Balancing Utility and Deal Probability for Auction-Based Negotiations in Highly Nonlinear Utility Spaces

Ivan Marsa-Maestre
Miguel A. Lopez-Carmona
Juan R. Velasco
Universidad de Alcala, Spain
ivan.marsa@uah.es
miguelangel.lopez@uah.es
juanramon.velasco@uah.es

Takayuki Ito[*] and **Mark Klein**
Center for Collective Intelligence
MIT Sloan School of Management
Cambridge, Massachusetts, US
takayuki@mit.edu
m_klein@mit.edu

Katsuhide Fujita
Nagoya Institute of Technology
Nagoya, Japan
fujita@itolab.mta.nitech.ac.jp

Abstract

Negotiation scenarios involving nonlinear utility functions are specially challenging, because traditional negotiation mechanisms cannot be applied. Even mechanisms designed and proven useful for nonlinear utility spaces may fail if the utility space is highly nonlinear. For example, although both contract sampling and constraint sampling have been successfully used in auction based negotiations with constraint-based utility spaces, they tend to fail in highly nonlinear utility scenarios. In this paper, we will show that the performance of these approaches decrease drastically in highly nonlinear utility scenarios, and propose a mechanism which balances utility and deal probability for the bidding and deal identification processes. The experiments show that the proposed mechanisms yield better results than the previous approaches in highly nonlinear negotiation scenarios.

1 Introduction

Complex negotiations scenarios where agents negotiate about multiple, interdependent issues are specially challenging, since issue interdependency yields nonlinear utility functions for the agents, and thus the classic mechanisms for linear negotiation models are not applicable. In particular, this work focuses on multilateral mediated negotiation, where several agents try to reach an agreement over a range of issues using a bidding based negotiation protocol with the aid of a mediator. The utility spaces for the agents are generated using weighted constraints, which results in nonlinear utility functions.

In [Ito *et al.*, 2007], a bidding mechanism is proposed, which is based on taking random samples of the contract space and applying simulated annealing to these samples to identify high utility regions for each agent, sending these regions as bids to a mediator, and then performing a search in the mediator to find overlaps between the bids of the different agents. In a similar scenario [Marsa-Maestre *et al.*, 2008], samples are taken from the constraints space instead. Experiments show that these approaches achieve high effectiveness (measured as high optimality rates and low failure rates for the negotiations) in the evaluation scenario they describe (Section 2). However, as we will show empirically in Section 5.2, these approaches perform worse as the circumstances of the scenario turn harder (that is, when the utility functions are highly nonlinear, like in B2B interactions or distributed automated control systems). Under these circumstances, the failure rate increases drastically, raising the need for an alternative approach.

Furthermore, as described in [Ito *et al.*, 2007], their bidding-based negotiation protocol presents some scalability concerns due to the extensive search for overlaps performed in the mediator, which finally limits the maximum number of bids each agent may send depending on the number of agents in the negotiation. In this paper, we intend to address these problems in the following ways:

- We propose a mechanism to take into account both the utility of a bid for an agent and its *viability* (a measure of the likelihood of the bid to yield a deal), and integrate this mechanism in the contracts sampling and constraint sampling approaches (Section 3). We will show that this balance between bid utility and deal probability yields a significant improvement in terms of optimality rate and failure rate over the previous approaches in highly nonlinear scenarios.

- We propose a heuristic search mechanism for the mediator which lowers the scalability problem while achieving acceptable optimality rates (Section 4).

A highly-nonlinear simulated scenario has been devised to validate our hypothesis and evaluate the effects of our contributions. This scenario is described in Section 5, along with the discussion of the results obtained. Finally, our proposal is briefly compared to the most closely-related works in the state-of-the-art (Section 6). The last section summarizes our conclusions and sheds light on some future research.

[*]Visiting from Nagoya Institute of Technology

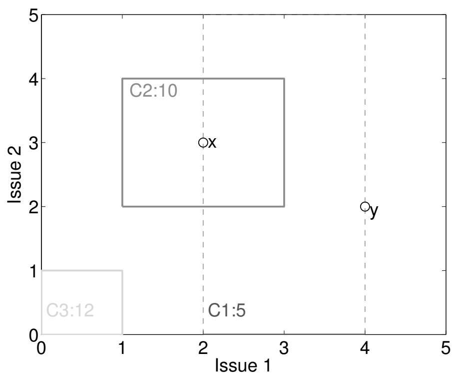

Figure 1: Example of a utility space with two issues and three constraints.

