Automated Negotiation with Gaussian Process-based Utility Models*

Haralambie Leahu , **Michael Kaisers** and **Tim Baarslag**
Centrum Wiskunde & Informatica, Amsterdam, The Netherlands
{H. Leahu, M. Kaisers, T. Baarslag}@cwi.nl

Abstract

Designing agents that can efficiently learn and integrate user's preferences into decision making processes is a key challenge in automated negotiation. While accurate knowledge of user preferences is highly desirable, eliciting the necessary information might be rather costly, since frequent user interactions may cause inconvenience. Therefore, efficient elicitation strategies (minimizing elicitation costs) for inferring relevant information are critical. We introduce a stochastic, inverse-ranking utility model compatible with the Gaussian Process preference learning framework and integrate it into a (belief) Markov Decision Process paradigm which formalizes automated negotiation processes with incomplete information. Our utility model, which naturally maps ordinal preferences (inferred from the user) into (random) utility values (with the randomness reflecting the underlying uncertainty), provides the basic quantitative modeling ingredient for automated (agent-based) negotiation.

1 Introduction

Automated (agent-based) negotiation formalizes a wide range of interactions in multi-agent systems, covering topics such as HF trading [McGroarty *et al.*, 2018], cloud computing [Sim, 2011], pervasive computing [Ramchurn *et al.*, 2004], smart grids [Ketter *et al.*, 2018], supply chain management [Wang *et al.*, 2009]. In such systems, agents can successfully substitute humans in making complex decisions, provided that they have good knowledge of the user's goals [Kraus *et al.*, 1995].

In many situations agents do not have access to all information required for taking optimal decisions. For instance, if negotiation takes place over multiple issues then quantifying the desirability of various outcomes requires the consideration of trade-offs. The combinatorial explosion in the number of potential outcomes makes it intractable (too costly) for a user to determine and communicate all dependencies beforehand, while many of them could be also irrelevant since, for instance, they could be highly unattractive for the other negotiating party (which, in the sequel, will be called *opponent*).

An important challenge in automated negotiation is designing agents which can efficiently strike a balance between negotiation and (user's) preference elicitation [Baarslag *et al.*, 2017]. (PO)MDP models have been employed for both negotiation [Paruchuri *et al.*, 2009] and preference elicitation [Boutilier, 2002; Chajewska *et al.*, 2000], while preference elicitation models were further adapted to negotiation processes in which agents may elicit utility values by submitting queries to the user [Baarslag and Gerding, 2015; Mohammad and Nakadai, 2018]. In this paper, which extends the framework presented in [Leahu *et al.*, 2019], we adopt a similar modeling paradigm, but have agents learn from comparative queries; thus not asking the user to quantify the desirability of a specific outcome, but to compare a pair of alternatives. Preference quantification is rarely available in practice - and even then may be inconsistent over time - [Kingsley and Brown, 2006], which motivates our choice of ordinal utility models, i.e. based on pair-wise comparisons.

A key ingredient in our modeling paradigm is defining suitable *utility models*, quantifying the preferences of both the user and the opponent for every possible negotiation outcome by means of (random) utility functions, with the randomness illustrating the underlying uncertainty. That is, in the absence of (full) knowledge of the user/opponent's preferences, the agent maintains quantitative beliefs over them, which are being updated, as new data emerges, through interactions with the user/opponent. Therefore, it is highly desirable that a utility model should be *general* (to account for a wide range of scenarios) and *flexible* (to allow easy integration of new data).

In this paper we propose a stochastic utility model, where the underlying randomness is formalized by a Gaussian Process (GP), which is integrated into a MDP negotiation framework. The use of GP's in Machine Learning has gained much popularity [Rasmussen and Williams, 2006] since GP's allow for a rather general treatment of random functions, compared to other models (e.g. linear with random slope coefficient). Moreover, the Laplace approximation of the Bayesian posterior distribution makes GP-based models flexible, as new data can be easily integrated by means of GP parameter updates. While the use of the GP's in formalizing uncertain preferences is well established [Chu and Ghahramani, 2005a; Chu and Ghahramani, 2005b], standard utility models (based on uncertain beliefs) for negotiating agents are still lacking. Filling this gap is one of the main contributions of this paper.

*An abridged version of this work was presented at AAMAS '19.

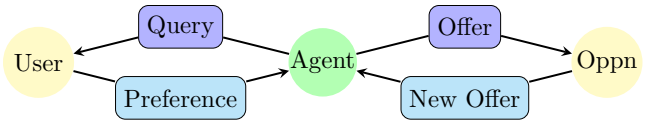

Figure 1: Agent interactions in preference learning and negotiation.

This paper furthers the state-of-the-art in the following directions: First, it extends the GP-based preference learning model [Chu and Ghahramani, 2005b] by allowing ties between alternatives; this is in contrast with the standard model which assumes that the user can *always* indicate a clear preference between *any* pair of alternatives. Secondly, we adapt the model to online learning, by deriving explicit sequential update rules which facilitate integration of new data in real time. Ultimately, we introduce a stochastic, *inverse-rank* utility model, mapping preferences into utility values in a natural way. The resulting utility mapping is a monotone transformation of a GP-like preference function (thus obeys essentially the same update rules), but unlike the preference function (which introduces subjective beliefs which can not be validated by user interactions) is data measurable, i.e. the randomness is completely (up to approximation errors) removed when full information on user preferences becomes available; this distinguishes our model from alternative GP-based utility models in preference elicitation [Bonilla *et al.*, 2010].

