Retrieving the Structure of Utility Graphs Used in Multi-Item Negotiation through Collaborative Filtering[1]

Valentin Robu, Han La Poutré
CWI, Dutch Center for Mathematics and Computer Science
Kruislaan 413, NL-1098 SJ Amsterdam, The Netherlands
{robu, hlp}@cwi.nl

Abstract

Graphical utility models represent powerful formalisms for modeling complex agent decisions involving multiple issues [2]. In the context of negotiation, it has been shown [8] that using utility graphs enables agents to reach Pareto-efficient agreements with a limited number of negotiation steps, even for high-dimensional negotiations over bundles of items involving complementarity/ substitutability dependencies. This paper considerably extends the results of [8], by proposing a method for constructing the utility graphs of buyers automatically, based on previous negotiation data. Our method is based on techniques inspired from item-based collaborative filtering, used in online recommendation algorithms. Experimental results show that our approach is able to retrieve the structure of utility graphs online, with a relatively high degree of accuracy, for complex, non-linear (k-additive) preference settings, even if a relatively small amount of data about concluded negotiations is available.

1 Introduction

Negotiation represents a key form of interaction between providers and consumers in electronic markets. One of the main benefits of negotiation in e-commerce is that it enables greater customization to individual customer preferences, and it supports buyer decisions in settings which require agreements over complex contracts. Automating the negotiation process, through the use of intelligent agents which negotiate on behalf of their owners, enables electronic merchants to go beyond price competition by providing flexible contracts, tailored to the needs of individual buyers.

Multi-issue (or multi-item) negotiation models are particularly useful for this task, since with multi-issue negotiations mutually beneficial ("win-win") contracts can be

---

[1]This paper has been recently presented at the RRS'06 workshop, Hakodate, Japan [12] (proceedings to to appear as part of the Springer Lecture Notes in Computational Intelligence series). In this version of the paper, due to space limitations, the experimental set-up and tests performed to validate the model were not included. The full paper [12] (which is considerably longer, and includes the experimental results) is available at: `http://homepages.cwi.nl/~robu/rss2006.pdf`. We should also mention that the RRS'06 paper [12] represents complementary work to work on multi-issue negotiation model presented at the AAMAS'05 conference [8]. The interested reader can also consulte this paper at: `http://homepages.cwi.nl/~robu/aamas05negotiation.pdf`.

found [11, 4, 5, 8]. In this paper we consider the negotiation over the contents of a bundle of items (thus we use the term "multi-item" negotiation), though, at a conceptual level, the setting is virtually identical to previous work on multi-issue negotiation involving only binary-valued issues (e.g. [4]). A bottleneck in most existing approaches to automated negotiation is that they only deal with linearly additive utility functions, and do not consider high-dimensional negotiations and in particular, the problem of interdependencies between evaluations for different items. This is a significant problem, since identifying and exploiting substitutability/complementarity effects between different items can be crucial in reaching mutually profitable deals.

1.1 Using utility graphs to model negotiations over bundles of items

In our previous work [8], in order to to model buyer preferences in high-dimensional negotiations, we have introduced the concept of utility graphs. Intuitively defined, a utility graph (UG) is a structural model of a buyer, representing a buyer's perception of dependencies between two items (i.e. whether the buyer perceives two items to be as complementary or substitutable). An estimation of the buyer's utility graph can be used by the seller to efficiently compute the buyer's utility for a "bundle" of items, and propose a bundle and price based on this utility. The main result presented in [8] is that Pareto-efficient agreements can be reached, even for high dimensional negotiations with a limited number of negotiation steps, but provided that the seller starts the negotiation with a reasonable approximation of the *structure* of the true utility graph of the type of buyer he is negotiating with (i.e. he has a maximal structure of which issues could be potentially complimentary/substitutable in the domain).

The seller agent can then use this graph to negotiate with a specific buyer. During this negotiation, the seller will adapt the weights and potentials in the graph, based on the buyer's past bids. However, this assumes the seller knows a super-graph of the utility graphs of the class of buyers he is negotiating with (i.e. a graph which subsumes the types of dependencies likely to be encountered in a given domain - c.f. Sec. 2.2).

Due to space limitations, and to avoid too much overlap in content with our previous AAMAS paper [8], in this paper we do not describe the full negotiation model, the way seller weights are updated throughout the process, the initialization settings etc. These results have been described in [8], and we ask the interested reader to consult this work.

