2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology

An Approach to Scalable Multi-issue Negotiation: Decomposing the Contract Space based on Issue Interdependencies

Katsuhide Fujita *, Takayuki Ito *† and Mark Klein†
* *Department of Computer Science and Engineering, Nagoya Institute of Technology, Gokiso, Showa-ku, Nagoya 466-8555, Japan, Email: fujita@itolab.mta.nitech.ac.jp, ito.takayuki@nitech.ac.jp*
† *Center for Collective Intelligence MIT Sloan School of Management, 5 Cambridge Center, Cambridge 02139, USA, Email: {takayuki, m_klein}@mit.edu*

***Abstract*—Most real-world negotiation involves multiple interdependent issues, which makes an agent's utility functions nonlinear. Traditional negotiation mechanisms, which were designed for linear utilities, do not fare well in nonlinear contexts. One of the main challenges in developing effective nonlinear negotiation protocols is scalability; they can produce excessively high failure rates, when there are many issues, due to computational intractability. One reasonable approach to reducing computational cost, while maintaining good quality outcomes, is to decompose the utility space into several largely independent sub-spaces. In this paper, we propose a method for decomposing a utility space based on four types of issue interdependencies. This method allows good outcomes with greater scalability than previous efforts. We also analyze how the types of issue interdependency influence the solution optimality and failure rate.**

***Keywords*-Multi-issue negotiation; Nonlinear utility function; Bargaining and negotiation**

I. Introduction

Negotiation is an important aspect of daily life and represents an important topic in the field of multi-agent systems research. There has been extensive work in the area of automated negotiation; that is, where automated agents negotiate with other agents in such contexts as e-commerce [1], large-scale argumentation [2], collaborative design, and so on. Even though many contributions have been made in this area [3] most have dealt exclusively with simple negotiations involving one or more independent issues. Many real-world negotiations, however, are complex and involve interdependent issues. When designers work together to design a car, for example, the utility of a given carburetor is highly dependent on which engine is chosen. The key impact of such issue dependencies is that they result in agent utility functions that are nonlinear, i.e. that have multiple optima. Most existing negotiation protocols, though well-suited for linear utility functions, work poorly when applied to nonlinear problems [4].

Recently, some studies have focused on negotiation with nonlinear utility functions. The following are the representative studies on multiple issues negotiations for complex utility spaces: A bidding-based protocol was proposed in [5]. Agents generate bids by finding high regions in their own utility functions, and the mediator finds the optimum combination of submitted bids from the agents. In [6], the representative based protocol for reducing the computational cost was proposed. In this method, the scalability of agents was improved, however, the scalability of issues was not enough. In [7], utility graphs were used to model issue dependencies for binary-valued issues. [8] proposed an approach based on a weighted approximation technique to simplify the utility space. [9] proposed bilateral multi-issue negotiations with time constraints. [10] proposed an auction-based protocol for nonlinear utility spaces generated using weighted constraints, and [11] extended this work to address highly-rugged utility spaces. However, unsolved problem is the scalability of the protocols against the number of issues. Thus, reducing this computational cost has been a key focus in this research.

We propose a new protocol in which a mediator tries to reorganize a highly complex utility space into several tractable utility subspaces, in order to reduce the computational cost. Issue groupings are generated by a mediator based on an examination of the issue interdependencies. First, we have to define a measure for the degree of interdependency between issues. In this paper, we define four such measures. Second, we generate a weighted non-directed interdependency graph based on this information. By analyzing the interdependency graph, a mediator can identify issue subgroups. Note that while others have discussed issue interdependencies in utility theory [12], this previous work doesn't identify optimal issue groups. Finally, we demonstrate that our protocol, based on issue-groups, has higher scalability than previous efforts, and discuss the impact on the optimality of the negotiation outcomes.

The remainder of this paper is organized as follows. First, we describe a model of nonlinear multi-issue negotiation and utility functions. Second, we describe several measures for assessing the degree of issue interdependency, present a technique for finding issue sub-groups, and propose a protocol that uses this information to enable more scalable negotiations. Third, we present the experimental results. Finally, we describe related works and draw conclusions.

