Automated Negotiations under User Preference Uncertainty: A Linear Programming Approach

Dimitrios Tsimpoukis[1,2], Tim Baarslag[1], Michael Kaisers[1], and
Nikolaos G. Paterakis[2]

[1] Centrum Wiskunde & Informatica, Science Park 123, 1098 XG Amsterdam, The
Netherlands, {T.Baarslag, D.Tsimpoukis, M.Kaisers}@cwi.nl

[2] Technische Universiteit Eindhoven, De Zaale, 5600 MB Eindhoven, The
Netherlands,
N.Paterakis@tue.nl

Abstract. Autonomous agents negotiating on our behalf find applications in everyday life in many domains such as high frequency trading, cloud computing and the smart grid among others. The agents negotiate with one another to reach the best agreement for the users they represent. An obstacle in the future of automated negotiators is that the agent may not always have a priori information about the preferences of the user it represents. The purpose of this work is to develop an agent that will be able to negotiate given partial information about the user's preferences. First, we present a new partial information model that is supplied to the agent, which is based on categorical data in the form of pairwise comparisons of outcomes instead of precise utility information. Using this partial information, we develop an estimation model that uses linear optimization and translates the information into utility estimates. We test our methods in a negotiation scenario based on a smart grid cooperative where agents participate in energy trade-offs. The results show that already with very limited information the model becomes accurate quickly and performs well in an actual negotiation setting. Our work provides valuable insight into how uncertainty affects an agent's negotiation performance, how much information is needed to be able to formulate an accurate user model, and shows a capability of negotiating effectively with minimal user feedback.

1 Introduction

Negotiation between two or more different parties is the joint decision making process towards a satisfactory outcome for all sides. If such an outcome is achieved, it constitutes an agreement.

In recent years there have been significant advancements in automating the negotiation process meaning that human negotiators are being represented by computer agents. Fully computerized negotiation offers a lot of benefits such as achieving better (win-win) deals for all sides, reduction in negotiation duration, and of course much reduced users' stress and frustration due to participation in the negotiation process [4]. *Automated negotiation* finds application in many

areas, some of which are high frequency trading, cloud computing and the smart grid. Such settings can be very dynamic, and as a result automating the negotiation process becomes imperative, considering that it is very uncomfortable for the user having to participate in negotiations so frequently, especially in domains in which they are not knowledgeable.

A major obstacle in the future of representative automated negotiation is the agent's level of knowledge about the preferences of the user it represents [6]. Preference elicitation is a tedious procedure to the users since they have to interact with the system repeatedly and participate in lengthy queries. To address this challenge, the agents should be able to accurately represent the users under minimal information about their preferences. Therefore, the agent must strike a balance between *user model* accuracy and user interference. Even though research in the field of automated negotiations has made progress on opponent modeling, in most cases the agents themselves were operating under fully specified preference profiles (see Section 2). The major questions that arise from the above problem are:

- How can we model the incomplete information about the users' preferences supplied to the agent in cases of uncertainty?
- How can we estimate user preferences from incomplete information?
- How does uncertainty about the user's preferences affect an agent's negotiation performance?

In this work we address the problems associated with negotiation under uncertainty and test the results in a scenario inspired from the smart grid. The contributions are threefold:

- We propose a way of representing user preference information, based on categorical data, showing preference relations between different possible outcomes.
- We present a method of estimating preference information from the incomplete information model based on linear optimization.
- We test the proposed method while negotiating on a smart grid cooperative scenario, examining the accuracy of the generated preference profile as well as actual negotiation performance.

The rest of this work is organized as follows: Section 2 discusses related work in the field of negotiation under uncertainty. Section 3 presents the problem setting, the key components in automated negotiation, as well as the new proposed ordinal data based incomplete information model. Section 4 describes the two proposed preference estimation methods. In Section 5 we are presenting the results of our strategies in terms of user model accuracy and negotiation performance, and in Section 7 we make suggestions for future work.

2   Related Work

The subject of modeling partial information about a user's preference profile, as well as the process of trying to formulate an accurate model of the user's real

preferences given incomplete information has been a topic of research through the years, but not a lot of it has been applied in the negotiation domain.

A key area of research in the field of user-preference modeling is *Multi-Attribute Utility Theory*. Many strategies have been proposed in this field, with the target of creating a preference profile under incomplete information of the user preferences. A large family of such strategies are the UTA (Utilité Additives) methods, originally proposed by Jacquet-Lagreze and Siskos in 1982 [16]. The UTA method and its extensions [14,21], obtain a ranked set of outcomes as input and formulate a new piecewise linear utility function through the use of linear programming. An application of the UTA method on the negotiation domain has been presented by Roszkowska [22]. The main limitation of the method is that the input outcome set needs to be a complete ranking of outcomes, meaning a total ordering (even though it might not include the whole outcome space). Even though the method we propose also utilizes a ranking as input in the experimental implementation, it can also work with any arbitrary set of partial orders.