2 Contract and Constraint Sampling for Negotiation in Nonlinear Utility Spaces

2.1 Constraint-based Nonlinear Utility Spaces

Nonlinear agent preferences can be described by using different categories of functions [Zhang and Pu, 2004]. In this work we focus on nonlinear utility spaces generated by means of weighted constraints [Ito *et al.*, 2007]. In these cases, agents' utility functions are described by defining a set of constraints. Each constraint represents a region with one or more dimensions, and has an associated utility value. The number of dimensions of the space is given by the number of issues n under negotiation, and the number of dimensions of each constraint must be lesser or equal than n. The utility yielded by a given potential solution (contract) in the utility space for an agent is the sum of the utility values of all the constraints that are satisfied by that contract. Figure 1 shows a very simple example of an agent's utility space for two issues and three constraints: a unary constraint $C1$ and two binary constraints $C2$ and $C3$. The utility values associated to the constraints are also shown in the figure. In this example, contract x would yield a utility value for the agent $u(x) = 15$, since it satisfies both $C1$ and $C2$, while contract y would yield a utility value $u(y) = 5$, because it only satisfies $C1$.

More formally, we can define the issues under negotiation as a finite set of variables $x = \{x_i|i = 1, ..., n\}$, and a contract (or a possible solution to the negotiation problem) as a vector $s = \{x_i^s|i = 1, ..., n\}$ defined by the issues' values. Issues take values from the domain of integers $[0, X]$.

Agent utility space is defined as a set of constraints $C = \{c_k|k = 1, ..., l\}$. Each constraint is given by a set of intervals which define the region where a contract must be contained to satisfy the constraint. In this way a constraint c is defined as $c = \{I_i^c|i = 1, ..., n\}$, where $I_i^c = [x_i^{min}, x_i^{max}]$ defines the minimum and maximum values for each issue to satisfy the constraint. Each constraint c_k has an associated utility value $u(c_k)$.

A contract s satisfies a constraint c if and only if $x_i^s \in I_i^c \forall i$. For notation simplicity, we denote this as $s \in x(c_k)$, meaning that s is in the set of contracts that satisfy c_k. An agent's utility for a contract s is defined as $u(s) = \sum_{c_k \in C | s \in x(c_k)} u(c_k)$, that is, the sum of the utility values of all constraints satisfied by s. This kind of utility functions produce nonlinear utility spaces, with high points where many constraints are satisfied, and lower regions where few or no constraints are satisfied.

For this work, we will consider as the *optimal contract* of a negotiation the contract which satisfies the Nash product [Nash, 1953], that is, the contract which maximizes the product of the utilities of all agents involved in the negotiation. Though the search for such a contract could be performed in a complete information scenario using distributed constraint optimization techniques [Yokoo, 2001] or well-known nonlinear optimization techniques such as evolutionary algorithms, we are assuming a competitive scenario, where agents are unwilling to fully reveal their preference information. In these scenarios, these optimization techniques are not applicable, and therefore other approaches are needed, like the ones described in the following sections.

2.2 Contracts Sampling and Simulated Annealing in Bidding-based Nonlinear Negotiation

Ito et al. [Ito *et al.*, 2007] presented a bidding-based protocol to deal with nonlinear utility spaces generated using weighted constraints. The protocol consists on the following four steps:

1. *Sampling:* Each agent takes a fixed number of random samples from the contract space, using a uniform distribution.
2. *Adjusting:* Each agent applies simulated annealing to each sample to try to find a local optimum in its neigborhood. This results in a set of high-utility contracts.
3. *Bidding:* Each agent generates a bid for each high-utility, adjusted contract. The bids are generated as the intersection of all constraints which are satisfied by the contract. Each agent sends its bids to the mediator, along with the utility associated to each bid.
4. *Deal identification:* The mediator employs breadth-first search with branch cutting to find overlaps between the bids of the different agents. The regions of the contract space corresponding to the intersections of at least one bid of each agent are tagged as potential solutions. The final solution is the one that maximizes joint utility, defined as the sum of the utilities for the different agents.