978-0-7695-4191-4/10 $26.00 © 2010 IEEE
DOI 10.1109/WI-IAT.2010.91

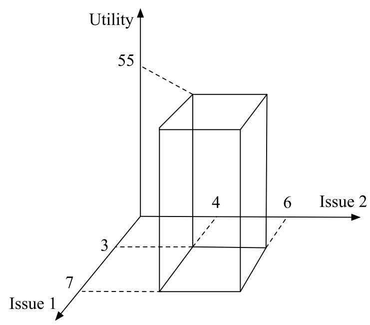

Figure 1. Example of constraint

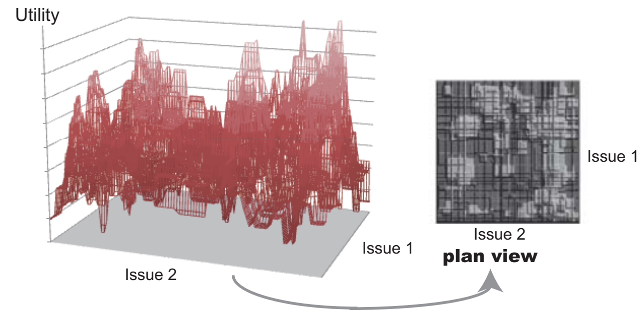

Figure 2. An example of a nonlinear utility space

II. Negotiation with Nonlinear Utility Functions

We consider the situation where N agents $(a_1, \ldots, a_N)$ want to reach an agreement with a mediator who manages the negotiation from a man-in-the-middle position. There are M issues $(i_1, \ldots, i_M)$ to be negotiated. The number of issues represents the number of dimensions in the utility space. The issues are shared: all agents are potentially interested in the values for all M issues. A contract is represented by a vector of issue values $\vec{s} = (s_1, ..., s_M)$. Each issue s_j has a value drawn from the domain of integers $[0, X]$, *i.e.*, $s_j \in \{0, 1, , \ldots, X\}(1 \leq j \leq M)$. We assume that agents have an incentive to cooperate to achieve win-win agreements because a non-agreement has lower utility than an agreement.

An agent's utility function, in our formulation, is described in terms of constraints. There are l constraints, $c_k \in C$. Each constraint represents a region in the contract space with one or more dimensions and an associated utility value. Constraint c_k has a value $v_a(c_k)(1 \leq k \leq l)$. In addition, c_k has value $w_a(c_k, \vec{s})$ if and only if it is satisfied by contract $\vec{s}$. Function $\delta_a(c_k, i_j)$ is a region of i_j in c_k, and function $\epsilon_a(c_k)$ is the number of terms in c_k. Actually, function $\epsilon_a(c_k)$ is $\emptyset$ if c_k has no region regarded as i_j. Every agent has its own, typically unique, set of constraints.

An agent's utility for contract $\vec{s}$ is defined as the sum of the utility for all the constraints it satisfies, i.e., as $u_a(\vec{s}) = \sum_{c_k \in C, \vec{s} \in x(c_k)} w_a(c_k, \vec{s})$, where $x(c_k)$ is a set of possible contracts (solutions) of c_k. This expression produces a "bumpy" nonlinear utility function with high points where many constraints are satisfied and lower regions where few or no constraints are satisfied. This represents a crucial departure from previous efforts on multi-issue negotiation, where contract utility is calculated as the weighted sum of the utilities for individual issues, producing utility functions shaped like flat hyper planes with a single optimum.

Figure 2 shows an example of a utility space generated via a collection of binary constraints involving Issues 1 and 2. In addition, the number of terms is two in Figure 2. Figure1, for example, which has a value of 55, holds if the value for Issue 1 is in the range $[3, 7]$ and the value for Issue 2 is in the range $[4, 6]$. The utility function is highly nonlinear with many hills and valleys. For our work, we assume that many real-world utility functions are more complex than this, involving more than two issues as well as higher-order (e.g. trinary and quaternary) constraints. In recent work (e.g. [10], [13]), several types of constraints were proposed.

This constraint-based utility function representation allows us to capture the issue interdependencies common in real world negotiations. The constraint in Figure 2, for example, captures the fact that a value of 4 is desirable for issue 1 if issue 2 has the value 4, 5 or 6. Note, however, that this representation is also capable of capturing linear utility functions as a special case (they can be captured as a series of unary constraints). A negotiation protocol for complex contracts can, therefore, handle linear contract negotiations.

The objective function for our protocol can be described as follows:

$$\arg\max_{\vec{s}} \sum_{a \in N} u_a(\vec{s}).$$

Our protocol, in other words, tries to find contracts that maximize social welfare, *i.e.*, the total utilities for all agents. Such contracts, by definition, will also be Pareto-optimal.

It is of course theoretically possible to gather all the individual agents' utility functions into one central place and then find all optimal contracts using such well-known nonlinear optimization techniques as simulated annealing or evolutionary algorithms. However, we do not employ such centralized methods for negotiation purposes because we assume, as is common in negotiation contexts, that agents prefer not to share their utility functions with each other, in order to preserve a competitive edge.

Note that, in negotiations with multiple *independent* issues, we can find the optimal value for each issue in isolation to quickly find a globally optimal negotiation outcome. In negotiation with multiple *interdependent* issues, however, the mediator can't treat issues independently because the utility of a choice for one issue is potentially influenced

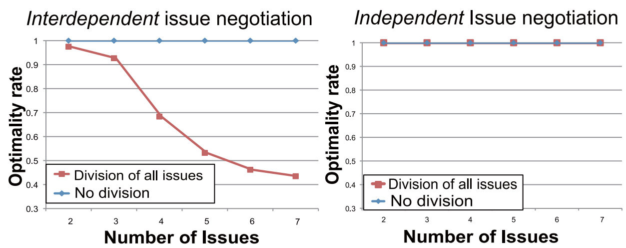

Figure 3. Relationship of interdependency and optimality rate with nonlinear utility function

ID	Issue1	Issue2	Issue3	Issue4	Utility
1	[2, 4]	$\emptyset$	[4,6]	$\emptyset$	20
2	$\emptyset$	5	[3,7]	[1,6]	40
3	[3,8]	$\emptyset$	$\emptyset$	$\emptyset$	25
4	4	[2,7]	9	[4,5]	50