In this paper, we show this initial graph information can also be retrieved automatically, by using information from completed negotiation data. The implicit assumption we use here is that buyer preferences are in some way clustered, i.e. by looking at buyers that have shown interest for the same combinations of items in the past, we can make a prediction about future buying patterns of the current customer. Note that this assumption is not uncommon: it is a building block of most recommendation mechanisms deployed in Internet today [10]. In order to generate this initial structure of our utility graph, in this paper we propose a technique inspired by collaborative filtering.

1.2 Collaborative filtering

Collaborative filtering [10] is the main underlying technique used to enable personalization and buyer decision aid in today's e-commerce, and has proven very successful both in research and practice.

The main idea of collaborative filtering is to output recommendations to buyers, based on the buying patterns detected from buyers in previous buy instances. There are two approaches to this problem. The first of these is use of the preference database to discover, for each buyer, a neighborhood of other buyers who, historically, had similar preferences to the current one. This method has the disadvantage that it requires storing a lot of personalized information and is not scalable (see [10]). The second method, of more relevant to our approach, is **item-based collaborative filtering**. Item based techniques first analyze the user-item matrix (i.e. a matrix which relates the users to the items they have expressed interest in buying), in order to identify relationships between different items, and then use these to compute recommendations to the users [10]. In our case, of course, the recommendation step is completely replaced by negotiation. What negotiation can add to such techniques is that enables a much higher degree of customization, also taking into account the preferences of a specific customer. For example, a customer expressing an interest to buy a book on Amazon is sometimes offered a "special deal" discount on a set (bundle) of books, including the one he initially asked for. The potential problem with such a recommendation mechanism is that it's static: the customer can only take it, leave it or stick to his initial buy, it cannot change slightly the content of the suggested bundle or try to negotiate a better discount. By using negotiation a greater degree of flexibility is possible, because the customer can critique the merchant's sub-optimal offers through her own counter-offers, so the space of mutually profitable deals can be better explored.

1.3 Paper structure and relationship to previous work

The paper is organized as follows. In Section 2 we briefly present the general setting of our negotiation problem, define the utility graph formalism and the way it can be used in negotiations. Section 3 describes the main result of this paper, namely how the structure of utility graphs can be elicited from existing negotiation data. Section 4 discusses very briefly the experimental results from our model, fully presented in the RRS'06 paper [12]. Section 5 concludes the paper with a discussion.

An important issue to discuss is the relationship of this paper with our previous work. In our paper at the AAMAS'05 conference [8], we first introduced the utility graph formalism and present an algorithm that exploits the decomposable structure of such graphs in order to reach faster agreements during negotiation. That paper, however, uses the assumption that a minimal super-graph of individual buyer graphs is already available to the seller at the start of the negotiation. In the RRS'06 paper [12], we provide show how collaborative filtering could be used to build the structure of this super-graph and we propose a criteria for selecting the edges returned by the collaborative filtering process. This paper can be viewed as an extended abstract of these results.

For lack of space, we cannot present the full negotiation model from the AA-MAS'05 paper [8] in this paper, except at a very general level. The interested reader is therefore asked to consult [8] for further details.

2 The multi-issue negotiation setting

2.1 Utility Graphs: Definition and Example

We consider the problem of a buyer who negotiates with a seller over a bundle of n items, denoted by $B = \{I_1, \ldots, I_n\}$. Each item I_i takes on either the value 0 or 1: 1 (0) means that the item is (not) purchased. The utility function $u : Dom(B) \mapsto \mathbb{R}$ specifies the monetary value a buyer assigns to the 2^n possible bundles ($Dom(B) = \{0, 1\}^n$).

In traditional multi-attribute utility theory, u would be decomposable as the sum of utilities over the individual issues (items) [7]. However, in this paper we follow the previous work of [2] by relaxing this assumption; they consider the case where u is decomposable in *sub-clusters* of individual items such that u is equal to the sum of the *sub-utilities* of different clusters.

Definition: Let C be a set of (not necessarily disjoint) clusters of items $C_1, \ldots, C_r$ (with $C_i \subseteq B$). We say that a utility function is factored according to C if there exists functions $u_i : Dom(C_i) \mapsto \mathbb{R}$ ($i = 1, \ldots, r$ and $Dom(C_i) = \{0, 1\}^{|C_i|}$) such that $u(\vec{b}) = \sum_i u_i(\vec{c_i})$ where $\vec{b}$ is the assignment to the variables in B and $\vec{c_i}$ is the corresponding assignment to variables in C_i. We call the functions u_i sub-utility functions.