2.3 Maximum Weight Independent Set and the Max-product Algorithm

In [Marsa-Maestre *et al.*, 2008], an alternative perspective for the bidding process is given, looking at the constraint-based agent utility space as a weighted undirected graph. Consider again the simple utility space example shown in Figure 1. Think about each constraint as a node in the graph, with an associated weight which is the utility value associated to the constraint. Now we will connect all nodes whose corresponding constraints are *incompatibles*, that is, they have no intersection. The resulting graph is shown in Figure 2.

To find the highest utility bid in such a graph can be seen as finding the set of unconnected nodes which maximizes the sum of the nodes' weights. Since only incompatible nodes are connected, the corresponding constraints will have non-null intersection. In the example, this would be achieved by taking the set $\{C1, C2\}$. The problem of finding a maximum weight set of unconnected nodes is a well-known problem called maximum weight independent set (MWIS). Though MWIS problems are NP-hard, in [Bayati *et al.*, 2008], a message passing algorithm is used to estimate MWIS. The algorithm is a reformulation of the classical max-product algorithm called "min-sum", and works as follows. Initially, every nodes i send their weights ω_i to their neighbors $N(i)$ as messages. At each iteration, each node i updates the message to send to each neighbor j by substracting from its weight ω_i the sum of the messages received from *all other* neighbors *except* j. If the result is negative, a zero value is sent as message. Upon receiving the messages, a node is included in the estimation of the $MWIS$ if and only if its weight is greater than the sum of al messages received from its neighbors. Message passing continues until $MWIS$ converges or the maximum number of iterations is exceeded. This is formally shown in Algorithm 1.

Input:
$i = 1, ..., n$: nodes (constraints) in the weighted graph
$\omega_i | i = 1, ..., n$: weight (utility) of each node (constraint)
$N(i)$: set of neighbors of each node (incompatible constraints)
t_{max}: maximum number of iterations
Output: $MWIS$: estimation of the MWIS
$t = 0; m^t_{i \to j} = \omega_i \forall j \in N(i)$
while $t < t_{max}$ **do**
  $t = t + 1$; **foreach** i **do**
    $m^t_{i \to j} = \max\{0, \omega_i - \sum_{k \neq j, k \in N(i)} m^{t-1}_{k \to i}\}$
  end
  $MWIS^t = \{i | \omega_i > \sum_{k \in N(i)} m^{t-1}_{k \to i}\}$
  if $t > 1$ *and* $MWIS^t = MWIS^{t-1}$ **then**
    return $MWIS^t$
end

Algorithm 1: Min-sum algorithm for MWIS estimation

Since the algorithm is deterministic, only one bid can be generated for a given set of constraints. To solve this, in [Marsa-Maestre *et al.*, 2008], the algorithm is applied to a subset of constraints $C' = \{c'_k | k = 1, ..., n_c; n_c < l; c'_k \in C\}$. The constraints c'_k are randomly chosen from the constraint set C. In this way, a different constraint subset C' is passed to the algorithm at each run, which will result in different, non-deterministic bids.

3 Bidding Mechanisms for Highly-nonlinear Utility Spaces

The use of weighted constraints generates a "bumpy" utility space, with many peaks and valleys. However, the degree of "bumpiness" is highly dependent on the way the constraint set is generated, and specially on the average width of the constraints. In [Ito *et al.*, 2007], constraints are generated by choosing the width of each constraint in each issue randomly within the [3,7] interval. Since the domain is chosen to be [0,9], this generates rather "wide" constraints. Figure

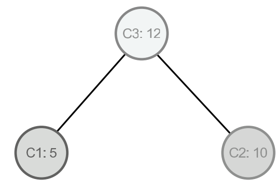

Figure 2: Weighted undirected graph resulting from the utility space in Figure 1.