We further integrate our utility model into a two-objective (belief) MDP model which combines negotiation actions with preference elicitation, resulting in a fairly general mathematical framework for automated negotiation with uncertain knowledge. Namely, a negotiating agent employs utility models (called henceforth *user*, resp. *opponent* model) for both parties involved in the negotiation process and performs successive interactions with the user (by means of comparative queries, incurring certain elicitation costs) and the opponent (deal offers), updating the two utility models accordingly; see Figure 1. While the user utility can be readily used to define rewards, the use of the opponent utility in formalizing its behavior requires, in general, further modeling assumptions.

To illustrate the use of our utility model, we consider a negotiating agent which, before taking any negotiation action (i.e. accepting or making a new offer), explores the user's preferences in order to reduce uncertainty around the utility values corresponding to the outcomes relevant for the negotiation. For a better insight into the decision mechanism, we choose a small-scale example for our numerical experiments.

Since our focus is on preference elicitation (rather than on negotiation), we shall assume that the agent employs a one-step look ahead negotiation strategy, thus aiming to maximize his immediate (one-step) expected reward. Our setup is somewhat similar to the ultimatum game [Zhong *et al.*, 2002], which can be seen as the one-shot, final-offer version of the classic alternating offers protocol [Baarslag *et al.*, 2015]. However, in our setting, the agents can have mutually non-exclusive objectives since arbitrary utilities over the outcome space are possible (e.g. single-peaked preferences), allowing for integrative negotiation scenarios where win-win agreements are possible [Kowalczyk and Bui, 2000].

2 Problem Formulation

An agent conducts a negotiation on some user's behalf, in a sequential way, based on an offer/counter-offer protocol. Negotiation ends when either party accepts the other party's offer, resulting in some outcome-dependent rewards for each party. The space of all possible outcomes of the negotiation is denoted by $\mathbb{X}$ and is assumed that, initially, an agreement-offer $y \in \mathbb{X}$ (e.g. an outside option) is available to the agent.

In the absence of complete information, the agent maintains a belief μ formalizing the user's utility and a belief η governing the opponent's behavior. A first challenge for the agent is to decide whether to accept y, make an offer $x \neq y$, or elicit some additional information (i.e. making a sequence of queries) from the user, in order to obtain a more accurate μ-belief, before taking any decision. Once the agent decides to negotiate (i.e. to make an offer $x \neq y$), it can be either that the offer gets accepted, with some *acceptance* probability (calculated based on the current η-belief), yielding a suitable reward (the expected value of which is calculated based on the current μ-belief), or the offer is further negotiated by the opponent, resulting in a new agreement-offer z and a suitable updated opponent belief. Alternatively, at any time, the agent may submit a user-query to update its (user) belief. In this context, the problem we tackle is devising a judicious rule for deciding whether to elicit more information (and, if yes, which information) before continuing negotiation.

In the above paradigm, negotiation actions alternate with preference elicitation actions (queries). By an *elicitation policy* we mean a sequence $\pi = (q_1, \dots, q_n)$, with $n \geq 0$, of queries made during an *elicitation cycle*, i.e. between two successive negotiation actions. To each query, a 'bother' cost $\gamma(q)$, in the form of a discount factor, is associated. Since our focus is on learning preferences (rather than elaborating optimal negotiation strategies, which requires a complex opponent behavior modeling and assumptions), we aim to derive efficient elicitation policies maximizing the immediate expected reward $\mathcal{R}(x|y,\mu,\eta)$ obtained by the agent by making offer x in state (y,μ,η) (to be understood as 'accept y' if $x = y$). To be more specific, we define

$$\mathcal{R}(x|y,\mu,\eta) := \begin{cases} \mathbb{E}_\mu[\mathcal{U}(y)] & x = y; \\ \mathbb{E}_{\mu,\eta}[\mathcal{U}(x)\mathcal{P}(x)] & x \neq y, \end{cases} \tag{1}$$

where $\mathcal{U}$ denotes the user's utility function under belief μ and $\mathcal{P}$ denotes the opponent's acceptance probability under belief η. We further define the maximal immediate expected reward

$$\mathcal{R}(y,\mu,\eta) := \max_x \mathcal{R}(x|y,\mu,\eta). \tag{2}$$

Finally, defining the state-value function

$$\mathcal{V}(y,\mu,\eta) := \max_\pi \gamma[\pi] \cdot \mathbb{E}[\mathcal{R}(y,[\mu|\pi],\eta)], \tag{3}$$

where $\gamma[\pi] := \gamma(q_1) \cdot \ldots \cdot \gamma(q_n)$ denotes the cost of the policy $\pi = (q_1, \dots, q_n)$ and $[\mu|\pi]$ denotes the random belief obtained by updating μ w.r.t. the (predicted) outcomes of π, an *optimal* elicitation policy is any maximizer in (3). An optimal elicitation cycle ends when a state (y,μ,η), satisfying

$$\mathcal{V}(y,\mu,\eta) = \mathcal{R}(y,\mu,\eta),$$

is reached and an offer maximizing (1) is made; note the the above equality can be regarded as a stopping condition.