We use the following factorization, which is a relatively natural choice within the context of negotiation. Single-item clusters ($|C_i| = 1$) represent the individual value of purchasing an item, regardless of whether other items are present in the same bundle. Clusters with more than one element ($|C_i| > 1$) represent the *synergy effect* of buying two or more items; these synergy effects are positive for complementary items and negative for substitutable ones. In this paper, we restrict our attention to clusters of size 1 and 2 ($|C_i| \in \{1, 2\}, \forall i$). This means we only consider binary item-item complementarity/substitutability relationships, though the case of retrieving larger clusters could form the object of future research.

The factorization defined above can be represented as an undirected graph $G = (V, E)$, where the vertexes V represent the set of items I under negotiation. An arc between two vertexes (items) $i, j \in V$ is present in this graph if and only if there is some cluster C_k that contains both I_i and I_j. We will henceforth call such a graph G a *utility graph*. *Example 1* Let $B = \{I_1, I_2, I_3, I_4\}$ and $C = \{\{I_1\}, \{I_2\}, \{I_1, I_2\}, \{I_2, I_3\}, \{I_2, I_4\}\}$ such that u_i is the sub-utility function associated with cluster i ($i = 1, \ldots, 5$). Then the utility of purchasing, for instance, items I_1, I_2, and I_3 (i.e., $\vec{b} = (1, 1, 1, 0)$) can be computed as follows: $u((1, 1, 1, 0)) = u_1(1) + u_2(1) + u_3((1, 1)) + u_4((1, 1))$, where we use the fact that $u_5(1, 0) = u_5(0, 1) = 0$ (synergy effect only occur when two or more items are purchased). The utility graph of this factorization is depicted in Fig. 1.
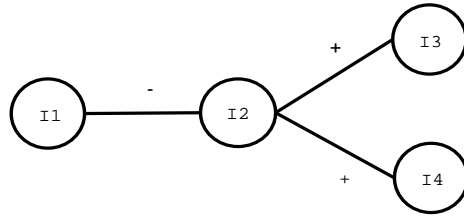
Figure 1: The utility graph that corresponds to the factorization according to C in Example 1. The $+$ and $-$ signs on the edges indicate whether the synergy represents a complementarity, respectively substitutability effect.

2.2 Minimal super-graph for a class of buyers

The definition of utility graphs given in Section 2.1 corresponds to the modeling the utility function of an individual buyer. In this paper, we call the utility graph of an individual buyer the *underlying* or *true* graph (to distinguish it from the *retrieved* or *learned* graph, reconstructed through our method). The underlying graph of any buyer remains hidden from the seller throughout the negotiation.

We do assume, however, that the buyers which negotiate with a given electronic merchant belong to a certain class or population of buyers. This means the utility buyers assign to different bundles of items follow a certain structure, specific to a buying domain (an assumption also used indirectly in [11, 10]). Buyers from the same population are expected to have largely overlapping graphs, though not all buyers will have all interdependencies specific to the class.

Definition: Let $A = \{A_1, ..A_n\}$ be a set (class, population) of n buyers. Each buyer $i = 1..n$ has a utility function u_i, which can be factored according to a set of clusters $C_i = \{C_{i,1}, C_{i,2}..C_{I,r(i)}\}$. We define the super-set of clusters for the class of buyers $A = \{A_1, ..A_n\}$ as: $C_A = C_1 \cup C_2 \cup .. \cup C_n$.

In graph-theoretic terms (as shown in Section 2.1), the set of clusters C_i according to which the utility a buyer A_i is structured is represented by a utility graph G_i, where each binary cluster from $\{C_{i,1}, ..C_{I,r(i)}\}$ represents a dependency or an edge in the graph. The super-set of buyer clusters C_A can also be represented by a graph G_A, which is the **minimal super-graph** of graphs G_i, $i = 1..n$. This graph is called **minimal** because it contains no other edges than those corresponding to a dependency in the graph of at least one buyer agent from this class. We illustrate this concept by a very simple example, which also relies on Fig. 1.

2.3 Application to negotiation

The negotiation, in our model, follows an alternating offers protocol. At each negotiation step each party (buyer/seller) makes an offer which contains an instantiation with 0/1 for all items in the negotiation set (denoting whether they are/are not included in the proposed bundle), as well as a price for that bundle. The decision process of the seller agent, at each negotiation step, is composed of 3 inter-related parts: (1) take into

account the previous offer made by the other party, by updating his estimated utility graph of the preferences of the other party, (2) compute the contents (i.e. item configuration) of the next bundle to be proposed, and (3) compute the price to be proposed for this bundle. In this model, the seller maintains of his buyer is represented by a utility graph, and tailors this graph towards the preferences of a given buyer, based on his/her previous offers.