Automated negotiation research has focused mostly on opponent preference modeling rather than on the user preference elicitation [7,8]. However, several techniques in opponent modeling are of interest to our case. Jonker et. al present an agent architecture that uses heuristics based on the opponent's bidding sequence to elicit opponent preference information [17]. Even though the user's preference profile is considered known in these methods, opponent modeling strategies could also be applied in user preference modeling.

Aside from multi-attribute utility theory, another option for representing user information was proposed by Boutilier et. al [10] and studied in the negotiation domain [6, 20]. CP-nets provide a graphical representation of conditional and qualitative preference relations. Cornelio et al. extend the CP-net concept to incorporate uncertainty in the networks including probabilistic elements [11], and Aydogan et al. [1,3] apply CP-nets theory in the negotiation domain using heuristics on the partial ordering of attribute values to generate a total ordering of outcomes. While CP-nets prove an effective way of representing partial information, our method is able to make the transition to utility-based negotiation based on ordinal data on full negotiation outcomes.

Our proposed decision model is inspired mainly by the work of Srinivasan and Shocker [24], who proposes a strategy for estimating the weights of different attributes given a set of pairwise comparisons of outcomes by using linear programming. The main limitation of this model, apart from the fact that it is limited to weight estimation, is that the evaluations of the stimuli values that appear on the comparison set need to be known. We extend this model to propose a different formulation of the problem using categorical data that estimates complete preference profiles based on the outcome set. We also formulate a simplified version of related work [24] for estimating the different negotiation issue weights.
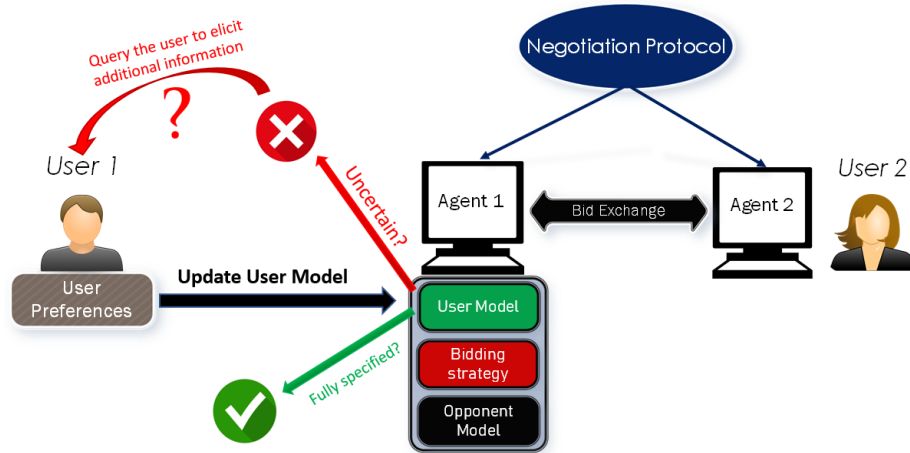
Fig. 1. Negotiation Setting

3   Problem Setting

3.1   Problem Description

Let us consider a setting where a computer *agent* is negotiating with an opponent agent on behalf of a *user* it is supposed to represent. The agents are exchanging bids (offers) to reach an agreement. The user that an agent represents has a set of preferences towards particular outcomes and the agent tries to achieve a negotiation outcome that satisfies these preferences as best as possible. This is schematically illustrated in Figure 1.

In a typical setting, an agent representing the user is assumed to have a *utility function* that fully describes the user's preference profile and translates every possible outcome to marginal values. However, the agent in a negotiation setting dost not necessarily have a priori, a fully specified user preference model, which we will call *user model*. Part of the reason is the user's discomfort to engage with the system continuously.

Consider a typical example in a smart grid domain where a user would have to interact with the energy management system in his residence to update his preferences. This is a tedious procedure to the user, which requires lengthy queries and on occasion even the user himself might not be sure about his preferences due to lack of negotiation domain knowledge. As a result, agents have to be able to negotiate on partial information about the user preferences, whilst also querying the user for additional information as little as possible.

In cases of uncertainty an agent might be provided with an initial set of information that might be known to it from domain knowledge, previous interactions with the user, past negotiation data etc. The information obtained from querying the user is another important concept when an agent needs to elicit information about the user's preferences. It may be difficult to query the user

for precise utility information, but, on the other hand, it could be much easier for a user to compare outcomes. To that regard, we will propose a partial information model based on ranked outcomes by the user which will be the initial information supplied to the agents.

Our goal is to formulate a strategy, where an agent will be able to generate a utility function that will approximate the real utility function as best as possible given a partial information model based on a set of ranked outcomes that are obtained from queries.

3.2   Formal Model

We present the key elements of a negotiation model architecture under preference uncertainty. These are: *the negotiation protocol, the negotiation scenario*, the users' *preference profiles*, the agents' *negotiation tactics*, presented below.