3 Stochastic Utility Models

A *utility belief* on $\mathbb{X}$ is a Gaussian probability law μ on $\mathbb{R}^{\mathbb{X}}$, specified by a mean and a covariance function, $\boldsymbol{\mu} : \mathbb{X} \longrightarrow \mathbb{R}$ resp. $\boldsymbol{k} : \mathbb{X}^2 \longrightarrow \mathbb{R}$. A sample from μ is a random function $G : \mathbb{X} \longrightarrow \mathbb{R}$, which will be called a Gaussian Process (GP). Utility beliefs are used to define utility models (random utility functions) which are updated based on pairwise comparisons.

3.1 Utility Belief Update

In *instance preference learning* [Chu and Ghahramani, 2005b], data is available in the form

$(+)$ $u \prec v$, meaning that "v is (strictly) preferred to u";

$(-)$ $u \succ v$, meaning that "u is (strictly) preferred to v";

$(\sim)$ $u \sim v$, meaning "no preference between u and v",

for some arbitrary pair $q := (u, v) \in \mathbb{X}^2$, with $u \neq v$.

Given a utility belief μ (specified by parameters $\boldsymbol{\mu}$ and $\boldsymbol{k}$) and some outcome $\epsilon = u \bowtie v$, where $\bowtie \in \{\prec, \sim, \succ\}$, of the pair-wise comparison (u, v), we denote by $[\mu|\epsilon]$ the updated belief, which we define as the Laplace approximation of the Bayesian posterior of μ, conditioned on ϵ. To derive the mean/covariation functions $\bar{\boldsymbol{\mu}}_\epsilon$, resp. $\bar{\boldsymbol{k}}_\epsilon$ of the updated belief $[\mu|\epsilon]$, we use an adaptation of the standard method [Chu and Ghahramani, 2005b] to our setup, i.e. we adjust the likelihood function to include ties between alternatives. We do so by introducing a new (precision) parameter $\delta \geq 0$ and interpreting the outcome $u \sim v$ as $|G(v) - G(u)| \leq \delta$.

For a pair (u, v), the likelihood $\mathbb{P}\{u \prec v|G\}$ is given by

$$\mathbb{P}\{u \prec v|G\} = \Phi\left(\frac{G(v) - G(u) - \delta}{\varsigma}\right),$$

where Φ denotes the standard Gaussian c.d.f. and ς is a model parameter accounting for the uncertainty in the user answer; note that, for $\delta = 0$ one recovers the standard model in [Chu and Ghahramani, 2005b]. Letting further

$$M_q := \boldsymbol{\mu}(v) - \boldsymbol{\mu}(u),\ V_q := \boldsymbol{k}(v, v) - 2\boldsymbol{k}(u, v) + \boldsymbol{k}(u, u),$$

the predictive distribution of the query outcomes is given by

$$\mathbb{P}\{(\pm)\} = \Phi\left(\frac{\pm M_q - \delta}{\sqrt{V_q + \varsigma^2}}\right); \mathbb{P}\{(\sim)\} = 1 - \mathbb{P}\{(+)\} - \mathbb{P}\{(-)\}. \quad (4)$$

Let $f(b) := \log \Phi\left((b - \delta)/\varsigma\right)$, for $b \in \mathbb{R}$, and $\beta(M, V)$, for $M \in \mathbb{R}$, $V \geq 0$, denote the unique solution of

$$\beta = M + V \cdot f'(\beta). \quad (5)$$

Then the updated belief $[\mu|\epsilon]$ is specified by

$$\bar{\boldsymbol{\mu}}_\epsilon(x) := \boldsymbol{\mu}(x) + B_\epsilon\left(\boldsymbol{k}(x, v) - \boldsymbol{k}(x, u)\right), \quad (6)$$

respectively

$$\bar{\boldsymbol{k}}_\epsilon(x, z) := \boldsymbol{k}(x, z) - \frac{C_\epsilon(\boldsymbol{k}(x, v) - \boldsymbol{k}(x, u))(\boldsymbol{k}(z, v) - \boldsymbol{k}(z, u))}{1 + C_\epsilon V_q}, \quad (7)$$

where $B_\epsilon \in \mathbb{R}$ and $C_\epsilon > 0$ are defined as follows:

- for $\epsilon = (\pm)$ we have
  $B_\epsilon = \pm f'\left(\beta(\pm M_q, V_q)\right),\ C_\epsilon = -f''\left(\beta(\pm M_q, V_q)\right);$
- for $\epsilon = (\sim)$ we have
  $$B_\epsilon = -\frac{M_q}{\varsigma^2 + V_q},\ C_\epsilon = \frac{1}{\varsigma^2}.$$

This concludes the belief update rules.

3.2 Ranking-based Utility Models

A *utility model* with underlying utility belief formalized by a GP G, is a random (utility) function $\mathcal{U} : \mathbb{X} \longrightarrow \mathbb{R}$ satisfying $\mathcal{U}(x) \leq \mathcal{U}(z)$ if and only if $G(x) \leq G(z)$. The monotonicity assumption ensures that ordinal data can be properly integrated in the belief update process; see Section 3.1.