The seller does not know, at any stage, the values in the actual utility graph of the buyer, he only has an approximation learned after a number of negotiation steps. However, the seller does have some prior information to guide his opponent modeling. He starts the negotiation by knowing a *super-graph of possible inter-dependencies* between the issues (items) which can be present for the class of buyers he may encounter. The utility graphs of buyers form subgraphs of this graph. Note that this assumption says nothing about values of the sub-utility functions, so the negotiation is still with double-sided incomplete information (i.e. neither party has full information about the preferences of the other).

2.4   Overview of our approach

There are two main stages of our approach:

1. Using information from previously concluded negotiations to construct the structure of the utility super-graph. In this phase the information used (past negotiation data) refers to a class of buyers and is not traceable to individuals.

2. The actual negotiation, in which the seller, starting from a super-graph for a class (population) of buyers, will negotiate with an individual buyer, drawn at random from the buyer population above. In this case, learning occurs based on the buyer's previous bids during the negotiation, so information is buyer-specific. However, this learning at this stage is guided by the structure of the super-graph extracted in the first phase.

3   Constructing the Structure of Utility Graphs Using Concluded Negotiation Data

Suppose the seller starts by having a dataset with information about previous concluded negotiations. This dataset may contain complete negotiation traces for different buyers, or we may choose, in order to minimize bias due to uneven-length negotiations, to consider only one record per negotiation. This can be either the first bid of the buyer or the bundle representing the outcome of the negotiation. The considered dataset is not personalized, i.e. the data which is collected online cannot be traced back to individual customers (this is a reasonable assumption in e-commerce, where storing a large amount of personalized information may harm customer privacy). However, in constructing of the minimal utility graph which the customers use, we implicitly assume that customers' preference functions are related - i.e. their corresponding utility graphs, have a (partially) overlapping structure.

Our goal is to retrieve the **minimal super-graph** of utility interdependencies which can be present for the class or population of buyers from which the negotiation data was generated. This past data can be seen as a $N * n$ matrix, where N is the number of previous negotiation instances considered and n is the number of issues (e.g. 50 for our tests). Item-based collaborative filtering [10] works by computing "similarity measures" between all pairs of items in the negotiation set. The steps used are:

1. Compute item-item similarity matrices (from the raw statistics)

2. Compute qualitative utility graph, by selecting which dependencies to consider from the similarity matrices.

In the following, we will use the following notations:

- N for the total number of previous negotiation outcomes considered

- For each item i=1..n, $N_i(1)$ and $N_i(0)$ represent the number of times the item was (respectively was not) asked by the buyer, from the total of N previous negotiations

- For each pair of issues $i, j = 1..n$ we denote by $N_{i,j}(0,0)$, $N_{i,j}(0,1)$, $N_{i,j}(1,0)$ and $N_{i,j}(1,1)$ all possibilities of joint acquisition (or non acquisition) of items i and j.

3.1 Computing the similarity matrices

The literature on item-based collaborative filtering defines two main criteria that could be used to compute the similarity between pairs of items: **cosine-based** and **correlation-based** similarity. In our work we have considered both, but experimental results showed that only correlation-based similarity seems to perform well for this task. Cosine-based similarity is conceptually simpler, and, from our experience, works well in detecting complementarity dependencies and only in the case when the data is relatively sparse (each buyer expresses interest only in a few items). Correlation-based similarity, however, does not have these limitations. Therefore, in this paper, we report the formulas and experimental results only for correlation-based similarity. Since the mathematical definitions (as presented in [10]) is given for real-valued preference ratings, we derive a more simplified form for the binary values case.