Table I
UTILITY FUNCTION FOR AN AGENT

by the choices made for other issues. Figure 3 shows the relationship between issue interdependency and negotiation optimality in an example with interdependent issues. In figure 3, we ran an exhaustive social welfare optimizer for each issue independently, as well as for all possible issue combinations. The number of agents is four, and the domain of per issues is five. The linear utility function (independent cases) is generated by $u_a(x) = k * x + c$ (where x is the value for that issue, k and c are constants, and a is the agent). The nonlinear function is generated by 10 unary constraints, 5 binary constraints. If the mediator ignores the issue interdependencies (i.e. finds optima for each issue in isolation), optimality declines rapidly as the number of issues increases. This means that the mediator must account for issue interdependencies in order to find high quality solutions. But if the negotiation protocol tries to do so by exhaustively considering all issue-value combinations, it quickly encounters intractable computational costs. If we have, for example, only 10 issues with 10 possible values per issue, this produces a space of 10^{10} (10 billion) possible contracts, which is too large to evaluate exhaustively. Negotiation with multiple interdependent issues thus introduces a difficult tradeoff between optimality and computational cost.

III. INTERDEPENDENCY RATE AND INTERDEPENDENCY GRAPH

A issue interdependency for multi-issue negotiations is defined as follows: If there is a constraint between issue X (i_X) and issue Y (i_Y), then we assume i_X and i_Y are interdependent. If, for example, an agent has a binary constraint between issue 1 and issue 3, issue 1 and issue 3 are interdependent for that agent - see Table I.

The strength of issue interdependency is measured by *interdependency rate*. We define four measures for the interdependency between $issue\ i_j$ and $issue\ i_{jj}$ for agent a:

(A) Number of constraints only: $D_a^{(A)}(i_j, i_{jj}) = \sharp\{c_k | \delta_a(c_k, i_j) \neq \emptyset \ \wedge \ \delta_a(c_k, i_{jj}) \neq \emptyset\}$. This measures the number of constraints that inter-relate the two issues.

(B) Number of terms of constraints: $D_a^{(B)}(i_j, i_{jj}) = \sum_{c_k \in C} \epsilon_a(c_k)\ if\ c_k\ is\ \delta_a(c_k, i_j) \neq \emptyset \ \wedge \ \delta_a(c_k, i_{jj}) \neq \emptyset$. This sums the order of the constraints relating two issues, based on the intuition that higher-order constraints are more important than lower-order (e.g. binary) constraints.

(C) Utility value of constraints: $D_a^{(C)}(i_j, i_{jj}) = \sum_{c_k \in C} v_a(c_k)\ if\ c_k\ is\ \delta_a(c_k, i_j) \neq \emptyset \wedge \delta_a(c_k, i_{jj}) \neq \emptyset$. This sums the weights of the constraints that inter-relate the two issues.

(D) Number of terms and utility of constraints: $D_a^{(D)}(i_j, i_{jj}) = D_a^{(B)}(i_j, i_{jj}) * D_a^{(C)}(i_j, i_{jj})$. This is the product of measures B and C. In addition, we assume that $D_a^{(B)}(i_j, i_{jj})$ and $D_a^{(C)}(i_j, i_{jj})$ are normalized.

The agents capture issue interdependency information as an interdependency graph. An interdependency graph is represented as a weighted non-directed graph, in which a node represents an issue, an edge represents the interdependency between issues, and the weight of an edge represents the interdependency rate between the issues. An interdependency graph is thus formally defined as:

$$G(P, E, w) : P = \{1, 2, \ldots, |I|\}(finite\ set),$$
$$E \subset \{\{x, y\} | x, y \in P\}, w : E \to R.$$

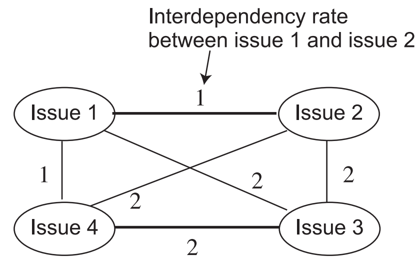

Figure 4. Interdependency Graph

Figure 4 shows the interdependency graph for the constraints listed in Table 4.

IV. NEGOTIATION PROTOCOL BASED ON ISSUE INTERDEPENDENCY

Our proposed negotiation protocol works as follows. A mediator gathers private issue interdependency graphs from each agent, generates a social interdependency graph, identifies issue sub-groups, and then uses that information to guide the search for a final agreement. In fact, we apply the concept of issue-grouping to a bidding based approach[5] in our negotiation protocol. In bidding based approach,

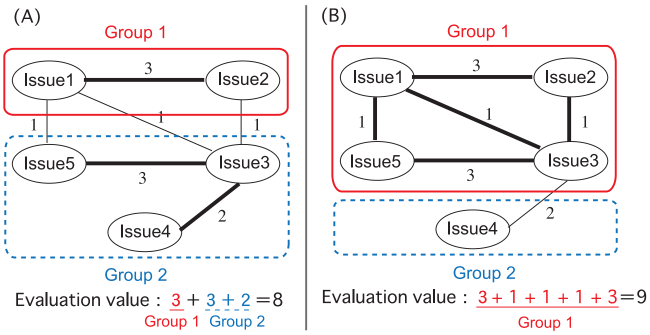

Figure 5. Evaluation value in identifying issue-groups

agents can make agreement without submitting all agents' privacy information, however, the scalability is not enough. By applying the concept of grouping-issues to bidding based approach, we can propose high scalable protocol considering the agents' privacy. We describe the details below:

[Step 1: Analyzing issue interdependency] Each agent analyzes issue interdependency in its own utility space, using Algorithm 1, and generates an interdependency graph. Each agent sends its' interdependency graph to the mediator.

