Using Iterative Narrowing to Enable Multi-Party Negotiations with Multiple Interdependent Issues

Hiromitsu Hattori Mark Klein
{hatto,m_klein}@mit.edu
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02142-1347 USA

Takayuki Ito
ito.takayuki@nitech.ac.jp
Dept. of Computer Science and Engineering,
Graduate School of Engineering
Nagoya Institute of Technology
Nagoya, Aichi 466-8555 JAPAN

ABSTRACT

Multi-issue negotiations are a central part of many coordination challenges, and thus represent an important research topic. Almost all previous work in this area has assumed that negotiation issues are independent, but this is rarely the case in real-world contexts. Our work focuses on negotiation with interdependent issues and, therefore, nonlinear (multi-optimum) agent utility functions. Since the utility functions are typically very complex, the challenge becomes finding high-quality negotiation outcomes without making unrealistic demands concerning how much agents reveal about their utilities. Since negotiations often involve more than two parties, the approach should also be scalable. In this paper, we propose a novel protocol for addressing these challenges, wherein agents approach agreements by iteratively narrowing the space of possible agreements. In the early stages, agents submit rough bids representing promising regions from their utility functions. In later stages, they submit increasingly narrow bids for the subset of those regions that the negotiating parties all liked. We show that our method outperforms existing methods in large nonlinear utility spaces, and is computationally feasible for negotiations with as many as ten agents.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems; Coherence and coordination

General Terms

Algorithms

Keywords

Multi-issue Negotiation

1. INTRODUCTION

In this paper, we present a novel multi-round negotiation protocol that uses iterative narrowing of agent bids to find high-quality outcomes without making unrealistic demands concerning how much agents reveal about their utilities. In the early rounds, agents submit rough bids representing promising contract space regions (rather than individual contracts) to a mediator. In later rounds, they submit increasingly narrow bids for the subset of those regions that all agents liked in the previous round, until a deal is found. Our experimental results show that our method outperforms existing methods in large nonlinear utility spaces, and is computationally feasible for negotiations with as many as ten agents.

2. NONLINEAR NEGOTIATION

We consider the situation where N agents want to reach an agreement. There are M issues, $i_j \in I$, to be negotiated. Each issue corresponds to an orthogonal dimension in the contract space. An issue i_j has a value drawn from the domain of integers $[0, X]$, *i.e.*, $i_j \in [0, X]$. A contract is represented by a vector of issue values $\vec{s} = (i_1, ..., i_M)$.

An agent's utility function is described in terms of constraints. Each constraint, $c_k \in C$, represents a region with one or more dimensions, and has an associated utility value. A constraint c_k has value $w_i(c_k, \vec{s})$ if and only if it is satisfied by contract $\vec{s}$. An agent i's utility for a contract $\vec{s}$ is defined as follows:

$$u_i(\vec{s}) = \sum_{c_k \in C} w_i(c_k, \vec{s}) \quad (1)$$

This expression produces a "bumpy" nonlinear utility space, with high points where many constraints are satisfied, and lower regions where few or no constraints are satisfied. This represents a crucial departure from most previous efforts on multi-issue negotiation.

3. A BIDDING-BASED NONLINEAR NEGOTIATION PROTOCOL

Our protocol consists of two key steps:

Bidding: Agents create bids that each include a sub-region of their contract space, and submit them, along with a value for the bid, to the auctioneer.

Deal Identification: The mediator (auctioneer) uses a combinatorial process to identify possible deals, by finding all the overlaps *i.e.* all the contract regions that satisfy at least one bid from every agent. Note that there can be more than one such overlap. If every agent creates bids for every contract in its utility space, this approach is guaranteed to find optimal contracts in one round. The mediator can use exhaustive search over all bid combinations to find the social welfare maximizing negotiation outcome. Agents,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
AAMAS'07, May 14–18, 2007, Honolulu, Hawai'i, USA.
Copyright 2007 IFAAMAS.

