
A Framework for Contracting in a Multi-Agent Environment

Maria Gini

Department of Computer Science and Engineering
University of Minnesota

Work done with John Collins, Corey Bilot, Paula Greeve, Len Josephs, Dave Kohn,
and Bamshad Mobasher.

Funded in part by the National Science Foundation under award NSF/IIS-0084202.

Outline of Talk

- A. B2B and Agent-based Contracting.
- B. The MAGNET system.
- C. Theoretical Issues.
- D. Conclusions and Future Directions.

How large is the market for B2B?

The business-to-business (B2B) e-commerce market is expected to expand rapidly, with the global market expected to exceed \$7.29 trillion in 2004, according to Gartner Group research.

Business-to-business hubs, which link buyers within a particular industry or across a shared need, are expected to handle as much as \$1.25 trillion by 2003.

A recent separate study from Boston Consulting Group predicts productivity gains from B2B e-commerce will equal 1% – 2% of sales by 2004 and 6% by 2010.

A report from Jupiter Media Metrix forecasts B2B e-marketplace spending to grow from \$2.6 billion to \$137.2 billion between 2000 and 2005.

Problems with current E-commerce systems

Current e-commerce systems typically:

- rely on either fixed-price catalogs or auctions. Companies usually work with prequalified suppliers and buyer-supplier relationships depend on factors such as quality, delivery performance, and flexibility as opposed to just cost.
- do not have any notion of time (except for domain specific systems such as SABRE used in the travel industry). Time plays a fundamental role in supply-chain formation and management, since many products are made up of different parts and require multiple suppliers who have to coordinate their work.

Why do we need a Market Infrastructure? (I)

The market infrastructure can provide support for multi-agent negotiation over extended time periods. The market tracks the requests, commitments, and progress towards task completion among the agents. In particular, it can:

- provide verification and persistence of identity;
- maintain the state of transactions over time;
- keep a common time reference;
- notify agents of negotiated deadlines;
- assess negotiated penalties.

Why do we need a Market Infrastructure? (II)

The market infrastructure can provide value-added services, such as:

- a Registry of market participants who have expressed interest in doing business in the market. Entries in this registry would include the identity of a participant, a catalog of that participant's interests, products or capabilities;
- authentication services, to protect participating agents against fraud and misrepresentation;
- credit and payment services;
- a domain Ontology, specifying the terms of discourse within that domain. In a commodity-oriented domain, the ontology would include terms for the products or services within the domain, terminology for quality, quantity, features, terms and conditions of business, etc.;
- matchmaking facilities to bring together suppliers and customers,
- a repository of statistical data about participants. This may include general statistics about availability of suppliers with specific capabilities, or independent ratings for both suppliers and customers based on past performance;
- publish-subscribe facilities to provide registered participants with notification of important events.

Why is making decisions about contracts difficult?

When agreeing to a contract, an agent takes multiple risks. One or more suppliers might back out of the agreement, or fail to deliver on time, with devastating cascade effects.

What happens if a part is not delivered on time, and the next step in the processing cannot be done? what if the agent does not succeed in renegotiating its contracts? should it decide to default on them and pay a penalty, or should it pay a premium for changing the terms of its current contracts?

Human decision makers are faced with these kind of choices all the times. Can the decision process be automated? How?