Algorithm 1 get_Interdependency(C)

C: a set of constraints

for c $\in$ C **do**
  for $i := 0$ **to** $Number\ of\ issues$ **do**
    for $j := i + 1$ **to** $Number\ of\ issues$ **do**
      if Issue i and Issue j are interdependent in c **then**
        $Calculate$ interdependencyGraph[i][j]
      end if
    end for
  end for
end for

[Step 2: Grouping issues] In this step, the mediator identifies the issue-groups. First, the mediator generates a social interdependency graph from the private interdependency graphs submitted by the agents. A social interdependency graph is almost same as a private interdependency graph. The only difference is that the weight of an edge represents the *social* interdependency rate. The social interdependency rate between $issue\ i_j$ and $issue\ i_{jj}$ is defined as: $\sum_{a \in N} D_a(i_j, i_{jj})$. ($D_a(i_j, i_{jj})$: Interdependency rate between $issue\ i_j$ and $issue\ i_{jj}$ by agent a).

Next, the mediator identifies the issue-groups based on the social interdependency graph. In this protocol, the mediator tries to find optimal issue-grouping using simulated annealing (SA) [14]. The evaluation function for the simulated annealing is the sum of the weights of the edges that do not span separate issue-groups. The goal is to maximize this value. Figure 5 shows an example of evaluation values for two issue-groups. In Figure 5 (A), the evaluation value is 8 because there are non-spanning edges between issue 1 and issue 2, issue 3 and issue 4, issue 3 and issue 5, and issue 4 and issue 5. In Figure 5 (B), the evaluation value is 9 because there are non-spanning edges among issue 1, issue 2, issue 3, and issue 5. The number of issue-groups is decided before the protocol begins.

Agents are at risk for making an agreement that is not optimal for themselves by dividing the interdependent issues. In other words, there is the possibility of making a low utility agreement by ignoring the interdependency of some issues. However, agents can make a better agreement in this protocol because the mediator identifies the issue-groups based on the rate of interdependency.

[Step 3: Generating bids] First, each agent generates bids for the entire set of issues using the bidding-based protocol [5]. Concretely speaking, each agent samples its entire utility space in order to find high-utility contract regions. After that, each agent uses a nonlinear optimizer based on simulated annealing [14] to try to find the local optimum in its neighborhood. For each contract $\vec{s}$ found by adjusted sampling, an agent evaluates its utility by summation of values of satisfied constraints. If that utility is larger than the reservation value δ(threshold), then the agent defines a bid that covers all the contracts in the region that have that utility value.

Next, agents divide these bids into sub-bids for each issue-group, and determine their valuations for each sub-bid. In this paper, agents set their valuation for a bid to be the utility of the highest-value contract in the bid region. In Figure 6, for example, an agent selects the global bid $B_{all} = [1, 2, 3]$ for all issues, and divides B_{all} into sub-bids $B_1 = [1, X, X]$ for issue group 1 and $B_2 = [X, 2, 3]$ for issue group 2 (X: any value). In this case, the agent's evaluations for both sub-bids are 9.

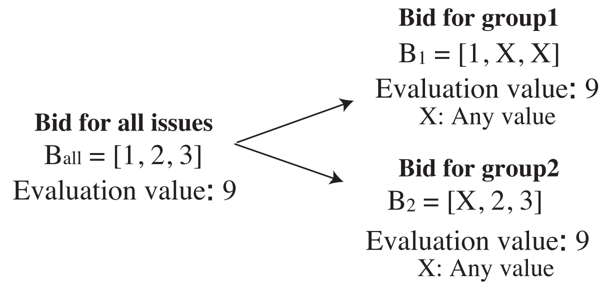

Figure 6. Division for the bid by agents

[Step 4: Finding the Solutions] The mediator identifies the final contract by finding all the combinations of bids, one from each agent, that are mutually consistent, *i.e.*, that specify overlapping contract regions[1]. If there is more than

[1] A bid can specify not just a specific contract but an entire region. For example, if a bid covers the region [0,2] for issue 1 and [3,5] for issue 2, the bid is satisfied by the contract where issue 1 has value 1 and issue 2 has value 4. For a combination of bids to be consistent, the bids must all overlap.

one such overlap, the mediator selects the one with the highest social welfare (i.e. the highest summed bid value). The mediator employs breadth-first search with branch cutting to find the social-welfare-maximizing bid overlaps. After that, the mediator finds the final contract by consolidating the winning sub-contracts from each issue-group.

In terms of an agent's strategic behavior, we assume agents are truthful in this paper. In addition, theoretically, our protocol can be made incentive-compatible (i.e. where agents are given incentive to provide the truthful bid values that are necessary to ensure [near-]optimal social welfare) if we employ the Groves mechanism [15] with some theoretical assumptions on unlimited budgets and unlimited computational resources. Also, we must assume that the cost (payment) does not depend on the other issues. Then, we can define agent i's utility function as follows: $u_i = v_i - c_i$, where v_i is value of agreement when some multiple issues are satisfied and c_i is the payment computed by one of the Grove's mechanisms. We describe the details in the appendix.