Negotiation domain During a negotiation, the participants are trying to reach an agreement over m issues which we denote as $I = \{1, ..., m\}$. For example, in the case of a smart grid collective these can include the price at which the energy is traded, the need for green energy utilization, the willingness to share, etc. Every issue i is discrete; i.e. each issue can take a finite number of n_i values which we denote as:

$$V_i = \left\{ x_1^{(i)}, x_2^{(i)}, ..., x_{n_i}^{(i)} \right\}. \tag{1}$$

The *negotiation domain* $\Omega = V_1 \times V_2 \times \ldots \times V_m$ is the set of all possible negotiation outcomes. A negotiation outcome $\omega \in \Omega$ is thus an m–tuple that assigns a single value $\omega_i \in V_i$ to every issue i, such as "0.15 €/kWh" for the issue "price of energy to be traded".

Negotiation Protocol The *negotiation protocol* dictates the actions that can be performed by an agent at any given moment. We will be using the widely used *Alternating Offers Protocol*, where each participant gets a *turn* per round [8]. On each turn, the agent can accept or propose a counter-offer (or *bid*)to the last offer by the opponent.

A negotiation *deadline* can be specified as the maximum number of negotiation *rounds*, or in real time quantities. If an agreement has not been achieved within the time-frame specified by the deadline, the negotiation ends and all participants obtain utility zero. We assume the negotiation deadline is universal and known to all participating agents [8, 13].

Preference Profile Every user participating in a negotiation has a specific set of preferences regarding the possible outcomes. The preference profile is given by an ordinal ranking over the set of possible outcomes: an outcome ω is said to be weakly preferred over an outcome ω' if $\omega \succeq \omega'$ where $\omega, \omega' \in \Omega$, or strictly preferred if $\omega \succ \omega'$.

Under mild assumptions [18], preference profiles can be expressed in a cardinal way through the use of a *utility function* such that:

$$\omega \succeq \omega' \iff u(\omega) \geq u(\omega'). \tag{2}$$

There is an outcome of minimum acceptable utility which is called *reservation value* [5, 12].

We will focus on *linear additive utility functions*, in which every issue i's value is calculated separately according to an evaluation function v_i as follows:

$$u : \Omega \mapsto [0,1] \subseteq \mathbb{R} \quad \text{with} \quad u(\omega) = \sum_{i=1}^{m} w_i \cdot v_i(\omega_i), \tag{3}$$

$$\text{where} \quad \sum_{i=1}^{m} w_i = 1. \tag{4}$$

Here, w_i are the normalized weights that indicate the importance of each issue to the user, and $v_i(\omega_i)$ is the evaluation function that maps the ith issue value to a utility. Note that the linear utility function does not take dependencies between issues into account. Alternatively, non-linear utility functions can be incorporated to describe such dependencies [2, 15, 19].

Agent The agent's *bidding strategy* defines the agent's structuring of the bids during a negotiation [4], mapping negotiation states to an action (*Acceptance*, or a *Counter-offer* in the Alternating Offers Protocol). The agent can perform better with an idea of the opponents' preferences and bidding strategy through *opponent modeling* techniques to propose bids which are more likely to be accepted [7, 23].

Well-known bidding strategies include the *time-dependent bidding tactics* where the result of the decision functions is based on the time passed in the negotiation [9, 13] as follows:

$$u(t) = P_{min} + (P_{max} - P_{min}) \cdot (1 - F(t)), \tag{5}$$

$$F(t) = k + (1 - k) \cdot (1 - k) \cdot t^{\frac{1}{e}}, \tag{6}$$

where P_{min}, P_{max} are the minimum and maximum accepted offers, t is the normalized[3] time $t \in [0,1]$ and $k \in [0,1]$ is the utility of the first offer. If $0 < e < 1$ the agent does not reduce its target utility in the early stages of the negotiation and concedes at the end of the deadline [9, 13]. The agent that follows this type of strategy is called *Boulware*. In the opposite case of $e \geq 1$, the agent is called Conceder as it concedes to its reservation value (P_{min}) very quickly.

---

[3] The time range of a negotiation usually is $[0, D]$ where D is the deadline in rounds or time units and is normalized to the values $[0, 1]$.