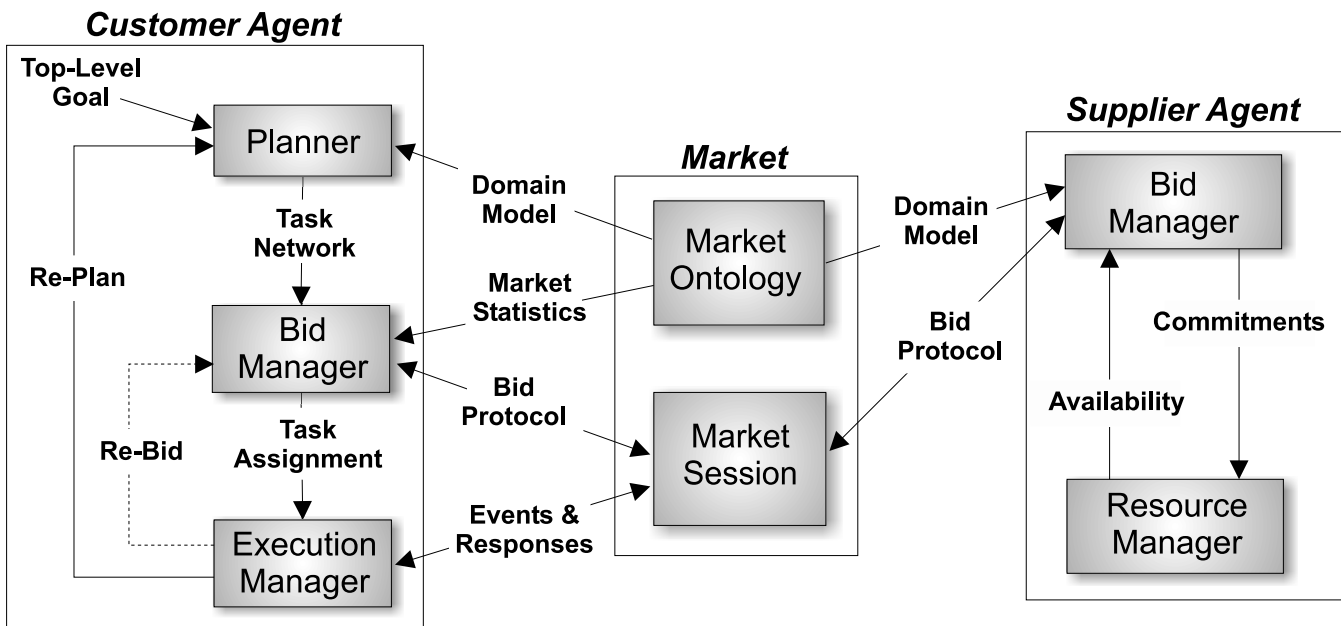
B. MAGNET

The University of Minnesota's MAGNET (Multi AGent NEgotiation Testbed) system is a testbed for multi-agent contract negotiation for tasks that have temporal and precedence constraints.

MAGNET adds a new dimension to business-to-business (B2B) interactions, by adding the ability to automate the negotiation and execution of complex combinations of contracts among multiple suppliers. This is especially important for the coordination of supply-chain management with production scheduling.

Agent interactions in MAGNET are mediated through an independent market infrastructure which, among other services, provides a domain ontology, a contracting protocol, authentication services, and tracks the requests, commitments, and progress towards task completion among the agents.

The MAGNET Architecture



MAGNET Agent Roles

A MAGNET agent has three basic functions:

- (i) planning
- (ii) negotiation
- (iii) execution monitoring

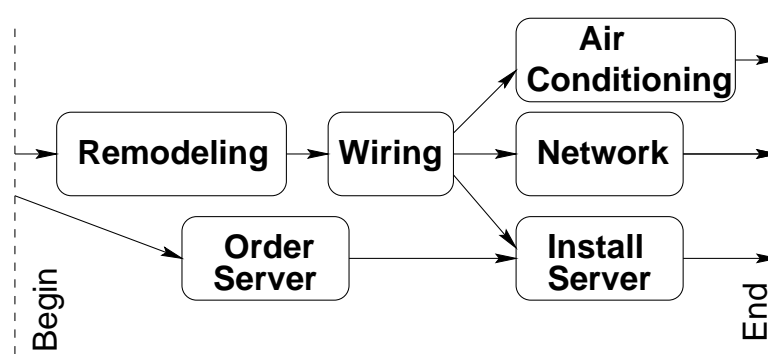
We distinguish between two agent *roles*, the *Customer* and the *Supplier*.

A Customer is an agent who has a task to accomplish, and needs resources outside its direct control in order to accomplish it. The task may have a *value* that varies over time.

A Supplier is an agent who has resources and who, in response to a *Request for Quotes* (RFQ), may offer to provide resources or services, for specified prices, over specified time periods.

An Example

Let's imagine we need to do a site preparation for installing a large-scale server.