3 shows an utility space obtained using "narrow" constraints, choosing their widths from the [1,5] interval. Though both utility spaces are nonlinear, the space generated using narrow constraints is more complex, with narrower peaks and valleys. As the number of issues under consideration increases, the differences between having wide or narrow constraints become more relevant. Though the approaches proposed in [Ito *et al.*, 2007] and [Marsa-Maestre *et al.*, 2008] work in scenarios defined using wide constraints, we will see that their performance (in terms of optimality and failure rate) decreases drastically in highly nonlinear scenarios defined using narrow constraints, and therefore an alternative approach is needed to deal with these highly nonlinear utility spaces.

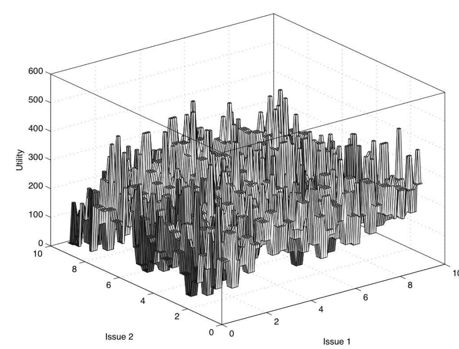

Figure 3: Example of a highly nonlinear utility space generated by using "narrow" constraints.

3.1 Constraint/Bid Quality Factor

The main difference between utility spaces generated using wide or narrow constraints is the width of the peaks. Highly-nonlinear scenarios will yield narrower peaks. Since the mechanisms outlined above lead agents to choose those peaks (or high-utility regions) as bids, the result is that narrower bids will be sent to the mediator. The width of the bids (or more generally, the *volume* of the bids, computed as the cardinality of the set of contracts which match the bid), will directly impact the probability that the bid overlaps a bid of another agent, and thus its *viability*, that is, the probability of the bid resulting in a deal. Intuitively, an agent with no knowledge of the other agents' preferences should try to adequately balance the utility of their bids (to maximize its own profit) and the volume of those bids (to maximize the probability of a successful negotiation). To formally represent this, we define the *quality factor* of a constraint or a bid as $Q_c = u_c^\alpha \cdot v_c^{1-\alpha}$, where u_c and v_c are, respectively, the utility and volume of the bid or constraint c, and $\alpha \in [0, 1]$ is a parameter which models the risk attitude of the agent. A risk averse agent ($\alpha < 0.5$) will tend to qualify as better bids those that are wider, and thus are more likely to result in a deal. A risk willing or selfish agent ($\alpha > 0.5$) will, in contrast, give more importance to bid utility.

Our hypothesis is that by taking into account this quality factor in the bidding mechanisms, with adequate values for the parameter α, will result in a better balance between utility and "width" in agent bids, and thus negotiations will yield higher optimality rates and lower failure rates.

3.2 Using the Quality Factor within the Simulated Annealing Algorithm

To make the simulated annealing bidding approach to take advantage of the quality factor Q is fairly straightforward. We just need to make the simulated annealing optimizer to search for contracts which maximize the quality factor Q instead of the agent utility. Since the quality factor Q is a feature of a region, not a contract, the adjusted contracts must be mapped to the high utility regions where they are contained before they are accepted or rejected by the simulated annealing engine. This can be easily done by checking all constraints in the agent preference model and computing the intersection of the constraints which are satisfied by the candidate contract. The volume of this intersection can then be used to compute the quality factor Q of the region.

3.3 Q-based Tournament Selection for the MWIS approach

The quality factor Q cannot be directly introduced into the max-product or min-sum algorithm, because the algorithm is based in a weighted graph where weights are additive, and the quality factor is not additive (that is, the quality factor of the intersection of a set of constraints is not the sum of the quality factor of the constraints). Thus, a different approach is needed to introduce this factor in the algorithm. We propose to use a *tournament selection* [Miller and Goldberg, 1995] based on the constraint quality factor Q when generating the subset of constraints C' to be passed to the max-product algorithm. This tournament selection works as follows. For each bid to generate, a number n_t of candidate constraint subsets are randomly generated. From these subsets, the one which maximizes the product of the quality factors Q of its constraints is chosen as the subset C' to be used for the max-product algorithm. In this way, since high-Q constraints are more likely to be selected, we expect the average Q for the resulting bids to be higher.