978-81-904262-7-5 (RPS) ©2007 IFAAMAS

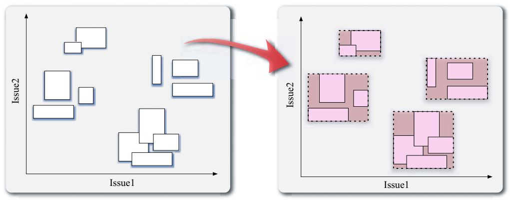

Figure 1: An Example of a Cluster Bid

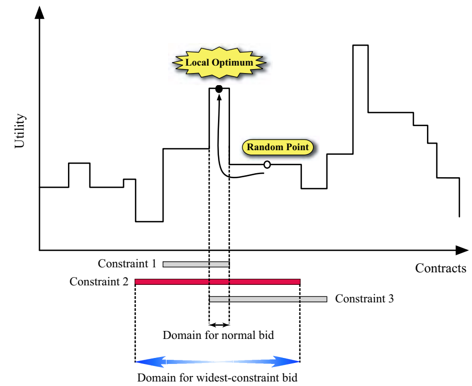

Figure 2: An Example of an Widest-Constraint Bid

however, may be reluctant or unable to provide their entire utility function to the mediator. In addition, we have to limit the number of bids the agents can submit so the deal identification process can complete in a reasonable amount of time. It is only practical to run our current deal identification algorithm if it explores no more than about 6,400,000 bid combinations per negotiation, which implies a limit of $\sqrt[N]{6400000}$ bids per agent for N agents. This limit becomes increasingly severe as the number of agents increases.

Agents of course will strive to create bids that cover the high utility regions in their contract space. If an agent has a large and complicated utility function with many such regions, it may be limited (if there are many agents) to creating bids for only a small subset of these regions. This raises the possibility that the deal identification algorithm will fail to find the optimal contract, or any contract at all, because the best (or only viable) contract may have come from a region that one or more agents did not bid on due to the bid limit. Our initial investigations bore this intuition out. We ran experiments where agents submitted bids for the $\sqrt[N]{6400000}$ highest peaks in their utility function. We found that the failure rate for the single-round bidding protocol becomes unacceptably high (80 percent or even higher) if there are more than 3 agents. This insight led us to develop a new class of bidding-based protocol that iteratively narrows the space of possible contracts over **several** rounds of bidding and deal identification. In each round, agents produce bids that cover high utility *subsections* of the contract space, cutting out regions with low utility. The overlaps returned by the deal identification step in one round become the boundaries within which agents search for bids in the next round. The contract space under consideration thus becomes smaller and smaller in each round, until a single winning contract is identified.

We describe the multi-round protocol we developed in the subsections below. Our current protocol consists of three rounds, each with a different bidding scheme, "cluster bidding", "widest-constraint bidding", and "peak bidding".

3.1 Cluster Bidding

To create a cluster-bid, each agent clusters the constraints in its utility space using the CLARANS [1] clustering algorithm. One cluster-bid is created for each constraint cluster. The domain of a cluster-bid is simply the union of the domains of all constraints included in the cluster. Figure 1 shows an example of creating cluster-bids for a two-issue utility function with binary constraints. The boxes with solid lines represent individual constraints, and the colored boxes with dotted lines represent the boundaries of a cluster bid. As we can see, while the cluster-bids include some "blank" regions (*i.e.*, where no constraints are satisfied), they also cover all the promising regions (*i.e.*, regions that satisfy at least one constraint) while cutting out "no interest" regions for that agents. The amount of the contract space eliminated by cluster bidding will be determined, of course, by how the constraints are distributed in the agents' utility function. If the constraints are naturally distributed into a few relatively tight clumps, cluster-bidding should be especially effective at narrowing the search space. We will return to this point in the experimental evaluation section below.

Agents invoke the clustering algorithm to find up to $\sqrt[N]{6400000}$ clusters in the agents' utility space, and then create bids for these clusters. Note that when there are many agents and the bid limit decreases, the domain of each cluster-bid becomes larger, since each cluster must include more constraints.

Cluster-bidding requires relatively little information revelation by the agents. Only relatively few bids are generated when there are substantial numbers of agents. The topology of the utility function within a bid is not revealed, which obscures which parts of the bidded region are valuable for the agent.

3.2 Widest-Constraint Bidding

Widest-constraint bidding is used by the agents to identify valuable regions from within the overlaps identified by applying deal identification to the cluster bids from round 1. Figure 2 gives an example of how this kind of bidding works. Agents pick points randomly from within each of the overlaps from round 1, and then use simulated annealing starting from these points to find locally optimal points (peaks) within those overlaps. Bids are then created whose domain is given by the largest volume constraint that is satisfied by each peak. The bid is thus guaranteed to include the local peak, as well as some of the surrounding high-utility regions. Unlike cluster bidding, widest-constraint bidding is also guaranteed to return bids that have no "blank" (zero-utility) regions.

Note that agents have a potentially less severe bid limit in the second (and third) round than in the first. The overlaps identified by the deal identification algorithm are guaranteed to be disjoint from each other, which means that, in round 2, the deal identification algorithm need only look for overlaps between bids *that come from the same round 1 overlap*. This greatly reduces the computational complexity of the deal identification process. Let's say there are N agents, and M overlaps were identified by deal identification in round 1. In order to stay within our limit of exploring only 6,400,000 combinations total in the deal identification step, each agent can submit $\sqrt[N]{6400000/M}$ bids *per overlap*. If there are 10 agents and 5 overlaps, for example, each agent can submit as many as $5 * \sqrt[10]{6400000/5} = 20$ bids, rather than the 5 bids allowed in round 1. This is important because it reduces the fraction of the agent's high-utility regions that may have to be ignored because of the bid limit.