In this paper, we propose the *inverse-rank* utility model

$$\mathcal{U}(x) := \sum_{z \in \mathbb{X}} \mathbf{1}\{G(z) \leq G(x)\}; \quad (8)$$

that is, $\mathcal{U}(x)$ denotes the (random) number of negotiation outcomes whose G-value does not exceed that of x. $\mathcal{U}(x)$ is an integer between 1 and $\aleph := \#\mathbb{X}$, representing the ranking of $G(x)$ within the increasing sequence of G-values, i.e.

$$\mathcal{U}\left(\arg\min_x G(x)\right) = 1,\ \mathcal{U}\left(\arg\max_x G(x)\right) = \aleph.$$

Next to its intuitive domain interpretation (utility as *number of outcomes that are not preferred*), the inverse-rank utility model (8) has two attractive formal properties, which set it apart from alternative models, e.g. $\mathcal{U} = G$ or $\mathcal{U} = \exp(G)$. First, the utility values remain in some fixed interval. Secondly, our model is *data-measurable*, in the sense that is built from *observable* variables only; put differently, a full knowledge of user's preferences makes the model deterministic.

Furthermore, under the utility belief $\mu = (\boldsymbol{\mu}, \boldsymbol{k})$, the expected utility of some particular outcome x will be given by

$$\mathbb{E}_\mu[\mathcal{U}(x)] = \sum_{z \in \mathbb{X}} \Phi\left(\frac{\boldsymbol{\mu}(x) - \boldsymbol{\mu}(z)}{\sqrt{\boldsymbol{k}(x, x) - 2\boldsymbol{k}(x, z) + \boldsymbol{k}(z, z)}}\right); \quad (9)$$

note, however, that the expected utility values are not integers anymore, but still remain in the (continuous) range $[0, \aleph]$.

In the context of negotiation, assume that the agent has an offer x on the table, which is accepted with probability $\mathcal{P}(x)$, or negotiated (hoping for a better deal) with remaining probability. Provided that the agent has no insight into the future negotiation process, a future agreement can be regarded as a uniform sample from the negotiation space; thus the chance of not improving the current deal x is $\mathcal{U}(x)/\aleph$, which provides a natural model for the acceptance probability $\mathcal{P}(x)$ (if interpreted as the probability of *not* getting a better deal).

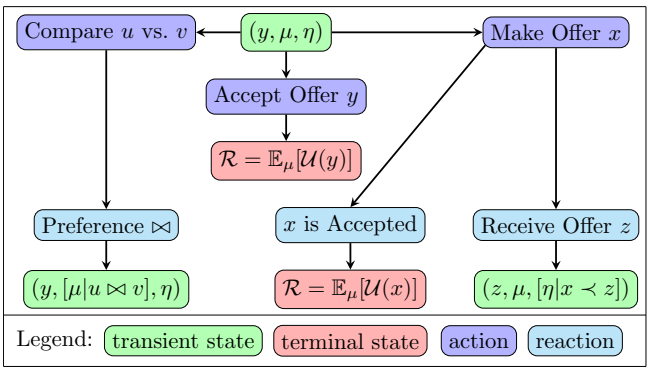

Figure 2: The belief MDP dynamics.

4 Belief MDP with Stochastic Utility Models

In the context of Section 2, let $\mu = (\boldsymbol{\mu}, \boldsymbol{k})$, $\eta = (\boldsymbol{\eta}, \boldsymbol{\ell})$ denote user's, resp. opponent's, utility beliefs; we denote by G, resp. H, the corresponding GP-samples. In particular, for a specific agreement $x \in \mathbb{X}$, the user's utility is given by

$$\mathcal{U}(x) = \sum_{s \in \mathbb{X}} \mathbf{1}\{G(s) \leq G(z)\},$$

and the corresponding opponent's acceptance probability by

$$\mathcal{P}(x) \approx \sum_{s \in \mathbb{X}} \mathbf{1}\{H(s) \leq H(z)\},$$

where $\approx$ means equality up to some multiplicative constant.

Assuming stochastic independence between the user's and the opponent's utility beliefs, μ and η, it follows that

$$\mathbb{E}_{\mu,\eta}[\mathcal{U}(x)\mathcal{P}(x)] = \mathbb{E}_{\mu}[\mathcal{U}(x)]\, \mathbb{E}_{\eta}[\mathcal{P}(x)],$$

where, in accordance with (9), it holds that

$$\mathbb{E}_{\mu}[\mathcal{U}(x)] = \sum_{s \in \mathbb{X}} \Phi\left(\frac{\boldsymbol{\mu}(x) - \boldsymbol{\mu}(s)}{\sqrt{\boldsymbol{k}(x,x) - 2\boldsymbol{k}(x,s) + \boldsymbol{k}(s,s)}} \right),$$

respectively

$$\mathbb{E}_{\eta}[\mathcal{P}(x)] \approx \sum_{s \in \mathbb{X}} \Phi\left(\frac{\boldsymbol{\eta}(x) - \boldsymbol{\eta}(s)}{\sqrt{\boldsymbol{\ell}(x,x) - 2\boldsymbol{\ell}(x,s) + \boldsymbol{\ell}(s,s)}} \right).$$

The above formulas express the numerical elements required in (1), (2), (3) by means of the agent's current beliefs.