User Model The representative agent has a *user model*, which consists of the agent's beliefs about the user's preferences. We assume the agent is supplied with an initial set of answers to queries to the user, in the form of a user *ranking* $\mathcal{O} \subseteq \Omega$ of d different negotiation outcomes [24]:

$$\mathcal{O} = \left\{ o^{(1)}, o^{(2)}, \dots, o^{(d)} \right\}, \tag{7}$$

$$\text{where } o^{(1)} \succeq o^{(2)} \cdots \succeq o^{(d)}, \quad o^{(i)} \in \Omega.$$

This set will usually only contain a fraction of all outcomes Ω and hence determines its level of uncertainty. Note that this notation allows us to denote the ith issue value of an outcome $o^{(j)} \in \mathcal{O}$ with $o_i^{(j)}$ as before. The ranking $\mathcal{O}$ can alternatively be expressed as a set of $d - 1$ pairwise comparisons:

$$\mathcal{D} = \{(o^{(j)}, o^{(j+1)}) \mid o^{(j)} \in \mathcal{O} \text{ and } 0 < j \leq d - 1\}. \tag{8}$$

Given the outcome ranking $\mathcal{O}$, the agent's goal is to formulate its own estimated utility function $\hat{u}(\omega)$ that approximates the real utility function $u(\omega)$ as much as possible. Establishing 'the most likely' utility function from a ranking of outcomes $\mathcal{O}$ is complicated, as there is far less information available in $\mathcal{D}$ than in $u(\omega)$. Furthermore, $\mathcal{O}$ might not contain any information about particular outcomes, especially in large domains, requiring completion of the ordering.

4   Estimating a utility function from a user model

If an agent is to operate under a non-fully specified preference profile, it needs to formulate a strategy that will be able to derive a utility function from a set of pairwise comparisons of outcomes. To do so, we will extend an approach followed in [24].

Consider a ranking $\mathcal{O}$ of negotiation outcomes and the set $\mathcal{D}$ of corresponding pairwise comparisons. Given the pairwise comparisons, the same inequality should hold for the utility function of the agent (2). From the definition of the utility function (2), we can integrate the weight and each evaluator value in one variable and we rewrite (3) as:

$$u : \Omega \mapsto [0, 1] \subseteq \mathbb{R} \quad \text{with} \quad u(\omega) = \sum_{i=1}^{m} \phi_i(\omega_i), \tag{9}$$

$$\text{with} \quad \phi_i(\omega_i) = w_i \cdot v_i(\omega_i). \tag{10}$$

This results in a new discrete set of variables

$$Y = \left\{ \phi_1(x_1^{(1)}), \dots, \phi_1(x_{n_1}^{(1)}), \phi_2(x_1^{(2)}), \dots, \phi_2(x_{n_2}^{(2)}), \phi_m(x_1^{(m)}) \dots, \phi_m(x_{n_m}^{(m)}) \right\}. \tag{11}$$

With one additional piece of information, estimating the utility function can be translated into a linear optimization problem with the set Y as the set of

unknown variables. For each pairwise comparison between outcomes $(o, o') \in \mathcal{D}$ we derive from (2) and (9) that:

$$\sum_{i=1}^{m} \big(\phi_i(o_i) - \phi_i(o_i') \big) \ \geq \ 0, \quad \text{with} \quad \phi_i(o_i), \phi_i(o_i') \in Y. \tag{12}$$

We denote the above term as $\Delta u_{o,o'}$ so

$$\Delta u_{o,o'} = \sum_{i=1}^{m} \big(\phi_i(o_i) - \phi_i(o_i') \big), \quad \Delta u_{o,o'} \geq 0. \tag{13}$$

Now, we can translate the above inequalities into a linear optimization problem using standard linear programming techniques. For this, we need to consider 'slack variables' namely $z_{o,o'}$ and the linear program is formulated as:

$$\text{Minimize:} \quad \sum_{(o,o') \in \mathcal{D}} z_{o,o'} = F, \tag{14}$$

subject to the constraints:

$$z_{o,o'} + \Delta u_{o,o'} \geq 0, \tag{15}$$

$$z_{o,o'} \geq 0, \text{ for } (o, o') \in \mathcal{D}, \tag{16}$$

$$\phi_i(x_j^{(i)}) \geq 0, \text{ for } i \in I, \ j \in \{1, 2, ..., n_i\}. \tag{17}$$

In its current form the optimization problem yields the *trivial solution* where all $\phi_i(x_j^{(i)}) = 0, z_{o,o'} = 0$. To tackle this problem an additional constraint is required. Hence, arises the need for some additional piece of information about the preferences of the user. In our solution, the additional information is the *best* outcome for the user, i.e. the outcome of maximum utility ω^*. Note that this does not mean that we know the importance of each separate issue, but only that particular outcome that is the most desired from the user. This translates into our final constraint for the optimization problem:

$$u(\omega^*) = 1 \quad \Rightarrow \quad \sum_{i=1}^{m} \phi_i'(\omega_i^*) = 1. \tag{18}$$

From the constraints (15) and (16) we can see that

$$z_{o,o'} \geq \max\{0, \ -\Delta u_{o,o'}\}. \tag{19}$$

Given that the goal is to minimize F, **the optimal solution** will be

$$z_{o,o'}^* = \max\{0, \ -\Delta u_{o,o'}\}. \tag{20}$$

We should note at this point that, in the case that the initial judgment about the comparisons in $\mathcal{D}$ is correct, Δu_{jk} will always be positive. As a result all $z_{o,o'}$ will equal 0. This is an interesting attribute of this method, since it can

determine and pinpoint errors in user judgment as well. If all the $z_{o,o'}$ are not zero after solving the linear program, there is no solution set that satisfies all the comparisons in $\mathcal{D}$. This means that at lease one stated judgment regarding the preference of two or more outcomes is wrong. This can prove very important when the agent queries the user for information, in cases where the user is not entirely sure about their preferences or does not have complete knowledge of the negotiation domain and might give wrong feedback about their own preferences. With our method these errors in user feedback can be pinpointed and addressed.