4 A Probabilistic Mechanism for Deal-Identification

Scalability is identified as one of the main drawbacks in a bidding based negotiation protocol [Ito *et al.*, 2007]. Once agents have placed their bids, the mediator performs an exhaustive search for overlaps between the bids using a breadth-first algorithm with branch cutting. In a worst case scenario, this means searching through a total of $n_b^{n_a}$ bid combinations, where n_b is the number of bids per agent, and n_a is the number of negotiating agents. In the experiments, the authors limit the number of combinations to $6,400,000$. This means that, for 4 negotiating agents, the maximum number of bids per agent is $\sqrt[n_a]{6400000} = 50$. This limit becomes harder as the number of agents increases. For example, for 10 agents, the limit is 4 bids per agent, which drastically reduces the probability of reaching a deal. This is specially true for highly-nonlinear utility spaces, where the bids are narrower.

To address this scalability limitation, we propose to perform a probabilistic search in the mediator instead of an exhaustive search. This means that the mediator will try a certain number n_{bc} of randomly chosen bid combinations, where $n_{bc} < n_b^{n_a}$. In this way, n_{bc} acts as a performance parameter in the mediator, which limits the computational cost of the deal identification phase. Of course, restricting the search for solutions to a limited number of combinations may cause the mediator to miss good deals. Taking this into account, the random selection of combinations is biased to maximize the probability of finding a good deal. Again, the parameter used to bias the random selection is Q, so that higher-Q bids have more probability of being selected for bid combinations at the mediator.

5 Experimental Evaluation

The hypothesis of this work is that the proposed mechanisms provide an improvement to the optimality of the negotiation process over the previous works described in Section 2. To evaluate this, we have performed a set of experiments to compare the results of the basic approaches with the results obtained introducing the quality factor Q in the bidding and deal identification mechanisms.

5.1 Experimental Settings

Several experiments have been conducted to validate our hypothesis. In each experiment, we ran 100 negotiations between agents with randomly generated utility functions. Each negotiation was run for each of the different approaches analyzed. For each set of utility functions we applied a non-linear optimizer to the product of all agents' utility functions to find the optimal contract using the *Nash* criterion and its associated joint utility value. As opposed to the optimization performed in the mediator during the negotiation, this optimal contract is computed using complete information about the agents preferences, so that it can be used to assess the optimality of the different approaches.

We ran experiments with the following parameters:

- Number of agents n_a = $\{4,...,14\}$. Number of issues n = $\{4,...,20\}$. Domain for issue values $[0, 9]$.

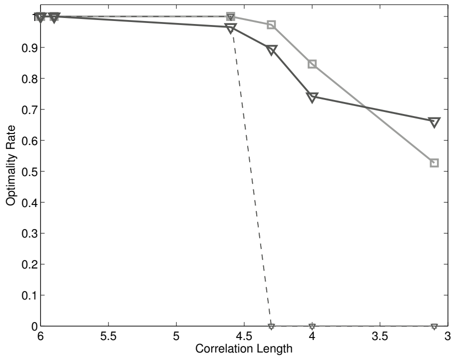

Figure 4: Optimality rate results for 6 agents and 6 issues with different constraint widths.

- l uniformly distributed random generated constraints per agent: 5 unary constraints, 5 binary contraints, 5 trinary constraints, etc.
- Utility for each m-ary constraint drawn from a uniform distribution in the domain $[0, 100 \times m]$.
- Different average widths for constraints, ranging from 2 to 7.
- Settings for simulated annealing: initial temperature $T_0 = 30$. Number of iterations: 30.
- Maximum number of bids generated per agent $n_b = 200 \times n$.
- Parameters for Q calculation: $\alpha = 0.5$.
- Parameters for tournament selection $n_c = \min(20, l/2)$, $n_t = 10$.
- Joint utility for a failed negotiation: 0.

Experiments were coded in Java and run on a 2x3.2Ghz Qad-Core Intel Xeon processor with 4Gb memory under Mac OS X 10.5.4.

5.2 Experimental Results

Figure 4 shows the results of 100 runs of the experiments for 6 agents and 6 issues. The vertical axis represents the median optimality rates of the experiments, while the horizontal axis represents the degree of non-linearity of the utility spaces of the agents, measured using a correlation length, which has been widely used to asses fitness landscape complexity in evolutionary computation [Weinberger, 1990]. Correlation length is defined as the minimum distance between samples in the utility space which makes the correlation between samples drop below a given threshold. For the purpose of this work, we have chosen a threshold of 0.7. Four sets of values have been represented: Q-based approaches are depicted using solid lines, while basic approaches are depicted using dashed lines. In each case, simulated annealing approaches have been represented using triangle vertices, while MWIS approaches have been represented using square vertices.