V. Experimental Results

A. Setting

We conducted several experiments to evaluate our approach. In each experiment, we ran 100 negotiations. The following parameters were used. The domain for the issue values was $[0, 9]$. The number of constraints was 10 unary constraints, 5 binary constraints, 5 trinary constraints, and so on. (a unary constraint relates to one issue, a binary constraint relates to two issues, etc). The maximum value for a constraint was *100 × (Number of Issues)*. Constraints that satisfy many issues have, on average, larger utility, which seems reasonable for many domains. In the meeting scheduling domain, for example, higher order constraints concern more people than lower order constraints, so they are more important. The maximum width for a constraint was 7. The following constraints would all be valid: Issue 1 = $[2, 6]$, Issue 3 = $[2, 9]$.

We compare the following six methods: "(a) Issue-groups (Number of constraints)," "(b) Issue-groups (Number of terms)," "(c) Issue-groups (utility)," "(d) Issue-groups (terms & utility)," "(e) Basic Bidding," and "(f) Q-Factor." (a)-(d) are variants of the issue-group protocol proposed in this paper, using the four different interdependency rate measures $D_n^{(A)} \sim D_n^{(D)}$ we described above. This allows us to compare the efficacy of the different interdependency rate measures. "(e) Basic Bidding" is the bidding-based protocol proposed in [5], which does not employ issue-grouping. In this protocol, agents generate bids by finding the highest utility regions in their utility functions, and the mediator finds the optimum combination of bids submitted from agents. "(f) Q-Factor" is the Maximum Weight Interdependent Set (MWIS) protocol proposed in [10], [11]. MWIS is a variant of bidding protocol where agents use the Q-factor, a combination of region and utility, to decide which bids to submit. This reduces the failure rate because agents are less likely to submit low-volume bids that do not overlap across agents.

The parameters for generating bids in (a)-(f) are as follows [5]. The number of samples taken during random sampling is *(Number of Issues) × 200*. The starting temperature for the simulated annealing algorithm used to find high points near the samples is 30 degrees . For each iteration, the temperature decreases 1 degree, so the annealer runs for 30 iterations. Note that it is important that the annealer does not run too long or too hot because then each search will tend to find the global optimum instead of the peak of the optimum nearest the sampling point. The threshold used to cut out contract points that have low utility is 100. The limitation on the number of bids per agent is $\sqrt[N]{6,400,000}$ for N agents, because it was only practical to run the deal identification algorithm if it explored no more than about 6,400,000 bid combinations. The reservation value for generating bids is 100. The parameters for identifying issue sub-groups, in (a)-(d), are as follows. The initial temperature for the simulated annealing algorithm is 30 degrees. For each iteration, the temperature decreased 3 degrees, producing a total of 10 iterations. The number of issue-groups generated is three. In "(f) Q-Factor," Q (Q-Factor) is defined as $Q = u^{\alpha} * v^{\beta}$($u$: utility value, v: volume of the bid or constraint), $\alpha = 0.5,\ \beta = 0.5$.

We used simulated annealing (SA) [14] to approximate the optimum social welfare for each negotiation test run. Exhaustive search was not a viable option because it becomes computationally intractable as the number of issues grows. The SA initial temperature is 50.0 and decreases linearly to 0 over the course of 2,500 iterations. The initial contract for each SA run is randomly selected. The optimality value for a negotiation run, in our experiments, is defined as *(The social welfare achieved by each protocol) / (The social welfare calculated by SA)*.

Our code is implemented in Java 2 (1.5) and run on a core 2-duo CPU with 1.0 GB memory on a Mac OS X (10.6).

B. Experimental Results

Figure 7 compares the optimality rate of the different protocols. The lines represent the min and max values, the boxes represent +/- *1* standard deviation, and the '-' represents the average. This results are counted when the negotiations don't fail. The optimality rate of our method ((a)-(d)) is higher than "(f) Q-Factor" when the number of issues is large. In addition, "(d) Issue-Groups (terms & utility)" produces a higher optimality rate than (a). In *t-test*, there is a significant difference between (a) and (d) in case 5 ($t(198) = 0.003, P < 0.05$, one-sided testing). Therefore, the interdependency rate measure based on constraint utility and number of constraint terms works best of those we