5 Experiments and Results

5.1 Setup

Scenario To show that our proposed model is applicable in a real negotiation setting we conducted experiments in a simulation of a negotiation scenario. The scenario is inspired by the smart grid domain and refers to a fictitious energy cooperative, where different residents participate in energy exchanges. The issues of negotiation consist of the amount of energy bought or sold in different periods of time, which in electrical energy systems are called *Programme Time Units* (PTUs), and the type of energy exchanged (Green, Conventional). The possible issue values for every PTU are {-3kWh, -2kWh, ...,+2kWh, +3kWh}. The sign of the value corresponds to whether the the user buys or sells the given amount of energy. After an agreement is reached, for every PTU the amount sold by one user is bought by the opponent. We created different preference profiles based on different energy requirements patterns.

Measures We selected two metrics to evaluate our model: *accuracy* towards the real preference profile and *negotiation performance.*

To evaluate our model in terms of accuracy we compare the estimated weights and evaluator values compare to the real preference profile. The comparison was made in terms of 3 different accuracy measures: the *Pearson Correlation Coefficient*, the *Spearman Ranking Coefficient* of bids between the entire resulting utility space and the real utility space, and the maximum single bid utility distance in the set. The Pearson Correlation Coefficient assesses linear relationships between the two utility functions, while the Spearman Correlation assesses monotonic relations based on their rank. The accuracy measures experiments were performed on two scenarios with 3 (2 PTUs & Type of Energy) and 5 (4 PTUs & Type of Energy) issues respectively. In practice, the outcome space of the 5 issue domain is 5000 times bigger than the 3 issue one. The calculations were the result of averaging after 100 iterations for each uncertainty level.

Apart from the accuracy measures, we also investigate the influence of preference uncertainty in an actual negotiation and how well an agent performs in this setup using our utility estimation strategies. To do so, we examined the trace of the target utilities throughout a negotiation session of the *Conceder* agent, presented in Section 3.2. To monitor the complete negotiation trace, we

pitted the Conceder agent against a *never-accepting agent*. The deadline was set at 180 rounds and the negotiation domain only on a 5-issue smart grid scenario (4 PTUs & Type of Energy).

Baseline Strategy We tested our strategy against a simple preference estimation method called *Benchmarking Strategy*, based on the intuition that the more desired outcomes appear in the high positions of the ranked outcome set $\mathcal{O} = \left\{ o^{(1)}, o^{(2)}, \ldots, o^{(d)} \right\}$. According to this method, all issue values occurring in $o^{(i)}$ are awarded $d - i$ points (for example, the values that make up the most preferred outcome $o^{(1)}$ all receive maximum points). These scores are summed for every i and then renormalized to values between 0 and 1 to determine the final score for each value of every issue.

Weight Estimation Strategy In some negotiation scenarios the preferences for each issue might be known but the importance of each issue to the user is not. For this case we created a simpler strategy based on our Linear Programming model where the evaluator functions are known but the weights are not. The solving strategy is exactly the same to the one presented in Section 4, with the difference that the unknown variables are the issue weights only, since the evaluator values are known. Hence, we replace the final constraint (18) with (4), which states that the sum of the weights must equal 1.

5.2   Results

Accuracy Measures Figure 2 presents the accuracy of the model compared to the real preferences for 2 different domain sizes: 3 and 5 issues. The level of uncertainty is expressed as the number of outcomes that appear on the ranked set that is supplied to the agent. The first thing that we infer from the accuracy result figures is that our model becomes very accurate even with very few comparisons (less than 1% of the $d - 1$ required for a perfect ranking of the outcomes). Both our models outperform the *benchmark strategy* significantly. Especially in the Weight Estimation case, all measures rapidly converge to the desired values. This is reasonable if we consider the fact that in this case the evaluator functions are considered known, which is already a very large amount of information about the user preferences. To verify this claim, we ran a test case for the 5 issue domain where the agent was supplied with 0 comparisons and the weights were set to 0.2. The results yielded Pearson and Spearman coefficient values above 0.7, which indicates relatively high accuracy even when the weights are off.