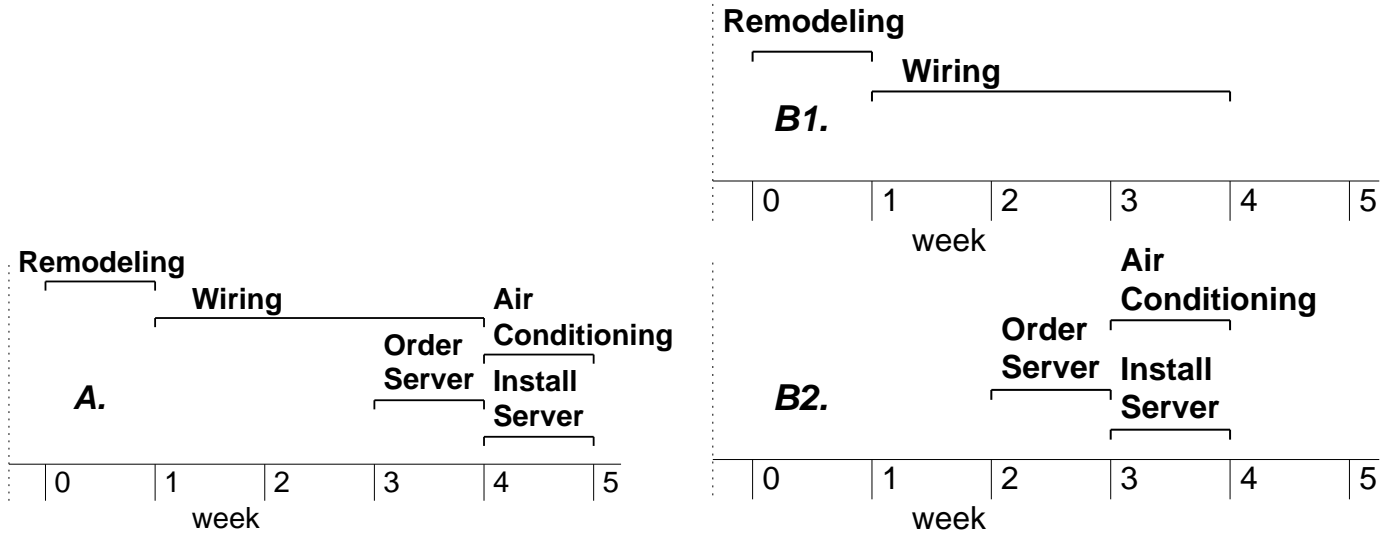
Our plan is complicated by a couple of factors. The server must be ordered ahead, and will arrive one week after we order it. Because it is the most expensive part of the installation, we do not want to leave it idle, and so it must be installed immediately when it arrives. Also, there is a network boom in our area, and people to do wiring are hard to find. We may have to wait.

The Bid Manager (I)

The Bid Manager is responsible for ensuring that resources are assigned to each of the tasks, that the assignments taken together form a feasible schedule, and that the cost and risk of executing the plan is minimized.

- It first assembles a schedule for the bidding process, and adds items to the agenda. For example, we might not want to take bids on the air conditioning for our site-preparation task until we have firm dates for the wiring.
- It then determines the scheduling constraints for the tasks and builds an RFQ. The RFQ contains tasks, precedence relations, and time windows.

RFQs for the Example



The figure shows two alternative ways to schedule and compose the RFQs for our site preparation project.

In version A, we believe we have 5 weeks to finish our site, and the only scarce resource is wiring. Therefore, we allow 3 weeks for the one-week wiring job, and we are guaranteed that if we receive bids on all tasks, they can be combined feasibly. In version B, we are interested in finishing the site as soon as possible. Therefore, we bid out the remodeling and wiring first in RFQ B1, and we bid out the remainder of the tasks in RFQ B2 after we get a bid that finishes the wiring by the end of week 3.

The Bid Manager (II)

Once bids are received, the task of the Bid Manager is to find an optimal or near-optimal mapping of bids to tasks, respecting the temporal constraints.

We have implemented two bid evaluators:

- One is based on Integer Programming. The Integer Programming (IP) solver operates in two phases. The first phase generates basic bid-compatibility constraints, and then walks all paths of length 2 or greater in the precedence network, across all compatible bid combinations, to discover feasibility constraints. These are then packaged up and sent off to an external IP solver.

IP guarantees an optimal solution, but there is a significant variation in the amount of time taken to terminate.