3.1.1 Correlation-based similarity

For correlation-based similarity, just one similarity matrix is computed containing both positive and negative values (to be more precise between -1 and 1). We first we define for each item $i = 1..n$, the average buy rate:

$$Av_i = \frac{N_i(1)}{N} \tag{1}$$

The following two terms are defi ned:

$$\psi_1 = N_{i,j}(0,0) * Av_i * Av_j - N_{i,j}(0,1) * Av_i * (1 - Av_j) \\ - N_{i,j}(1,0) * (1 - Av_i) * Av_j + N_{i,j}(1,1) * (1 - Av_i) * (1 - Av_j)$$

and the normalization factor:

$$\psi_2 = \sqrt{\frac{N_i(0) * N_i(1)}{N}} * \sqrt{\frac{N_j(0) * N_j(1)}{N}}$$

The values in the correlation-based similarity matrix are then computed as:

$$Sim(i,j) = \frac{\psi_1}{\psi_2} \qquad (2)$$

3.2 Building the super-graph of buyer utilities

After constructing the similarity matrices, the next step is to use this information to build the utility super-graph for the class of buyers likely to be encountered in future negotiations. The item-item correlation similarity already provides a measure of how strong complementarity/substitutability dependencies are on average, by closeness to 1 or -1. However, we still need a method for deciding how many of the item-item relationships from the similarity matrices should be included in the fi nal graph.

Ideally, all the inter-dependencies corresponding to the arcs in the graph representing the true underlying preferences of the buyer should feature among the highest (respectively the lowest) values in the retrieved correlation tables. When an interdependency is returned that was not actually in the true graph, we call this is an excess (extra, erroneous) arc or interdependency. Due to noise in the data, it is unavoidable that a number of such excess arcs are returned. For example, if item I_1 has a complimentary value with I_2 and I_2 is substitutable with I_3, it may be that items I_1 and I_3 often do not appear together, so the algorithm detects a substitutability relationship between them, which is in fact erroneous.

The question on the part of the seller is: how many dependencies should be considered from the ones with highest correlation, as returned by the fi ltering algorithm? There are two aspects that affect this cut-off decision:

- If too few dependencies are considered, then it is very likely that some dependencies (edges) that are in the true underlying graph of the buyer will be missed. This means that the seller will ignore some interdependencies in the negotiation stage completely, which can adversely affect the Pareto-effi ciency of the reached agreements.

- If too many dependencies are considered, then the initial starting super-graph of the seller will be considerably more dense than the "true" underlying graph of the buyer (i.e. it contains many excess or extra edges). Actually, this is always the case to some degree, and in [8] we claim that Pareto-effi cient agreements can be reached starting from a super-graph of the buyer graphs. However, this super-graph cannot be of unlimited size. For example, starting from a graph close to

full connectivity (i.e. with n^2 edges for a graph with n issues or vertexes) would be equivalent to providing no prior information to guide the negotiation process.

In the general case, we consider graphs whose number of edges (or dependencies) is a linear in the number of items (issues) in the negotiation set. Otherwise stated, we restrict our attention to graphs in which the number of edges considered is some linear factor k times the number of items (vertexes) negotiated on. Framed in this way, the problem becomes of choosing the optimal value for parameter k (henceforth denoted by k_{opt}).

3.3 Minimization of expected loss in Gains from Trade as cut-off criteria

Denote by $N_{missing}$ the number of edges that are in the "true", hidden utility graph of the buyer, but will not be present in the super-graph built through collaborative filtering. Similarly, we denote by N_{extra} the number of excess (or erroneous) edges, that will be retrieved, but are not in the true utility graph of the buyer.

The number of edges which are missing (not accurately retrieved) or excess (too many extra edges) depend on the accuracy and precision of the underlying collaborative filtering process. More precisely stated, the number of missing edges depends on 3 parameters: the type of filtering used (correlation or cosine-based), the amount of concluded negotiation records available for filtering (we denote this number by N_r) and the number of edges considered in the cut-off criteria, k. Formally, we can thus write: $N_{missing}(corr, N_r, k)$. In this section we focus, however, exclusively on choosing a value for k, and consider the other two parameters as already chosen at the earlier step. Thus we simplify the notation to: $N_{missing}(k)$ and $N_{extra}(k)$, respectively.

As discussed in Section 3.2, both having missing and too many extra edges influences the efficiency of outcome of the subsequent negotiation process. Our goal is to choose a value for k that minimizes this expected efficiency loss during the negotiation. The efficiency loss, in our case, is measured as the difference in Gains from Trade which can be achieved using a larger/smaller graph, compared to the Gains from Trade which can be achieved by using the "true" underlying utility graph of the buyer (in earlier work [11, 8], we have shown that maximizing the Gains from Trade in this setting is equivalent to reaching Pareto optimality).