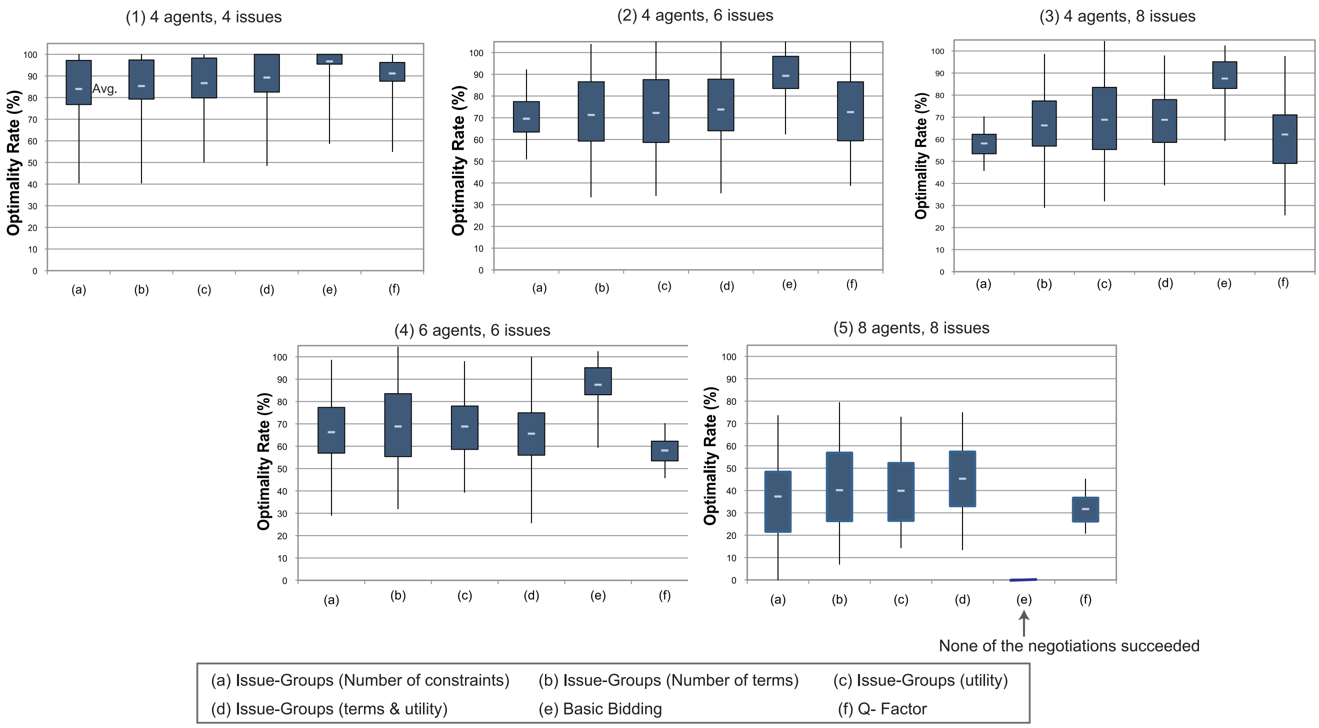

Figure 7. Box-plots of the optimality rate

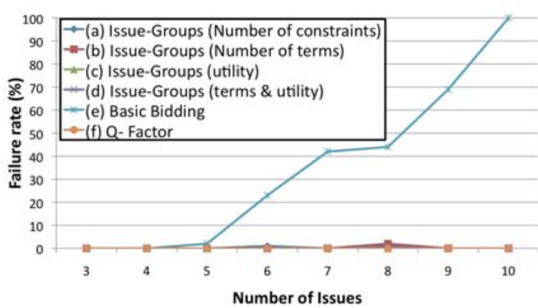

Figure 8. Failure rate

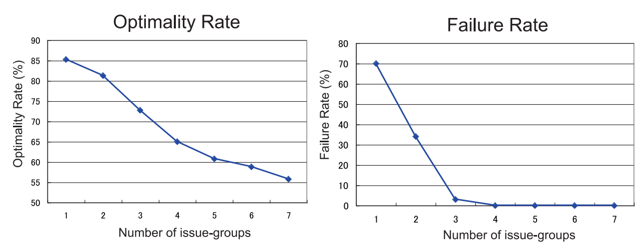

Figure 9. Effect of the number of issue-groups

tried. "(e) Basic Bidding" produces the highest optimality scores in case 1 and case 2 where it does not fail. This is because that (e) doesn't generate the issue-groups. However, (e) succeeded for none of the negotiations in case 3, so it's scalability is limited.

Figure 8 compares the failure rates. The failure rate of our method ((a)-(d)) is lower than "(e) Basic Bidding", especially as the number of issues increases. Also, our method ((a)-(d)) has essentially the same (very low) failure rate as "(f) Q-Factor." Our proposed method and Q-Factor thus achieve the same reduction in failure rate by different means: one by negotiating by issue-groups, the other by bidding based on the quality factor. It should be noted, however, that the Q-factor approach is probably *not* incentive-compatible. While using the Q-factor to pick bids does reduce the failure rate, there is an incentive for agents to cheat and submit bids based only on their utility. This increases the likely utility of the final deal, for them, and may not substantially increase the probability of a failed negotiation *if the other agents do not cheat as well*. This thus creates an prisoner's dilemma game, such that all agents are individually incented to take actions that make things worse for everybody. Our issue-clumping protocol, by contrast, does not require that agents selflessly prefer higher volume bids, and thus avoids this incentive compatibility problem.

Figure 9 shows the optimality rate and failure rate as a function of the number of issue subgroups in our protocol, for experiments with four agents. The optimality rate decreases as the number of issue subgroups increases. This is because the possibility that important interdependencies cut across issue subgroups (and are thus ignored) increases when there are more subgroups. On the other hand, the failure rate for making agreements decreases as the number of issue subgroups increases. This is because the number of issues in each issue subgroup decreases, and the computational cost for finding agreements becomes smaller, thereby reducing the likelihood of missing an agreement and therefore having a failed negotiation. Thus, there is a trade-off between the optimality rate and the failure rate in selecting the number of issue groups in our protocol.

VI. Related Work

Even though negotiation seems to involve a straightforward distributed constraint optimization problem [16], [17], we have been unable to exploit existing work on high efficiency constraint optimizers. Such solvers attempt to find the solutions that maximize the weights of the satisfied constraints, but do not account for the fact that the final solution must satisfy at least one bid/constraint *from every agent*.

