., Hindriks, K.: Automated negotiating agents competition (ANAC). In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017), pp. 5070–5072. AAAI Press (2017)
14. Jonker, C.M., Aydoğan, R.: Deniz: a robust bidding strategy for negotiation support systems. In: Ito, T., Zhang, M., Aydoğan, R. (eds.) ACAN 2018. SCI, vol. 905, pp. 29–44. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-5869-6_3
15. Levenshtein, V.: Binary codes capable of correcting deletions, insertions and reversals. Soviet Phys. Doklady **10**, 707 (1966)
16. Lin, R., Kraus, S., Baarslag, T., Tykhonov, D., Hindriks, K.V., Jonker, C.M.: Genius: an integrated environment for supporting the design of generic automated negotiators. Comput. Intell. **30**(1), 48–70 (2014). https://doi.org/10.1111/j.1467-8640.2012.00463.x
17. Marsa-Maestre, I., Klein, M., Jonker, C.M., Aydoğan, R.: From problems to protocols: towards a negotiation handbook. Decis. Support Syst. 39–54 (2014)

18. Marsá-Maestre, I., López-Carmona, M.A., Klein, M., Ito, T., Fujita, K.: Addressing utility space complexity in negotiations involving highly uncorrelated, constraint-based utility spaces. Comput. Intell. **30**(1), 1–29 (2014). https://doi.org/10.1111/j.1467-8640.2012.00461.x
19. Mell, J., Gratch, J., Baarslag, T., Aydoğan, R., Jonker, C.M.: Results of the first annual human-agent league of the automated negotiating agents competition. In: In Proceedings of the 18th International Conference on Intelligent Virtual Agents, pp. 23–28. ACM (2018)
20. Mohammad, Y., Viqueira, E.A., Ayerza, N.A., Greenwald, A., Nakadai, S., Morinaga, S.: Supply chain management world. In: Baldoni, M., Dastani, M., Liao, B., Sakurai, Y., Zalila Wenkstern, R. (eds.) PRIMA 2019. LNCS (LNAI), vol. 11873, pp. 153–169. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33792-6_10
21. Razeghi, Y., Yavuz, O., Aydoğan, R.: Deep reinforcement learning for acceptance strategy in bilateral negotiations. Turk. J. Electr. Eng. Comput. Sci. **28**, 1824–1840 (2020). https://doi.org/10.3906/elk-1907-215
22. Sanchez-Anguix, V., Tunalı, O., Aydoğan, R., Julian, V.: Can social agents efficiently perform in automated negotiation? Appl. Sci. **11**(13) (2021). https://doi.org/10.3390/app11136022. https://www.mdpi.com/2076-3417/11/13/6022
23. Srinivasan, V., Shocker, A.: Estimating the weights for multiple attributes in a composite criterion using pairwise judgments. Psychometrika **38**, 473–493 (1973)
24. Tsimpoukis, D., Baarslag, T., Kaisers, M., Paterakis, N.G.: Automated negotiations under user preference uncertainty: a linear programming approach. In: Lujak, M. (ed.) AT 2018. LNCS (LNAI), vol. 11327, pp. 115–129. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17294-7_9
25. Tunalı, O., Aydoğan, R., Sanchez-Anguix, V.: Rethinking frequency opponent modeling in automated negotiation. In: An, B., Bazzan, A., Leite, J., Villata, S., van der Torre, L. (eds.) PRIMA 2017. LNCS (LNAI), vol. 10621, pp. 263–279. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69131-2_16