To describe the MDP dynamics, we first note that the MDP states can be classified into *transient* and *terminal* states, with rewards being obtained only upon reaching a terminal state. Furthermore, given a transient state (y, μ, η), the set of available actions consists of all possible user queries, formalized as pairs $q = (u, v)$, and negotiation actions (offers) $x \in \mathbb{X}$ (recall that y means 'accept'). Making a query $q = (u, v)$ results in an outcome ϵ, which can be either $(+)$ $u \prec v$, $(-)$ $u \succ v$ or $(\sim)$ $u \sim v$, having predictive distribution specified by (4). The agent moves in the new transient state $(y, [\mu|\epsilon], \eta)$, where $[\mu|\epsilon]$ is specified by (6)–(7). On the other hand, the agent can either accept y, expecting the reward $\mathcal{U}(y)$, or make the offer $x \neq y$ expecting one of the following scenarios:

(i) x is accepted by the opponent, with probability $\mathcal{P}(x)$, resulting in a reward $\mathcal{U}(x)$;

(ii) x is negotiated by the opponent, resulting in the agent receiving a (counter-) offer $z \neq x$.

Acceptance (by either agent or opponent) results in terminal states, where belief updates are not necessary. On the other hand, should the opponent decide to negotiate the agent's offer - case (ii) - the agent moves into the new (transient) state $(z, \mu, [\eta|x \prec z])$, inferring that z is preferred (by the opponent) to x. The belief MDP dynamics are illustrated in Figure 2, which is the particularization of Figure 1.

Remark: *The probability of receiving some counter-offer z depends on the opponent's reasoning and requires further modeling assumptions for the agent. However, for the problem formulated in Section 2, this probability (of the opponent making a specific counter-offer) is not relevant, as the agent does not optimize his actions w.r.t. future opponent offers; it is only the acceptance probability of x which matters.*

Algorithm 1 decides between elicitation and negotiation

Input: $[(y, \mu, \eta); \mathfrak{L}]$
Parameter: query cost-function γ
Output: myopic action "Action"

```
ϱ ← R(y, μ, η); // reward based on current belief
Act ← arg max_x R(x|y, μ, η);
for {any relevant query q ∉ 𝔏} do
    if {ϱ < Q(y, μ, η; q)} then
        ϱ ← Q(y, μ, η; q);
        Act ← q; // make q a candidate for the next query
    end if
end for
return Act
```

5 The Myopic Elicitation Strategy

We devise an algorithm that determines an optimal elicitation cycle w.r.t. a myopic look ahead strategy. Namely, the queries are decided in a sequential manner, by evaluating at each step whether to negotiate (accept y or make an offer) or make a new query, which, by reducing the utility uncertainty around the points of interest, could possibly increase the immediate expected reward obtained by future negotiation actions.

Formally, our algorithm evaluates

$$\mathcal{Q}(y, \mu, \eta; q) := \gamma(q)\, \mathbb{E}[\mathcal{R}(y, [\mu|\epsilon], \eta)], \tag{10}$$

for all queries $q := (u, v)$ and outcomes $\epsilon = u \bowtie v$ of q, where the expectation is calculated w.r.t. the predictive distribution (4) of the random preference $\bowtie = \prec, \sim, \succ$, based on the current μ-belief. Furthermore, it decides as follows:

- if the stopping condition

$$\max_q \mathcal{Q}(y, \mu, \eta; q) \leq \mathcal{R}(y, \mu, \eta), \tag{11}$$

  is fulfilled, then the elicitation cycle ends with the offer

$$x^*(y, \mu, \eta) = \arg\max_x\ \mathcal{R}(x|y, \mu, \eta); \tag{12}$$

- else, a query $q^* := \arg\max_q \mathcal{Q}(y, \mu, \eta; q)$ is made.

Algorithm 1 illustrates the myopic (one-step ahead) exploration of all relevant (in some sense to be specified) queries and decides whether extra elicitation is profitable; if yes, it also returns an optimal query. To reduce the complexity of the exploration process, the agent maintains a list $\mathfrak{L}$ of pairwise preferences based on the information elicited so far, i.e. preferences either confirmed directly by the user (as answers to queries) or inferred from known preferences by transitivity (assuming user's preference consistency). The queries which do not have a certain answer based on the ordinal data in $\mathfrak{L}$ are called *relevant* and the list is being updated after each answered query, to ensure that the search space for new relevant queries decreases after each user interaction.

By iterating the decision rule formalized by Algorithm 1 until condition (11) is fulfilled, one obtains a *myopic* elicitation cycle; see Algorithm 2. Note that, restricting the maximization to relevant queries (only) ensures convergence.

Remark: *Our algorithm performs only a one-step ahead search, corresponding to maximizing in (3) w.r.t. policies (interpreted as sequences of queries) of length $n = 1$. Possibly more accurate solutions can be obtained by extending the optimization range to multiple-step search.*

Algorithm 2 generates a myopic elicitation cycle

Input: $[(y, \mu, \eta); \mathfrak{L}]$
Parameter: query cost-function γ
Output: the next negotiation action

Act $\leftarrow$ Query.
while {Act = Query} **do**
  if {Action$[(y, \mu, \eta); \mathfrak{L}]$ $\neq$ Offer} **then**
    $q \leftarrow$ Action$[(y, \mu, \eta); \mathfrak{L}]$
    make query q and obtain outcome ϵ
    $\mu \leftarrow [\mu|\epsilon]$; // update belief μ cf. (6) and (7)
    $\mathfrak{L} \leftarrow [\mathfrak{L}; \epsilon]$; // update the list $\mathfrak{L}$
  else
    Act $\leftarrow$ Offer;
  end if
end while
return $x^*(y, \mu, \eta)$ given by (12);

6 Experimental Setup

To illustrate our approach, we perform an experimental comparison of the myopic elicitation strategy formalized by Algorithm 1 against a 'randomized' elicitation strategy, in which the agent randomly generates a sequence of relevant queries and then makes the optimal negotiation offer (12), based on the resulting updated belief (given the corresponding query outcomes). We shall achieve that by simulating 'parallel' negotiation processes, in which the two elicitation strategies are confronted with the same opponent behavior.