[18] explored a range of protocols based on mutation and selection on binary contracts. This paper does not describe what kind of utility function is used, nor does it present any experimental analyses, so it remains unclear whether this strategy enables sufficient exploration of utility space.

[4] presented a protocol applied with near optimal results to medium-sized bilateral negotiations with binary dependencies, but was not applied to multilateral negotiations and higher order dependencies.

[8] proposed an approach based on a weighted approximation technique to simplify the utility space. The resulting approximated utility function without dependencies can be handled by negotiation algorithms that can efficiently deal with independent multiple issues, and has a polynomial time complexity. Our protocol can find an optimal agreement point if agents don't have in common the expected negotiation outcome.

[19], [9] proposed bilateral multi-issue negotiations with time constraints. This method can find approximate equilibrium in polynomial time where the utility function is nonlinear. However, this paper focused on bilateral multi-issue negotiations. Our protocol focuses on multilateral negotiations.

[20] presents an axiomatic analysis of negotiation problems within task-oriented domains (TOD). In this paper, three classical bargaining solutions (Nash solution, Egalitarian solution, Kalai-Smorodinsky solution) coincide when they are applied to a TOD with mixed deals but diverge if their outcomes are restricted to pure deals.

[10], [11] proposed an auction-based protocol for nonlinear utility spaces generated using weighted constraints, and proposed a set of decision mechanisms for the bidding and deal identification steps of the protocol. They proposed the use of a quality factor to balance utility and deal probability in the negotiation process. This quality factor is used to bias bid generation and deal identification taking into account the agents' attitudes towards risk. The scalability on the number of issues is still problem in these works.

In [7], [21], utility graphs were used to model issue dependencies for binary-valued issues. Our utility model is more general.

VII. Conclusion

In this paper, we proposed a new negotiation protocol, based on grouping issues, which can find high-quality agreements in interdependent issue negotiation. In this protocol, agents generate their private issue interdependency graphs, the mediator identifies the issue-groups based on these graphs, and multiple independent negotiations proceed for each issue sub-group. We demonstrated that our proposed protocol has greater scalability than previous work, and analyzed the effectiveness of different measures of the interdependency rate.

For future work, we will investigate how to improve optimality while maintaining the failure rate advantages of our protocol. One possible track, for example, is to select the number of issue groups adaptively based on the issue dependency topology. Another is to conduct additional negotiation, after the concurrent sub-contract negotiations, to try to increase the satisfaction of constraints that crossed sub-contract boundaries.

References

[1] S. Kraus, *Strategic Negotiation in Multiagent Environments*. Cambridge University Press, 2001.

[2] T. W. Malone and M. Klein, "Harnessing collective intelligence to address global climate change," *Innovations Journal*, vol. 2(3), pp. 15–26, 2007.

[3] P. Faratin, C. Sierra, and N. R. Jennings, "Using similarity criteria to make issue trade-offs in automated negotiations," in *Artificial Intelligence*, 2002, pp. 142:205–237.

[4] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam, "Negotiating complex contracts," *Group Decision and Negotiation*, vol. 12, no. 2, pp. 58–73, 2003.

[5] T. Ito, H. Hattori, and M. Klein, "Multi-issue negotiation protocol for agents : Exploring nonlinear utility spaces," in *Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI-2007)*, 2007, pp. 1347–1352.

[6] K. Fujita, T. Ito, and M. Klein, "A representative-based multi-round protocol for multi-issue negotiations," in *Proc. of the Seventh Inernational Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-2008)*, 2008, pp. 1573–1576.

[7] C. M. Jonker, V. Robu, and J. Treur, "An agent architecture for multi-attribute negotiation using incomplete preference information," *Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)*, vol. 15, pp. 221–252, 2007.

[8] K. Hindriks, C. Jonker, and D. Tykhonov, "Eliminating interdependencies between issues for multi-issue negotiation," *Cooperative Information Agents X, Lecture Notes in Computer Science*, vol. 4149, pp. 301–316, 2006.

[9] S. S. Fatima, M. Wooldridge, and N. R. Jennings, "An analysis of feasible solutions for multi-issue negotiation involving nonlinear utility functions," in *Proc. of the Eighth International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-2009)*, 2007, pp. 1041–1048.

[10] I. Marsa-Maestre, M. A. Lopez-Carmona, J. R. Velasco, and E. de la Hoz, "Effective bidding and deal identification for negotiations in highly nonlinear scenarios," in *Proc. of the Eighth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-2009)*, 2009, pp. 1057–1064.

[11] I. Marsa-Maestre, M. A. Lopez-Carmona, J. R. Velasco, T. Ito, M. Klein, and K. Fujita, "Balancing utility and deal probability for negotiations in highly nonlinear utility spaces," in *Proc. of the Twenty-first International Joint Conference on Artificial Intelligence (IJCAI-2009)*, 2009, pp. 214–219.

[12] H. Tamura and Y. Nakamura, "Decompositions of multiattribute utility functions based on convex dependence," *Operations Research*, vol. 31, no. 3, pp. 488–506, 1983.

[13] K. Fujita, T. Ito, and M. Klein, "A secure and fair negotiation protocol in highly complex utilityspace based on cone-constraints," in *Proc. of the 2009 International Joint Conference on Intelligent Agent Technology (IAT-2009)*, 2009.