- The other is a highly modular Simulated Annealing (SA) search engine. Simulated-Annealing is a stochastic search algorithm. Starting with a task network and a set of bids, the algorithm generates and evaluates mappings of bids to tasks until one of several stopping conditions holds. These include failure to find improvement for some number of iterations, expiration of the deliberation time limit, and lack of mappings that have any untried expansions.

SA does not guarantee an optimal solution, but it is an anytime algorithm. The non-optimal solutions it reports are typically within a few percent of optimal.

Experimental results using Integer Programming

Experiments with dual-processor 850 MHz Linux boxes, timings in wall-clock time. Each problem set consists of 200 problems, with randomly-generated plans and randomly-generated bids.

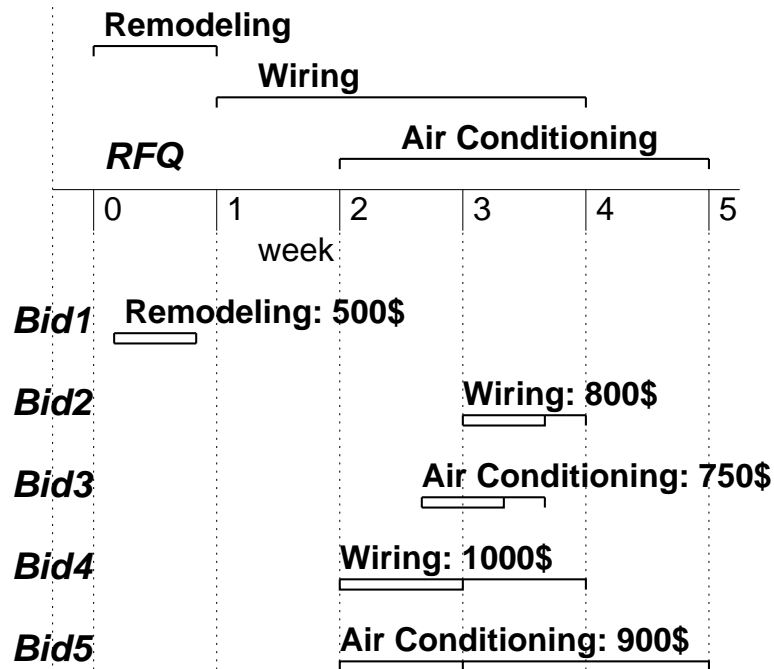
Task Count	Bid Count	Bid Size	Solved	Rows	PreProc Time (msec)	IP Time (msec)	$\sigma(\text{time})$
5	13.5	2.14	188	19.1	10.4	34.5	15
10	28.5	3.20	184	44.3	70.1	34.3	57
15	45.4	4.35	181	85.6	191	30.2	170
20	60.8	5.21	176	145	489	40.7	380
25	77.6	6.30	165	281	1268	56.2	1423
30	93.3	7.41	157	514	3375	194	4353
35	110.6	8.69	137	861	5181	317	7100

Experimental results using Simulated Annealing

Experiments with dual-processor 850 MHz Linux boxes, timings in wall-clock time. Each problem set consists of 200 problems, with randomly-generated plans and randomly-generated bids.

Task Count	Bid Count	Bid Size	Solved	Optimum	Time (msec)	$\sigma(\text{time})$
5	13.5	2.14	188	188	10	10
10	28.5	3.20	184	178	147	391
15	45.4	4.35	179	141	524	986
20	60.8	5.21	169	118	1171	1707
25	77.6	6.30	147	89	1958	2683
30	93.3	7.41	141	80	3445	4012
35	110.6	8.69	116	66	4012	4289

Bids for the example



- Bid 2 indicates wiring could start at the beginning of week 3, would take 3 days, and the supplier was willing to shift that out 2 more days to accommodate our schedule.
- Bid 3 shows that air conditioning could start through week 2, take 3 days, and finish through week 3. Clearly these two bids cannot be combined.
- Bid 4 has is more expensive for wiring, could start earlier, but needs a week to finish. This can be combined with Bid 3, but with no slack for contingencies.
- Bid 5 gives us a large enough time window for the air conditioning task to be combined with either Bid 2 or Bid 4.

The best combination is Bid1, Bid2, Bid5.

Theoretical Issues

There are a number of more theoretical issues in the design of MAGNET.