To implement our experimental setup, we assume the existence of 'ground truth' utility functions ϕ and ψ, quantifying the preferences for the agent, resp. opponent, and perform simulations, based on the following assumptions:

- the answer to a query (u, v) is generated in accordance with the utility ϕ, by comparing $\phi(u)$ with $\phi(v)$;
- upon receiving an offer x, the opponent either accepts it, with a probability proportional to $\psi(x)$, or negotiates the offer, by making a random counter-offer z, satisfying $\psi(x) < \psi(z)$, with probability proportional to $\psi(z)$, i.e.

$$\mathbb{P}\{\text{counter-offer} = z | \text{offer} = x\} = \frac{\psi(z)\, \mathbf{1}\{x \prec z\}}{\sum_s \psi(s)\, \mathbf{1}\{x \prec s\}},$$

  where $\prec$ denotes the opponent's preference ordering.

Remark: *The functions ϕ and ψ satisfy the equation*

$$f(x) = 1 + \sum_{z \neq x} \mathbf{1}\{f(z) < f(x)\} + \frac{1}{2} \sum_{z \neq x} \mathbf{1}\{f(x) = f(z)\};$$

that is, both can be recovered from a full set of ordinal data.

For the 'random' strategy, we use a parameter $p \in (0, 1)$ formalizing the probability of making a new query. The agent generates an elicitation cycle, as follows: at each step decides with probability p to make a new (relevant) query and with probability $(1-p)$ to stop elicitation and take the best negotiation action (12) based on the current belief; should it decide to elicit more information, it will randomly select a new query from the remaining relevant ones.

Remark: *While myopic elicitation (described in Section 5) is, essentially, an 'exploitation' strategy, random elicitation can be regarded as a fully 'exploration' strategy.*

x	0	1	2	3	4	5	6	7	8	9
$\phi(x)$	1	2.5	4	9	7	6	8	10	5	2.5
$\psi(x)$	1	2	4	6	8	10	9	7	5	3

Table 1: Ground truth utility functions

Finally, we assume that all queries are equally costly, with each query inducing a discount $\gamma \in (0, 1]$ on the final reward and we compare the two strategies with respect to the 'true' expected reward (discounted by the total elicitation costs); that is, for each of the two scenarios we calculate the expected discounted (true) utility $\mathbb{E}[\gamma^\nu \phi(\omega)]$ obtained by the agent, where ω denotes the negotiation outcome, ν denotes the total number of queries made during the negotiation process and the expectation accounts for the random opponent actions (and for the randomly generated elicitation cycles).

7 Tables and Numerical Results

For our experiments, we choose a negotiation space consisting of 10 items/options, labeled $0, 1, \ldots, 9$ and use the ordinal utility functions ϕ (for the user) and ψ (for the opponent) displayed in Table 1. The agent starts the negotiation, having the outside option $y = 0$ and 'flat' utility beliefs over both utility functions, i.e. GP's with $\boldsymbol{\mu}, \boldsymbol{\eta} = \mathbf{0}$. In addition, we assume a distance-based correlation structure $\boldsymbol{k}(x, z) = \kappa(|x - z|)$, with $\kappa(0) = 0.5$, $\kappa(1) = 0.3$, $\kappa(2) = 0.2$, $\kappa(3) = 0.1$ and $\kappa = 0$ otherwise; that is, closer items are stronger (positively) correlated. Such a model is appropriate for negotiations over items which can be organized in a spectrum, e.g. colors.

Our numerical experiments are summarized in Table 2, illustrating the dependence, w.r.t. the query cost-factor γ, of the (expected) discounted utility "Util" and the number of queries "Que", for both the myopic and random elicitation approaches, based on averages over 100 (myopic), respectively 400 (random), simulations of the negotiation process.

We note that the myopic approach clearly outperforms the random one in terms of discounted expected utility, scoring higher for any cost-factor γ. A graphical head-to-head comparison of the two strategies is provided in Figure 4.

Remark: *The number of simulation runs based on which the metrics of interest are calculated does only influence the accuracy (variance) of the estimates in Table 2; thus the different numbers of simulation runs for the two approaches do not induce any bias in the above head-to-head comparison.*

In the myopic approach, both the expected discounted utility and the query number are increasing in the cost-factor γ, as expected, reaching the maximal values of 7.66, resp. 10.03, in the cost-free scenario $\gamma = 1$; see Figure 3. The intuition is that, as the elicitation cost decreases (γ increases), the agent will elicit (on average) more information at the same costs, learning more accurately the user's preferences, which will result in better negotiation outcomes. On the other hand, in the random approach, the discounted utility is still monotone, but the number of queries is (roughly) constant since, in this case, the elicitation strategy does not account for the cost-factor γ (but only on p), whereas the discount (still) does.