We can see that both basic contract sampling and basic constraint sampling yield high optimality rates for high correlation lengths, but the median optimality rate decreases drastically (in fact, it drops to zero) as the correlation length decreases (that is, for highly nonlinear utility spaces). The Q-based approaches yield slightly lower optimality rates for wider constraints, which is reasonable, since the Q is used to make a trade-off between utility and deal probability. However, as the agents' preference model turns highly nonlinear, introducing the quality factor Q in the bidding and deal identification mechanisms significantly outperforms the previous approaches, yielding acceptable optimality rates even with the lowest correlation lengths. From these results we can conclude that the quality factor Q can be used to improve failure rate in highly-nonlinear utility spaces, and both simulated annealing and tournament constraint selection with MWIS are suitable ways to select which constraints to use for bid generation.

Table 1: Scalability with the number of agents and issues

		n			
n_a	Approach	6	10	14	20
6	MWIS	0.9915	0.7521	0	0
	MWIS-Q	0.9757	0.9280	0.8964	0.8491
10	MWIS	0.6732	0	0	0
	MWIS-Q	0.9491	0.9329	0.9124	0.8331
14	MWIS	0	0	0	0
	MWIS-Q	0.9450	0.9420	0.8966	0.8313

Table 2: Performance comparison

Approaches	Time Ratio	
	median	conf. interval
T_b; SA-Q / SA	5.767	$[5.668, 5.866]$
T_b; MWIS-Q / MWIS	2.007	$[1.942, 2.071]$
T_d; Q-mediator / mediator	1.8060	$[0.597, 3.016]$
T_T; SA-Q / SA	4.947	$[4.675, 5.219]$
T_T; MWIS-Q / MWIS	1.748	$[1.514, 1.982]$
T_T; MWIS-Q / SA-Q	0.1688	$[0.1665, 0.1710]$

Regarding scalability, Table 1 shows the optimality rates obtained for one of the studied approaches (MWIS) for a fixed maximum constraint width (5) when the number of agents and issues increases, comparing the results obtained using the basic approach with those obtained introducing the quality factor. We can see that introducing the quality factor Q in the mechanisms significantly improves scalability with the number of agents and issues.

Finally, Table 2 shows the medians and their 95% confidence intervals for the ratio between the times of our proposed approaches and their corresponding basic approaches. Separate results are shown for the bidding time (T_b), the deal identification time (T_d), and the total negotiation time (T_T). We can see that for both contracts and constraints sampling the use of the quality factor Q introduces a certain overhead over the bidding and deal identification times, and that this overhead is specially significant in the case of simulated annealing bidding, due to the high number of times that the utility function must be evaluated. The comparison has been made for a high correlation distance (5), since for highly nonlinear utility spaces the basic approaches fail to find a solution.

6 Discussion and Related Work

In [Ito *et al.*, 2007], the authors propose a single-shot, auction-based protocol which samples the contracts space and uses simulated annealing to identify high utility regions in the agent's utility spaces to be sent as bids to a mediator. In [Marsa-Maestre *et al.*, 2008], instead of performing a direct sampling of the contract space, different techniques are used over the constraint space to generate bids. We use these works as a starting point to provide effective bidding and deal identification mechanisms for highly-nonlinear utility spaces, where the "narrowness" of the agents' high-utility regions makes the failure rate of their approaches drastically higher. Our approach is based on using a quality factor Q, which balances bid utility and bid volume to take into account the likelihood of the bid resulting in a deal. This is a somewhat similar approach to the notion of *viability* seen in [Lopez-Carmona and Velasco, 2006] for fuzzy-constraint based negotiation or the similarity criteria used in [Faratin *et al.*, 2000] for linear utility spaces. Graph theory is also used in the context of negotiation in [Robu *et al.*, 2005], where utility graphs are used to model issue inter-dependencies for binary-valued issues. We take advantage of graph theory in a different way, to perform a utility optimization in a constraint-based framework. There are other techniques for addressing non-linearity in negotiation. Instead of an auction based protocol, an alternating offers protocol is used in [Lai *et al.*, 2008] for unmediated bilateral negotiation. Other technique is to approximate the utility functions by means of linear regression techniques or average weighting methods, as proposed in [Hindriks *et al.*, 2006]. However, as authors acknowledge, these approaches are not useful for highly-nonlinear spaces.