In order to estimate this error rate, we consider a second **negotiation test set**, different from the one used for filtering. The purpose of this second test set is to obtain an estimation of the loss in gains from trade which occurs if we use a sparser/denser graph than the true underlying graph of the buyer. In more formal terms, the expected utility loss for using k edges can be written as:

$$E_{loss\_GT}(k) = max \langle E_{loss\_GT}(N_{missing}(k)), E_{loss\_GT}(N_{extra}(k)) \rangle \qquad (3)$$

The optimal choice of k can then be computed as:

$$k_{opt} = argmin_k E_{loss\_GT}(k) \qquad (4)$$

Our criteria for choosing k presented in Equations 4 are not dissimilar to "min-max regret" decision criteria, often used in preference elicitation problems [1]. We could also use the name "regret" for the expected loss in gains from trade, but to keep the names consistent with our earlier AAMAS work [8] we prefer the term "GT loss".

4 Experimental evaluation

The model above was tested for a setting involving 50 binary-valued issues (items). For each set of tests, the structure of the graph was generated at random (with uniform distribution), by selecting at random the items (vertexes) connected by each edge representing a utility inter-dependency. For 50 issues, 75 random binary dependencies were generated for each test set, 50 of which were positive dependencies and 25 negative. Two sets of tests were performed: one for assessing the efficiency of the collaborative filtering itself (i.e. the cosine and correlation similarity criteria) and one for detecting the cut-off limit for the maximal graph. In this paper we only report the results for correlation-based filtering, since this was found to perform considerably better than the cosine-based one. Next, we studied the effect of different cut-off criteria (values of k) on the negotiation process itself.

4.1 Results for the efficiency of the filtering criteria

There are two dimensions across which the two criteria need to be tested:

- **The strength of the interdependencies in the generated buyer profiles**. This is measured as a ratio of the average strength of the inter-dependency over the average utilities of an individual item. Each buyer profile is generated as follows:

  First, for each item, an individual value is generated by drawing from identical, independent normal distributions (i.i.d.) of center $C_{individual-item} = 1$ and variance 0.5. Next, the substitutability/complementarity effects for each binary issue dependency (i.e. each cluster containing two items) are generated by drawing from a normal i.i.d-s with a centers $C_{non-linearity}$ and the same spread 0.5. The strength of the interdependency is then taken to be $\frac{C_{non-linearity}}{C_{individual-item}}$. The smaller this ratio is, the more difficult it will be to detect non-linearity (i.e. complementarity and substitutability effects between items). In fact, if this ratio takes the value 0, there are no effects to detect (which explains the performance at this point), at 0.1 the effects are very weak, but they become stronger as it approaches 1 and 2.

- **Number of previous negotiations from which information (i.e. negotiation trace) is available**.

The performance measure used is computed as follows. Each run of an algorithm (for a given history of negotiations, and a certain probability distribution for generating that history) returns an estimation of the utility graph of the buyer. Our performance measure is the recall, i.e. the percentage of the dependencies from the underlying

utility graph of the buyer (from which buyer profiles are generated) which are found in the graph retrieved by the seller.
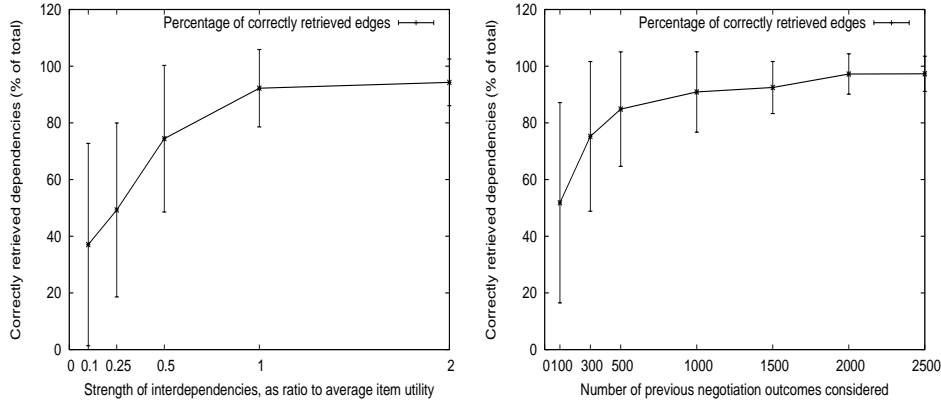

Figure 2: Results for the correlation-based similarity. Left-side graph gives the percentage of correctly retrieved dependencies, with respect to the average interdependency strength, while right-side graph gives the percentage of correctly retrieved dependencies with respect to the size of the available dataset of past negotiation traces.