We present the results up until 150 comparisons for each experiment since all metrics converge to their final values by then. However, we need to remark that only the Spearman correlation coefficient reaches its target value of 1 in the case of a total ranking of outcomes ($d - 1$ comparisons). This means that although the actual values of the weights and corresponding utilities might not be exactly the same the ranking of outcomes is correct.
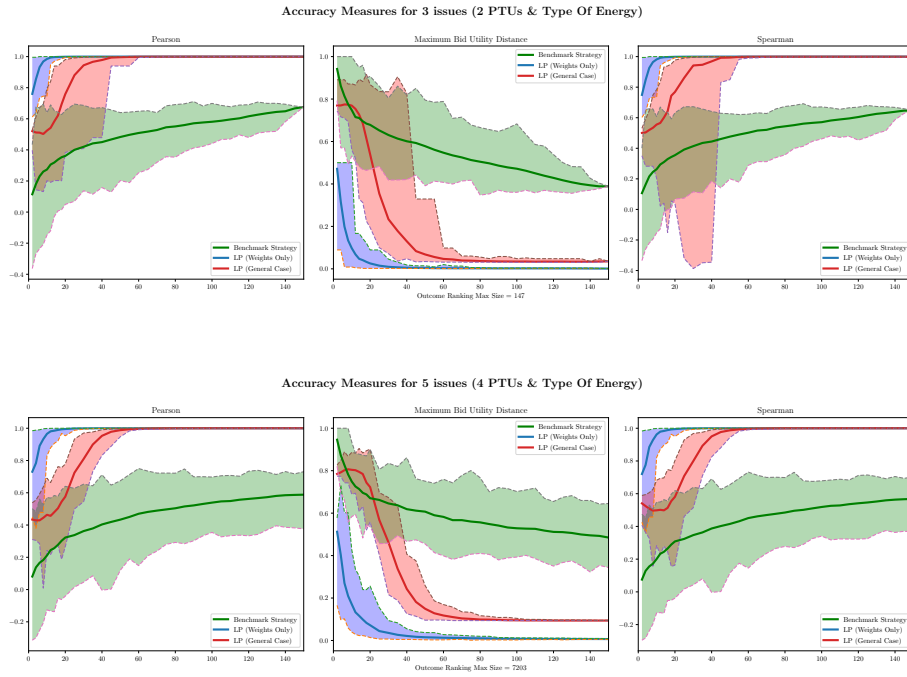
Fig. 2. Accuracy measures of simulations. The horizontal axis represents the size of the outcome ranking that is supplied to the agent and the vertical axis the different metric values.

Negotiation Performance: The accuracy results suggest that our models should perform well in an actual negotiation setting. To visualize the performance of our models in an actual negotiation, we visualize the Conceder agents traces for different amounts of comparisons for a whole negotiation session (180 rounds). That is, we observe the *target utilities* that are proposed at any given round of negotiation according to the conceder strategy. Figure 3 shows these traces for different amounts of preference information when following the Benchmark Strategy and our proposed Linear Programming Strategy as an estimation method. The superiority of the Linear Programming model becomes evident, as with a small amount of comparisons ($d \leq 50$) the real trace is followed almost perfectly, whereas the Benchmark model does not even for a much higher amount of comparisons ($d = 1000$). For 50 comparisons, the sum over all rounds of the average bid distances (averaged over 100 iterations) of the trace compared to the real trace was 2.564 following the Baseline strategy, compared to 0.120 following the Linear Programming strategy.

The negotiation results mirror the knowledge we obtained from the accuracy experiments. Given that at around 50 outcomes our method yields a Spearman's rank coefficient of $\rho \simeq 1$, the user model becomes fully accurate, hence the negotiation performance is not be affected. For the benchmark strategy on the
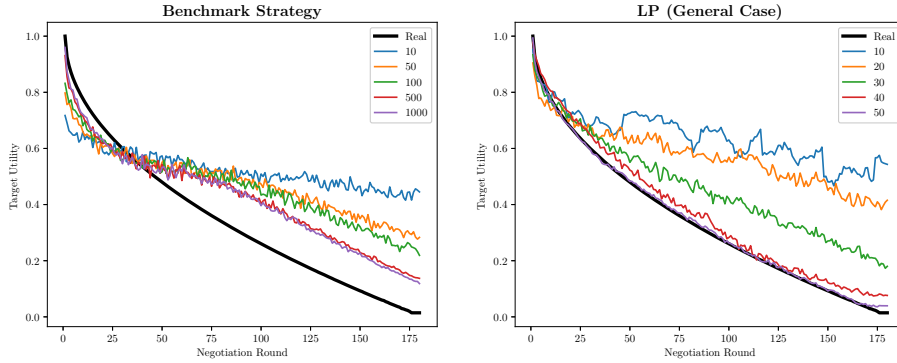
Fig. 3. Target utilities for a Conceder agent strategy under different levels of uncertainty using the *Benchmark Strategy* and the *Linear Programming Strategy*

other hand, the trace never converges to the real profile. This is explained by the fact that the benchmark method takes only the evaluator values into account during its reasoning and not the issues' importance (weights) at all. This can result at wrong issue weights, which in turn affects the performance negatively.