γ	97.0	97.5	98.0	98.5	99.0	99.5	100.0
Util	6.54	6.57	6.89	6.98	7.26	7.30	7.66
Que	1.82	2.38	4.27	4.60	6.35	8.26	10.03
Util	3.97	4.03	4.22	4.28	4.33	4.76	4.80
Que	4.39	4.62	4.20	4.40	4.72	4.12	4.10

Table 2: Summary of simulation results for the myopic (top rows) and random (bottom rows) elicitation strategies, for various γ's.

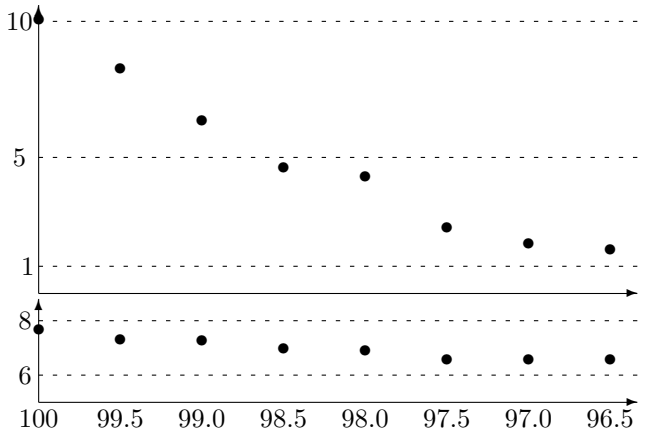

Figure 3: The expected number of queries (top), resp. the discounted utility (bottom) plotted with respect to the cost-factor γ (expressed as percentage). Dashed lines represent relevant performance levels.

Finally, note that, with complete knowledge of both user's and opponent's utilities, the myopic action for the agent is to make offer 6, maximizing $\mathcal{R}(x|0) = \phi(x)\psi(x)/10$. Since $\psi(6) = 9$, the opponent will accept it (with probability 0.9) or will propose item 5, since $\psi(5) = 10$. The myopic action for the agent is again 6 (which brings an immediate expected reward $\mathcal{R}(6|5) = 7.2 > 6 = \phi(5)$ and the agent, resp. the opponent, will keep repeating offers 6, resp. 5, until the opponent accepts item 6; this will happen quite fast, after an expected number of $1/9$ iterations. Therefore, under complete knowledge of ϕ and ψ, following the myopic negotiation strategy, the agent obtains (with probability 1) an agreement over item 6, having the utility value $\phi(6) = 8$, which is an upper bound on what it can achieve using the myopic negotiation strategy. The outcome 6 coincides, in this case, with the maximizer of the 'combined' utility $\phi \cdot \psi$, which has the following interpretation: it is the outcome which maximizes the number of pairs of negotiation outcomes which are simultaneously *not* better for the user and opponent, respectively.

Remark: *The gap between the cost-free expected utility* 7.66 *(obtained for* $\gamma = 1$*) and the upper limit* 8 *is motivated by the lack of information about the opponent's preferences. Indeed, although the agent can make an unlimited number of cost-free queries, presumably learning the exact user utility function, the lack of information on the opponent's utility function could end up in having sub-optimal offers accepted by the opponent, thus resulting in a lower (expected) utility.*

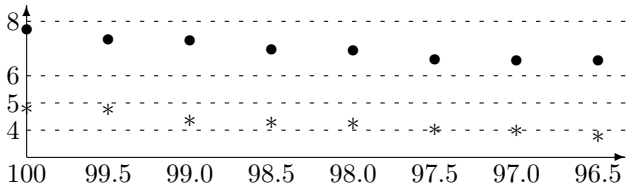

Figure 4: The expected discounted utility for the myopic strategy (•) vs. random strategy (∗). Dashed lines indicate relevant utility levels.

8 Conclusion and Future Research Directions

In this paper, we consider an agent negotiating on behalf of a user, using a GP-based adaptive model to formalize the uncertain information on both the user and the opponent's preferences. We formulate a myopically optimal elicitation algorithm that computes the best query to pose to the user in order to achieve a good negotiation outcome while minimizing elicitation costs. Through numerical experiments we find that our myopic elicitation strategy performs clearly better than a baseline strategy which randomly decides to pose queries.

Our algorithm performs a one-step look-ahead search to decide on the next negotiation action. Going beyond the myopic approach would require a more demanding optimization process, but also a more complex modeling paradigm, which, in order to account for the opponent's future actions, requires more than an acceptance probability (predictive) model.

While, for illustrative purposes, our numerical experiments are performed on a one-dimensional negotiation space with merely 10 items, our approach carries over to far more complex negotiation spaces. In terms of complexity, computation of (expected) utility is quadratic and is required after each belief update. For a more efficient approach, the computation can be formalized as a matrix-vector multiplication, for which parallelization methods offer attractive alternatives. Furthermore, our GP-based utility model can be adapted to multi-issue negotiation spaces, where monotonicity properties and various types of dependency between the issues and attributes (which typically reduce computational complexity) can be included through appropriate tuning of the covariance function.

Given our elicitation model of preference judgments, the queries are homogeneous (in type and cost structure). However, our model admits generalizations to different types of queries, with type and/or timing dependent costs.

Based on the above considerations, we are confident that the modelling paradigm proposed in this paper opens up a wide range of promising avenues for extending our approach to more complex (automated) negotiation frameworks characterized by (user) preference uncertainty.

Acknowledgments

This research has received funding from the ERA-Net Smart Energy Systems' focus initiative Smart Grids Plus, within the European Union's Horizon 2020 research and innovation programme, under grant No. 646039, and through the Veni research programme of the Dutch Organisation for Scientific Research (NWO), under the project No. 639.021.751.