We will examine briefly two of them:

1. Modeling the decision processes: how can we model the uncertainties in the decision process and have MAGNET agents making optimal decisions?
2. Auction theories: what algorithm is needed for optimal bid selection?

Modeling the decision processes (I)

To model uncertainties in the decision process, we use Expected Utility Theory. Expected Utility Theory models decision-making under uncertainty by using probabilities and a utility curve, $U(W)$, a function that maps a level of “wealth” to a level of utility.

A decision-maker who is faced with an opportunity consisting of a set of n wealth-based outcomes will calculate the expected utility over the set of outcomes:

$$E(U) = \sum_{i=1}^n U(W_i)p_i$$

where p_i is the probability of outcome i , and W_i is the resulting wealth of the decision-maker if outcome i is realized.

To make a decision, the decision-maker compares this expected utility, $E(U)$, to her current utility, $U(W_0)$, where W_0 represents her current wealth. If the opportunity’s expected utility, $E(U)$, exceeds her current utility, $U(W_0)$, she will pursue the opportunity. Similarly, a decision-maker faced with multiple opportunities can decide which (if any) she will pursue by comparing the expected utilities of the opportunities and her current level of utility.

Modeling the decision processes (II)

We use Expected Utility Theory to guide the agent:

- in preparing the RFQ. Our experimental studies show that there is a correlation between the time windows specified in the RFQ, the number of bids submitted, and the likelihood of composing them in a feasible way.
- in making decisions when there is a trade-off between the overall cost of a plan and the likelihood of the plan succeeding. For example, the agent may need to choose between suppliers, some of whom charge a higher price but are more likely to complete the task successfully, while others may be less likely to complete the task on time but may charge less.

Auctions

Auctions are a widely used way of allocating goods. Their popularity has increased with the availability of auctions on the Internet. Most Internet marketplaces have very simple auctions, in which several bidders bid for an item, and the auctioneer select the highest bid.

Combinatorial auctions are auctions in which bidders are allowed to bid on collections of goods, instead of individual goods. The collections (or bundles) are chosen by the bidders. This is good for the bidders, but hard for the seller, since determining the winners of combinatorial auctions is computationally hard.

Our IP and SA formulations solve the winner determination problem for combinatorial auctions. We handle two additional complications:

1. the bid selection algorithm must insure the temporal feasibility of the bids accepted.
2. the bid selection algorithm must insure complete coverage of the tasks.

D. Conclusions and Future Work

Challenges

Develop more sophisticated negotiation strategies (risk assessment, coalition formation, etc).

Combine Expected Utility Theory and automated contracting with interaction with a human decision maker (mixed-initiative system).

Elicite the utility curve from the human decision maker.

Implementation

Our current implementation of the Customer Agent and of the MAGNET infrastructure will be made available to the research community at the end of Summer. Let me know if you are interested in trying it.

More information on software agents

- Jeffrey Rosenschein and Gilad Zlotkin, “Designing Conventions for Automated Negotiation,” *AI Magazine*, Fall 1994, 29-46.
- Nicholas R. Jennings and Michael J. Wooldridge, “Applications of Intelligent Agents,” in Nicholas R. Jennings and Michael J. Wooldridge (Ed.), *Agent Technology Foundations, Applications, and Markets*, Springer-Verlag, 1998.
- Jeffrey Bradshaw (ed.), *Software Agents*, AAAI Press/The MIT Press, 1997.

More information specific to our project

- J. Collins, C. Bilot, M. Gini, and B. Mobasher, “Decision Processes in Agent-Based Automated Contracting,” *IEEE Internet Computing*, pp 61-72, March/April 2001.
- Tapan Biswas. *Decision-Making Under Uncertainty*. St. Martin’s Press, Inc., 1997.
- Noam Nisan. “Bidding and allocation in combinatorial auctions,” In *Proc. of ACM Conf on Electronic Commerce (EC’00)*, pages 1–12, Minneapolis, Minnesota, October 2000. ACM SIGecom, ACM Press.
- Colin R. Reeves. *Modern Heuristic Techniques for Combinatorial Problems*. John Wiley & Sons, New York, NY, 1993.

Information on MAGNET and related papers can be found on the Web at

<http://www.cs.umn.edu/Research/airvl/magnet/>