6   Conclusion

We presented an information model that is based on categorical data and has the form of a partial order of outcomes. This form of information may be readily elicited by asking a user to compare outcomes. Given this input, the agent needs to find the best sequence of queries that elicits as much preference information as possible while minimizing user bother.

We created a decision model that utilizes the pairwise comparison information, and through the use of linear programming we estimated a linear additive utility function, which the agent will use to negotiate. The proposed decision model was tested in terms of user-model accuracy and negotiation performance. The accuracy results showed that even with a very small numbers of comparisons (less than 1% of a total ranking of outcomes), the agent can reach high levels of accuracy (Spearman $\rho \simeq 1.0$). We achieved similar results in performance tests, with negotiation agreement utilities reaching no-uncertainty levels. Finally, our proposed Linear Programing model outperformed a baseline model both in terms of accuracy, as well as performance.

In a nutshell, we created a model that obtains incomplete ordinal preference data in the form of rankings and through the use of linear programming approximates the preferences of the user as best as possible. The results show that even given a few outcome comparisons (i.e user queries), together with the knowledge of the overall best bid, an agent can improve the user model accuracy significantly, and negotiates well as a result.

7   Discussion and future work

Computerized agents poised to represent users in negotiations should do so under incomplete information about the preferences of the users they represent. This work is a first step towards a reliable way of implementing such automated negotiation under preference uncertainty.

In this work, we present a method of representing incomplete user-preference information and a decision model that utilizes rankings of outcomes to elicit as much information as possible. We focus on bilateral negotiation, but pave the way for future work to examine how uncertainty affects an agent against multiple opponents. It is reasonable to expect that when facing more opponents the margin of error for an uncertain agent is smaller given that mistakes are more likely to be exploited.

Furthermore, our model allows the incorporation of error in user feedback and is able to pinpoint inconsistencies in user judgment, which could prove useful in cases where the agent does not have enough knowledge about the domain or is not certain about the user fidelity. The effect of judgment error could also be further examined, e.g., on a total ranking of outcomes.

One last avenue for future work would be to test the model's performance in more heterogeneous scenarios. The results show small differences in the amount of comparisons needed for high user model accuracy levels for various domains sizes. This is explained by the fact that there is only a small increase in the number of linear utility function parameters relative to the size of the outcome space. This finding may change for real-life domains without the linear attributes that we assume in our model. Possible interdependencies between negotiation issues would require non-linear optimization techniques and new and more complex preference elicitation strategies.

Acknowledgment

References

1. Aydoğan, R., Baarslag, T., Hindriks, K.V., Jonker, C.M., Yolum, P.: Heuristics for using cp-nets in utility-based negotiation without knowing utilities. Knowledge and Information Systems **45**(2), 357–388 (Nov 2015). https://doi.org/10.1007/s10115-014-0798-z
2. Aydogan, R., Baarslag, T., Jonker, C.M., Fujita, K., Ito, T., Hadfi, R., Hayakawa, K.: A baseline for non-linear bilateral negotiations: the full results of the agents competing in ANAC 2014. In: Intelligent Computational Systems: a Multi-Disciplinary Perspective, pp. 1–25. Bentham Science (July 2016), https://eprints.soton.ac.uk/399235/

3. Aydoğan, R., Yolum, P.: Learning opponents preferences for effective negotiation: an approach based on concept learning. Autonomous Agents and Multi-Agent Systems **24**(1), 104–140 (2012)

4. Baarslag, T.: Exploring the Strategy Space of Negotiating Agents: A Framework for Bidding, Learning and Accepting in Automated Negotiation. Springer Theses: Recognizing Outstanding Ph.D. Research, Springer International Publishing (2016). https://doi.org/10.1007/978-3-319-28243-5

5. Baarslag, T., Fujita, K., Gerding, E.H., Hindriks, K., Ito, T., Jennings, N.R., Jonker, C., Kraus, S., Lin, R., Robu, V., Williams, C.R.: Evaluating practical negotiating agents: Results and analysis of the 2011 international competition. Artificial Intelligence **198**, 73 – 103 (2013). https://doi.org/https://doi.org/10.1016/j.artint.2012.09.004

6. Baarslag, T., Gerding, E.H.: Optimal incremental preference elicitation during negotiation. In: Proceedings of the Twenty-fourth International Joint Conference on Artificial Intelligence. pp. 3–9. IJCAI'15, AAAI Press (2015), http://dl.acm.org/citation.cfm?id=2832249.2832250

7. Baarslag, T., Hendrikx, M.J.C., Hindriks, K.V., Jonker, C.M.: Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques. Autonomous Agents and Multi-Agent Systems **30**(5), 849–898 (Sep 2016). https://doi.org/10.1007/s10458-015-9309-1