4.2   Effect of the maximal graph size considered on the negotiation process

After measuring the effect of the two similarity criteria considered (i.e. cosine and correlation-based), as well as the effect of different amounts of data, we present results for different cut-off sizes for the maximal graph (i.e. the k parameter introduced in Section 3.3). For all tests reported in this Section, we used correlation-based similarity and we assumed 1000 records of previous negotiations are available for filtering. We chose to focus on correlation-based similarity since this criteria clearly performs better, in this setting, than cosine-based similarity. As shown in Sec. 4.1, 1000 records is a reasonable amount of data to ensure a good retrieval accuracy.

From Figs. 3 and 4, several conclusions can be drawn. First, missing edges from the graph the Seller starts the negotiation with has a considerably greater negative effect than adding too many extra (erroneous) edges.

Thus, as shown in Fig. 3, in order to get above 90% of the optimal Gains from Trade in future negotiations, the retrieval process cannot miss more than about 15% of the true inter-dependencies in the true graph of the Buyer. However, having a considerably denser starting graph does not degrade the performance so significantly. In fact, as we see in Fig. 4, having 3 times as many edges than in the original buyer graph (which means 2/3 of all edges are erroneous), only decreases performance with around 4%. The fact that there is still a decreasing effect can probably be explained
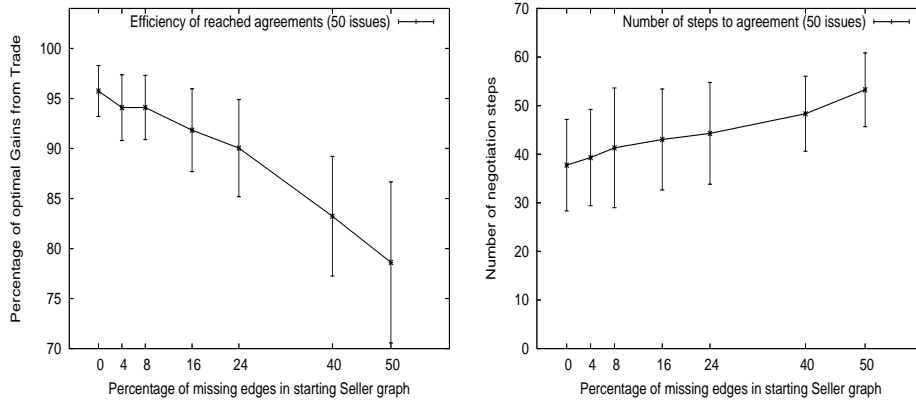
Figure 3: Effect of missing edges (dependencies) in the starting Seller graph on the Pareto-optimality of reached negotiation outcomes
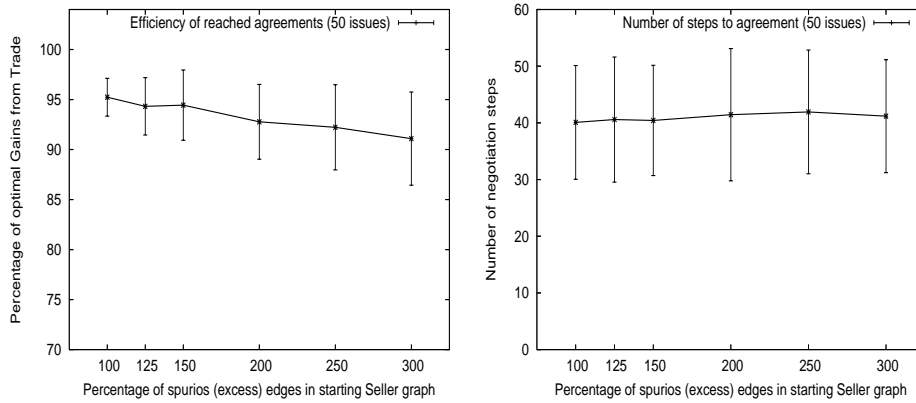

Figure 4: Effect of excess (erroneous) edges in the starting Seller graph on the Pareto-optimality of reached negotiation outcomes

from the interaction between the non-linear effects introduced by the structure and the non-linear effects introduced by the tails of normal distributions in each cluster. Finally, we observe that, in both cases, the negotiation speed does not seem to be very significantly affected and it remains around 40 steps/thread, on average.

5 Discussion

Several previous results model automated negotiation as a tool for supporting the buyer's decision process in complex e-commerce domains [11, 4, 6]. Most of the work in multi-issue negotiations has focused on the independent valuations case. Faratin, Sierra & Jennings [5] introduce a method to search the utility space over multiple at-

tributes, which uses fuzzy similarity criteria between attribute value labels as prior information. These papers have the advantage that they allow flexibility in modeling and deal with incomplete preference information supplied by the negotiation partner. They do not consider the question of functional interdependencies between issues, however.