3.3 Peak Bidding

In the final round (round 3), agents use "peak" bidding to create bids for the overlaps identified at the end of round 2. The agents pick points randomly from within each of these overlaps, and then use simulated annealing starting from these points to find peaks within the overlaps. The agents then generate, for each peak, a bid that includes the entire plateau of equi-utility contracts surrounding the peak. This is easy to do: the agent need merely find the intersection of all the constraints satisfied by that peak.

4. EXPERIMENTS

4.1 Setting

We conducted an experiment to evaluate the effectiveness and scalability of our approach. In each experiment, we ran 100 negotiations between agents with randomly generated utility functions. For each run, we applied a simulated annealing (SA) optimizer ([2]) to the sum of all the agents' utility functions to find the contract with the highest social welfare. This value was used to assess the economic efficiency (*i.e.*, how closely optimal social welfare was approached) for the negotiation protocols. We evaluated our iterative narrowing protocol. The parameters for our experiments were as follows:

- Agents: The number of agents N varies from 2 to 10.
- Contract Space: The number of issues is 10. The domain for issue values is $[0, 9]$.
- Utility Functions: The agent's utility functions each consist of 50 10-dimensional constraints. Each issue in a constraint has a width randomly selected from $[3, 7]$. issue1 = $[2, 6]$, issue2 = $[2, 9]$ would thus, for example, be a valid constraint. Each constraint has a utility value randomly selected from $[1, 100]$. Constraints are grouped into clusters, 10 clusters per utility function, 5 constraints per cluster. Clusters have maximize sizes that vary with the experimental condition. We manipulated the degree of correlation among the agents' utility functions, by varying the number of ("*shared*") clusters whose location is the same across all agents. All other clusters were randomly distributed throughout the contract space.

4.2 Results

Our experiments revealed that the iterative narrowing protocol can be effective at producing near optimal negotiation outcomes for up to 10 agents. Figure 3 shows the optimality values for the case where clusters have a width selected randomly from $[3, 7]$. As we can see, we get contracts averaging over 90% of optimal when there are four or fewer shared clusters. When there are more than four shared clusters, *i.e.*, when the utility functions are highly correlated across the agents, it reduces the optimality values because high correlation across agents leads the deal identification algorithm to find many overlaps between the agents. Due to computational constraints, only a subset of overlaps can often be passed on to the agents as the starting point for the next round of bidding, which

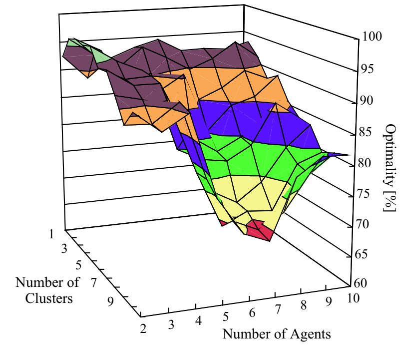

Figure 3: Utility Values for the Small Cluster Case

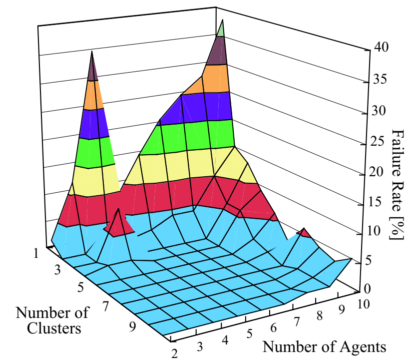

Figure 4: Failure Rates for the Small Cluster Case

means that some overlap areas will have no bids made for them, increasing the chance that good deals will be missed.

The failure rates for the small cluster case are shown in Figure 4. The failure rates become significant when the agent functions have a very low correlation with each other (*i.e.*, when they have just one or two shared clusters, so there are relatively few contracts that will make all the agents happy) but otherwise the failure rates are in the single digits.

5. CONCLUSION

In this paper, we have proposed a novel auction-based protocol designed for the important challenge of negotiation with multiple interdependent issues. Our experimental results show that our method offers near-optimal negotiation outcomes for up to 10 agents, if they have substantially clustered utility functions. We are aware of no previous work that scales up nonlinear negotiation beyond two agents.

6. REFERENCES

[1] R. T. Ng and J. Han. Clarans: A method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, 14(5):1003–1016, 2002.

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