References

[Baarslag and Gerding, 2015] Tim Baarslag and Enrico Gerding. Optimal incremental preference elicitation during negotiation. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 3–9, Palo Alto, CA, USA, 2015. AAAI Press.

[Baarslag *et al.*, 2015] Tim Baarslag, Enrico Gerding, Reyhan Aydogan, and Monica Schraefel. Optimal negotiation decision functions in time-sensitive domains. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 2, pages 190–197, Dec 2015.

[Baarslag *et al.*, 2017] Tim Baarslag, Michael Kaisers, Enrico Gerding, Catholijn Jonker, and Jonathan Gratch. When will negotiation agents be able to represent us? the challenges and opportunities for autonomous negotiators. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4684–4690, Freiburg, DE, 2017. IJCAI.

[Bonilla *et al.*, 2010] Edwin Bonilla, Shengbo Guo, and Scott Sanner. Gaussian processes preference elicitation. *Advances in Neural Information Processing Systems 23*, pages 262–270, 2010.

[Boutilier, 2002] Craig Boutilier. A POMDP formulation of preference elicitation problems. In *Proceedings of the 18th National Conference on Artificial Intelligence*, pages 239–246, Menlo Park, CA, USA, 2002. AAAI.

[Chajewska *et al.*, 2000] Urszula Chajewska, Daphne Koller, and Ron Parr. Making rational decisions using adaptive utility elicitation. In *Proceedings of the 17th National Conference on Artificial Intelligence*, pages 363–369, Palo Alto, CA, USA, 2000. AAAI Press.

[Chu and Ghahramani, 2005a] Wei Chu and Zoubin Ghahramani. Gaussian processes for ordinal regression. *Journal of Machine Learning Research*, 6:1019–1041, 2005.

[Chu and Ghahramani, 2005b] Wei Chu and Zoubin Ghahramani. Preference learning with gaussian processes. In *Proceedings of the 22nd Conference on Machine Learning*, pages 137–144, New York, NY, USA, 2005. ACM.

[Ketter *et al.*, 2018] Wolfgang Ketter, John Collins, and Prashant Reddy. Power tac: A competitive economic simulation of the smart grid. *Energy Economics*, 39:262–270, 2018.

[Kingsley and Brown, 2006] David Kingsley and Thomas Brown. Preference uncertainty, preference learning and paired comparison experiments. *Land Economics*, 84:530–544, 2006.

[Kowalczyk and Bui, 2000] Ryszard Kowalczyk and Vinh Bui. On fuzzy e-negotiation agents: autonomous negotiation with incomplete and imprecise information. In *Proceedings 11th International Workshop on Database and Expert Systems Applications*, pages 1034–1038, Sept 2000.

[Kraus *et al.*, 1995] Sarit Kraus, Jonathan Wilkenfeld, and Gilad Zlotkin. Multiagent negotiation under time constraints. *Artificial Intelligence*, 75:297–345, 1995.

[Leahu *et al.*, 2019] Haralambie Leahu, Michael Kaisers, and Tim Baarslag. Preference learning in automated negotiation using gaussian uncertainty models. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*, pages 2087–2089, Richland, SC, USA, 2019. IFAAMAS.

[McGroarty *et al.*, 2018] Frank McGroarty, Ash Booth, Enrico Gerding, and Raju Chinthalapati. High-frequency trading strategies, market fragility and price spikes: An agent-based model perspective. *Annals of Operations Research*, pages 1–28, 2018.

[Mohammad and Nakadai, 2018] Yasser Mohammad and Shinji Nakadai. Fastvoi: Efficient utility elicitation during negotiations. In *Proceedings of the International Conference on Principles and Practice of Multi-Agent Systems*, pages 560–567, Zürich, Switzerland, 2018. Springer.

[Paruchuri *et al.*, 2009] Praveen Paruchuri, Nilanjan Chakraborty, Roie Zivan, Katia Sykara, Miroslav Dudik, and Geoffrey Gordon. Pomdp based negotiation modeling. *IJCAI MICON Workshop*, 2009.

[Ramchurn *et al.*, 2004] Sarvapali Ramchurn, Benjamin Deitch, Mark Thompson, David De Roure, Nicholas Jennings, and Michael Luck. Minimising intrusiveness in pervasive computing environments using multi-agent negotiation. In *Proceedings of the First Annual International MobiQuitous Conference: Networking and Services*, pages 364–371, Los Alamitos, CA, USA, 2004. IEEE Computer Society Press.

[Rasmussen and Williams, 2006] Carl Edward Rasmussen and Cristopher Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, USA, 2006.

[Sim, 2011] Kwang Mong Sim. Agent-based cloud computing. *IEEE Transactions on Services Computing*, 5:564–577, 2011.

[Wang *et al.*, 2009] Minhong Wang, Huaiqing Wang, Doug Vogel, Kuldeep Kumar, and Dickson Chiu. Agent-based negotiation and decision making for dynamic supply chain formation. *Engineering Applications of Artificial Intelligence*, 22:1046–1055, 2009.

[Zhong *et al.*, 2002] Fang Zhong, Steven Kimbrough, and Dong Wu. Cooperative agent systems: Artificial agents play the ultimatum game. *Group Decision and Negotiation*, 11(6):433–447, Nov 2002.