A negotiation approach that specifically address the problem of complex interdependencies between multiple issues is Klein et al. [4]. They consider a setting similar to the one considered in this paper, namely bilateral negotiations over a large number of boolean-valued issues with binary interdependencies. In this setting, they compare the performance of two search approaches: hill-climbing and simulated annealing and show that if both parties agree to use simulated annealing, then Pareto-efficient outcomes can be reached. By comparison to our work, this approach does not try to use prior information, in the form of the clustering effect between the preference functions of different buyers, in order to shorten individual negotiation threads.

Our approach to modeling multi-issue negotiation relies on constructing an explicit model of the buyer utility function - in the form of a utility graph. A difference of our approach (presented both in this paper and in [8]) from other existing negotiation approaches is that we use information from previous negotiations in order to aid buyer modeling in future negotiation instances. This does not mean that personalized negotiation information about specific customers needs to be stored, only aggregate information about all customers. The main intuition behind our model is that we explicitly utilize, during the negotiation, the clustering effect between the structure of utility functions of a population of buyers. This is an effect used by many Internet product recommendation engines today, in order to shorten the period required for customers to search for items (though it comes under different names: collaborative filtering, social filtering etc.). When adapted and used in a negotiation context, such techniques enable us to handle high dimensional and complex negotiations efficiently (with a limited number of negotiation steps).

The main contribution of this paper, in addition to the one highlighted in [8], is that it shows that the whole process can be automatic: no human input is needed in order to achieve efficient outcomes. We achieve this by using techniques derived from collaborative filtering (widely used in current e-commerce practice) to learn the structure of utility graphs used for such negotiations. We thus show that the link between collaborative filtering and negotiation is a fruitful research area, which, we argue, can lead to significant practical applications of automated negotiation systems.

As future work, there are several directions which could be explored in this area. An immediate one is to obtain a precise definition of the classes of non-linearity (in terms of utility graph structure and density) for which it is possible to reach efficient agreements with a linear number of negotiation steps. To this end, we intend to make use of results from random graph theory [9] and constraint processing [3].

References

[1] D. Brazunias and C. Boutilier. Local utility elicitation in gai models. In *Proc. of the Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 42–49, 2005.

[2] U. Chajewska and D. Koller. Utilities as random variables: Density estimation and structure discovery. In *Proceedings of sixteenth Annual Conference on Uncertainty in Artificial Intelligence UAI-00*, pages 63–71, 2000.

[3] R. Dechter. *Constraint Processing*. Morgan Kaufmann Publishers, San Francisco, USA, 2003.

[4] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12:111–125, 2003.

[5] N. R. Jennings P. Faratin, C. Sierra. Using similarity criteria to make issue trade-offs in automated negotiations. *Journal of Artificial Intelligence*, 142(2):205–237, 2002.

[6] P. Maes R. Gutman. Agent-mediated integrative negotiation for retail electronic commerce. In *Agent Mediated Electronic Commerce, Springer LNAI vol. 1571*, pages 70–90, 1998.

[7] H. Raiffa. *The art and science of negotiation*. Harvard University Press, Cambridge, Massachussets USA, 1982.

[8] V. Robu, D.J.A. Somefun, and J. A. La Poutré. Modeling complex multi-issue negotiations using utility graphs. In *4th Int. Conf. on Autonomous Agents & Multi Agent Systems (AAMAS), Utrecht, The Netherlands*, 2005.

[9] A. Rucinski S. Janson, T. Luczak. *Random Graphs*. Wiley, New York, USA, 2000.

[10] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Tenth International WWW Conference (WWW10), Hong Kong*, 2001.

[11] D.J.A. Somefun, T.B. Klos, and J.A. La Poutré. Online learning of aggregate knowledge about nonlinear preferences applied to negotiating prices and bundles. In *Proc. 6th Int Conf. on E-Commerce, Delft*, pages 361–370, 2004.

[12] J.A. La Poutré V. Robu. Retrieving the structure of utility graphs used in multi-item negotiations through collaborative filtering of aggregate buyer preferences. In *Proc. of the 2nd Int. Wk. on Rational, Robust and Secure Negotiations in MAS, Hakodate, Japan*. Springer LNCI (to appear), 2006.