8. Baarslag, T., Kaisers, M.: The value of information in automated negotiation: A decision model for eliciting user preferences. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems. pp. 391–400. AAMAS '17, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2017), http://dl.acm.org/citation.cfm?id=3091125.3091185

9. Baarslag, T., Kaisers, M., Gerding, E.H., Jonker, C.M., Gratch, J.: Computers that negotiate on our behalf: Major challenges for self-sufficient, self-directed, and interdependent negotiating agents. In: Sukthankar, G., Rodriguez-Aguilar, J.A. (eds.) Autonomous Agents and Multiagent Systems. pp. 143–163. Springer International Publishing, Cham (2017)

10. Boutilier, C., Brafman, R.I., Domshlak, C., Hoos, H.H., Poole, D.: CP-nets: A Tool for Representing and Reasoning withConditional Ceteris Paribus Preference Statements. ArXiv e-prints (Jun 2011)

11. Cornelio, C., Goldsmith, J., Mattei, N., Rossi, F., Venable, K.B.: Updates and uncertainty in cp-nets. In: Cranefield, S., Nayak, A. (eds.) AI 2013: Advances in Artificial Intelligence. pp. 301–312. Springer International Publishing, Cham (2013)

12. Fatima, S.S., Wooldridge, M., Jennings, N.R.: Optimal negotiation strategies for agents with incomplete information. In: Revised Papers from the 8th International Workshop on Intelligent Agents VIII. pp. 377–392. ATAL '01, Springer-Verlag, London, UK, UK (2002), http://dl.acm.org/citation.cfm?id=648208.757345

13. Fatima, S.S., Wooldridge, M., Jennings, N.R.: Multi-issue negotiation under time constraints. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1. pp. 143–150. AAMAS '02, ACM, New York, NY, USA (2002). https://doi.org/10.1145/544741.544775

14. Greco, S., Kadziski, M., Mousseau, V., Sowiski, R.: Robust ordinal regression for multiple criteria group decision: Utagms-group and utadisgms-group. Decision Support Systems **52**(3), 549 – 561 (2012). https://doi.org/https://doi.org/10.1016/j.dss.2011.10.005

15. Ito, T., Klein, M., Hattori, H.: A multi-issue negotiation protocol among agents with nonlinear utility functions. Multiagent and Grid Systems **4**(1), 67–83 (2008)

16. Jacquet-Lagreze, E., Siskos, J.: Assessing a set of additive utility functions for multicriteria decision-making, the uta method. European Journal of Operational Research **10**(2), 151 – 164 (1982). https://doi.org/https://doi.org/10.1016/0377-2217(82)90155-2

17. Jonker, C.M., Robu, V., Treur, J.: An agent architecture for multi-attribute negotiation using incomplete preference information. Autonomous Agents and Multi-Agent Systems **15**(2), 221–252 (Oct 2007). https://doi.org/10.1007/s10458-006-9009-y

18. Keeney, R., Raiffa, H.: Decisions with Multiple Objectives: Preferences and Value Trade-Offs. Wiley series in probability and mathematical statistics. Applied probability and statistics, Cambridge University Press (1993), https://books.google.nl/books?id=GPE6ZAqGrnoC

19. Marsa-Maestre, I., Lopez-Carmona, M.A., Velasco, J.R., Ito, T., Klein, M., Fujita, K.: Balancing utility and deal probability for auction-based negotiations in highly nonlinear utility spaces. In: IJCAI. vol. 9, pp. 214–219 (2009)

20. Mohammad, Y., Nakadai, S.: Fastvoi: Efficient utility elicitation during negotiations. In: International Conference on Principles and Practice of Multi-Agent Systems. pp. 560–567. Springer (2018)

21. Nguyen, D.V.: Global maximization of uta functions in multi-objective optimization. European Journal of Operational Research **228**(2), 397 – 404 (2013). https://doi.org/https://doi.org/10.1016/j.ejor.2012.06.022

22. Roszkowska, E.: The application of UTA method for support evaluation negotiation offers. Optimum. Studia Ekonomiczne (2(80)), 144–162 (2016). https://doi.org/10.15290/ose.2016.02.80.11, https://doi.org/10.15290/ose.2016.02.80.11

23. Sanchez-Anguix, V., Aydoğan, R., Baarslag, T., Jonker, C.M.: Can we reach pareto optimal outcomes using bottom-up approaches? In: Aydoğan, R., Baarslag, T., Gerding, E., Jonker, C.M., Julian, V., Sanchez-Anguix, V. (eds.) Conflict Resolution in Decision Making. pp. 19–35. Springer International Publishing, Cham (2017)

24. Srinivasan, V., Shocker, A.D.: Estimating the weights for multiple attributes in a composite criterion using pairwise judgments. Psychometrika **38**(4), 473–493 (Dec 1973). https://doi.org/10.1007/BF02291490