[14] S. J. Russell and P. Norvig, *Artificial Intelligence : A Modern Approach*. Prentice Hall, 2002.

[15] T. Groves, "Incentives in teams," *Econometrica*, vol. 41, no. 4, pp. 617–631, 1973.

[16] R. Greenstadt, J. Pearce, and M. Tambe, "Analysis of privacy loss in distributed constraint optimization," in *Proc. of the 21th Association for the Advancement of Artificial Intelligence (AAAI-2006)*, 2006, pp. 647–653.

[17] R. T. Maheswaran, J. P. Pearce, P. Varakantham, and E. Bowring, "Valuations of possible states (vps):a quantitative framework for analysis of privacy loss among collaborative personal assistant agents," in *Proc. of the Forth Inernational Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-2005)*, 2005, pp. 1030–1037.

[18] R. J. Lin and S. T. Chou, "Bilateral multi-issue negotiations in a dynamic environment," in *Proc. of the AAMAS Workshop on Agent Mediated Electronic Commerce (AMEC-2003)*, 2003.

[19] S. S. Fatima, M. Wooldridge, and N. R. Jennings, "Approximate and online multi-issue negotiation," in *Proc. of the 7th International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-2007)*, 2007, pp. 947–954.

[20] D. Zhang, "Axiomatic characterization of task oriented negotiation," in *Proc. of Twenty-first International Joint Conference on Artificial Intelligence (IJCAI-2009)*, 2009, pp. 367–372.

[21] V. Robu, D. J. A. Somefun, and J. L. Poutre, "Modeling complex multi-issue negotiations using utility graphs," in *Proc. of the 4th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2005)*, 2005, pp. 280–287.

[22] L. M. Ausubel and P. Milgrom, "Ascending auctions with package bidding." *Frontiers of Theoretical Economics*, vol. 1, no. 1, 2002.

APPENDIX: INCENTIVE COMPATIBILITY

Our negotiation protocol can be made incentive compatible by defining payments for agents and employing Groves mechanism[15]. We assume unlimited agent budgets, which is a standard assumption for these kinds of incentive analyses [22]. We also assume each agent knows its own utility space completely and can find the optimal points without any cost. We call the new mechanism (protocol) $\mathcal{M}$. We define agent is type θ_i to be a set of constraints C_i and its value w_i: $\theta_i = (C_i, w_i)$, where $w_i = \sum_{c \in C_i} w(c)$. θ_i can be viewed as a bid from agent i. In this mechanism, agent i submits type $\hat{\theta}(a\ bid)$, which may not be true (i.e. may not represent the true weight for those constraints). Based on the reported types $\theta = (\theta_1, \ldots, \theta_N)$, our mechanism computes: $s^*(\hat{\theta}) = arg_{s \in S, s\ is\ consistent} max \sum_i z_i(s, \hat{\theta}_i)$, where S is a set of contracts, $z_i(s, \hat{\theta}_i)$ is agent i's valuation function on the consistent contract s when i reports $\hat{\theta}_i$. s does not violate any constraints in $\hat{\theta}$. $z_i(s, \hat{\theta}_i))$ is a nonlinear function in our case. For the purpose of this analysis, we will assume an ideal case in which each agent has complete knowledge on his/her own utility space. We define agent is payments as follows a direct adaptation of Groves mechanism: $t_i(\hat{\theta}) = h_i(\hat{\theta}_{-i}) - \sum_{j \neq i} z_j(s^*(\hat{\theta}), \hat{\theta}_j) - (1)$ The first term, $h_i(\hat{\theta}_{-i})$, in the right hand in the equation (2) is an arbitrary function on the reported types of every agent except i. Agent i's utility for making a bid (i.e. reporting a type) $\hat{\theta}_i$ can be defined as follows: $u_i^{\mathcal{M}}(\hat{\theta}_i) = z_i(s^*(\hat{\theta}), \theta_i) - t_i(\hat{\theta}) - (2)$

Proposition 1 (Incentive compatibility). $\mathcal{M}$ is incentive compatible (i.e. truth telling is a dominant strategy).

Proof. The proof is almost the same as that for Grove's mechanism. Based on the utility function (2), $u_i^{\mathcal{M}}(\hat{\theta}_i) = z_i(s^*(\hat{\theta}), \theta_i) - t_i(\hat{\theta}) = z_i(s^*(\hat{\theta}_i), \theta_i) + \sum_{j \neq i}(s^*(\hat{\theta}), \hat{\theta}_j) - h_i(\hat{\theta}_{-i})$. Agent i can not control $h_i(\hat{\theta}_{-i})$. Therefore he wants to maximize $z_i(s^*(\hat{\theta}_i), \theta_i) + \sum_{j \neq i}(s^*(\hat{\theta}), \hat{\theta}_j)(*)$. On the other hand, mechanism $\mathcal{M}$ computes the following because to maximize social welfare efficiency: $argmax_{s \in S} z_i(s, \hat{\theta}_i)$. This can be written as follows: $argmax_{s \in S}[z_i(s, \hat{\theta}_i) + \sum_{j \neq i} z_j(s, \hat{\theta}_j)]$. For agent i, to maximize the equation (*), he must report $\hat{\theta}_i = \theta_i$, i.e. his truthful type.