7 Conclusions and Future work

The performance of existing auction-based approaches for negotiation in nonlinear scenarios dramatically decreases when confronted with highly nonlinear scenarios where the negotiating agents' high utility regions are very "narrow" and so it is very unlikely that high utility bids overlap. This paper presents a mechanisms to balance bid "width" and bid utility, and integrate this mechanism into two previous approaches. The experiments show that the proposed mechanisms significantly improve the previous approaches in highly nonlinear utility spaces in terms of failure rate and optimality. However, there is still plenty of research to be done in this area. The impact of the parameter α, which model the risk attitude of the agents, in the optimality rate should be analyzed, both to establish collective optimal strategies and to determine the incentive compatibility of the approach when agents involved in a negotiation may have different attitudes. Another challenge is the extension of the proposed approaches to handle other utility models not based on constraints, like quasi-concave functions. Finally, we are working on iterative negotiation protocols, where agents may change their attitudes or relax their bids as the protocol iterates.

Acknowledgments

This work has been partially developed in the framework of the ITEA-2 project 2008005, "Do-it-Yourself Smart Experiences", and partially supported by the Spanish Ministry of Education and Science grant TIN2008-06739-C04-04.

References

[Bayati *et al.*, 2008] M. Bayati, D. Shah, and M. Sharma. Max-product for maximum weight matching: Convergence, correctness, and lp duality. *IEEE Transactions on Information Theory*, 54(3):1241–1251, 2008.

[Faratin *et al.*, 2000] P. Faratin, C. Sierra, and N.R. Jennings. Using similarity criteria to make negotiation trade-offs. In *Proceedings of the 4th International Conference on Multi-Agent Systems*, pages 119–126, 2000.

[Hindriks *et al.*, 2006] Koen Hindriks, Catholijn M Jonker, and Dmytro Tykhonov. Eliminating interdependencies between issues for multi-issue negotiation. In *Cooperative Information Agents X*, volume 4149 of *Lecture Notes in Computer Science*, pages 301–316, 2006.

[Ito *et al.*, 2007] Takayuki Ito, Mark Klein, and Hiromitsu Hattori. Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI07)*, pages 1347–1352, 2007.

[Lai *et al.*, 2008] G. Lai, K. Sycara and C. Li A Decentralized Model for Multi-Attribute Negotiations with Incomplete Information and General Utility Functions. *Studies in Computational Intelligence*, 89:39–57, 2008.

[Lopez-Carmona and Velasco, 2006] Miguel A. Lopez-Carmona and Juan R. Velasco. An expressive approach to fuzzy constraint based agent purchase negotiation. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS06)*, 2006.

[Marsa-Maestre *et al.*, 2008] Ivan Marsa-Maestre, Miguel A. Lopez-Carmona, and Juan R. Velasco. Bidding mechanisms for nonlinear scenarios. Technical Report TR-2008-301, Telematic Engineering Research Group. Universidad de Alcala, 2008.

[Miller and Goldberg, 1995] B. L. Miller and D. E. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9(3):193–212, 1995.

[Nash, 1953] J.F. Nash. Two-person cooperative games. *Econometrica*, 21:128–140, 1953.

[Robu *et al.*, 2005] V. Robu, D.J.A. Somefun and J.A La Poutre Modeling Complex Multi-Issue Negotiations Using Utility Graphs. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS05)*, 2005.

[Weinberger, 1990] E.D. Weinberger. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biol. Cybern*, 63:325–336, 1990.

[Yokoo, 2001] M. Yokoo. *Distributed Constraint Satisfaction*. Springer, 2001.

[Zhang and Pu, 2004] J. Zhang and P. Pu. Survey on solving multi-attribute decision problems. Technical Report IC/2004/54, EPFL, 2004.