A Methodology and Equilibria for Design Tradeoffs of Autonomous Trading Agents

Ioannis A. Vetsikas
Computer Science Dept., Cornell University
Ithaca, NY 14853, USA
vetsikas@cs.cornell.edu

Bart Selman
Computer Science Dept., Cornell University
Ithaca, NY 14853, USA
selman@cs.cornell.edu

ABSTRACT

In previous work we have introduced a principled methodology for systematically exploring the space of bidding strategies when agents participate in a significant number of simultaneous auctions, and thus finding an analytical solution is not possible. We decompose the problem into sub-problems and then use rigorous experimentation to determine the best partial strategies. In this paper we clarify and extend our methodology. We also examine the Bayes-Nash equilibria for some of the design tradeoffs that arise when the problem is decomposed and each tradeoff is examined independently and provide some equilibrium strategies. Our agent WhiteBear was created by using the results of this methodology and has consistently been a top-scoring agent in all the competitions.

1. INTRODUCTION

Auctions are becoming an increasingly popular method for transacting business either between individuals over the Internet (e.g. eBay) or even between businesses and their suppliers. While optimal strategies for bidding in an auction for a single item are known, in practice agents (or humans) are rarely interested in a single item. They wish to bid in several auctions in parallel for multiple substitutable or complementary goods. The fact that the value of each good to an agent depends on other goods is what makes this problem particularly hard. There had been relatively few studies about agents bidding in multiple simultaneous auctions and they mostly involve bidding for substitutable goods (e.g. [1], [3], [4]). However recently the introduction of the Trading Agent Competition (see [8]) has given momentum to the research in this field by providing researchers with a challenging benchmark domain which incorporates several elements found in real marketplaces. One may find the rules of TAC on the web at $http:\backslash\backslash www.sics.se\backslash tac$.

In this short version of the paper, we present our methodology for determining the strategy an agent should use when bidding in simultaneous auctions (see section 2). We also find Bayes-Nash equilibria that a rational agent would consider when it participates in an auction that can close at one or more predetermined times (see section 3). For more details refer to the extended version of this paper [6]; in that version we also present a "complete" set of experiments, based on our methodology, for determining an overall best strategy in the Trading Agent Competition and further improvements to our agent and methodology (e.g. reducing the number of games needed to get significant results in our experiments).

AAMAS '04 New York, NY USA
2004 ACM X-XXXXX-XX-X/XX/XX ....

2. METHODOLOGY AND AGENT ARCHITECTURE

In this section we extend and clarify the methodology that we presented in [7], in order to systematically explore the space of bidding strategies when it is not possible to find an equilibrium solution, because the agent participates in a significant number of simultaneous auctions. We decompose the problem into sub-problems and then use rigorous experimentation to determine the strategies for each subproblem. This is contrary to the direction that most other researchers have explored; because of the obvious dependencies among the bids most teams participating in TAC did not wish to examine each auction independently.

2.1 Problem Decomposition

According to our methodology we must decompose the problem as much as possible. This involves two phases:

1. The quantities to be bought for each commodity are determined independently of the price and the placement time of the bid and are computed by maximizing the utility of the agent assuming that all the goods are bought at some predicted prices and that every unit will be bought instantly. A dedicated module called "the planner" (or optimizer) is doing this task. For more information on this module see [7].
2. Once the quantity that the agent should buy has been computed we need to find the price and the placement time of each bid. We consider separately each different auction type, which corresponds to a different commodity type, and we compute the space of possible "partial strategies"[1] for this auction type. Initially we compute the extent of this space by setting boundary partial strategies, that is strategies that place bids at the lowest or highest prices that a rational agent would consider possible for this setting and at the earliest or latest time possible. This defines the boundaries of the strategy space for each auction type. Having accomplished this, it is necessary to generate intermediate strategies[2] to use in our experiments, since we cannot possibly consider all possible strategies in the strategy space. This is done systematically by
   - computing equilibrium strategies for each auction,
   - combining the boundary strategies, and
   - modifying the boundary strategies.

[1] We called these strategies partial because they only deal with one particular type of auction. These are usually computed analytically or using domain knowledge.

[2] If the boundary strategies will place bids at price p_{low} and p_{high} in a certain case, then the intermediate strategy should place its bid at price $p : p_{low} \leq p \leq p_{high}$.

2.2 Experiment Setup

The number of possible combination of partial strategies coupled with the fact that each strategy can behave differently based on the mixture of agents that participate in the game leads to a prohibitive number of experiments that need to be performed in order to get the "overall best strategy". We propose one possible way of exploring this space. We can search it efficiently by exploring the result that changing only one of the partial strategies has on the behavior of the multi-agent system. To do this in a systematic way we use several sets of experiments, each of which is designed to evaluate one particular partial strategy in different mixtures of agents.[3] In order to explore the whole spectrum of possible mixtures we propose to keep a fixed number of agents who are using the intermediate strategies, while systematically changing the mixture of agents using the boundary cases. This will explore sufficiently the different multi-agent environments that the agents can participate in, since the behavior caused by the intermediate strategies is within the bounds of the behavior caused by the boundary strategies. Using statistical tests we can evaluate the performance of the agent and determine whether some agent performs significantly better than the others. Therefore each experiment set explores the strategy space in one dimension and we need to run experiments varying different partial strategies each time in order to explore the whole strategy space. We can accomplish this by using a different partial strategy to explore each time and by setting the strategies that are not varied equal to the best strategy (or strategies found so far). In case our experiments show that the fixed strategy we used for one particular subproblem was not the best one, then we vary this partial strategy in the next experiment. In addition, if we want to explore a feature that would improve the performance of the optimizer, we set half the agents to use this feature and half not to, while all agents use the best partial strategy combinations that we have found thus far. This way we can reach a strategy combination (or combinations) that perform consistently well against varying mixtures of agents. An extra benefit of this methodology is that it allows us to derive general observations about the behavior of certain strategies in different domains.

To accommodate our methodology we used an *adaptive*, *flexible* and *easily modifiable* architecture that follows the "Sense Model Plan Act (SMPA)" architecture. Other trading agents (e.g., [2], [5]) have used a similar global design. The overall architecture can be summarized as follows:

while (not end of game) {
1. Get price quotes and transaction information
2. Calculate price estimates
3. *Planner*: Form and solve optimization problem
4. *Bidder*: Bid to implement plan
   Determine each bid independently of all other bids
   Use a different "partial strategy" for each different bid }

3. BAYES-NASH EQUILIBRIA

In this section we compute Bayes-Nash equilibria for the tradeoff that arises when we consider the purchase of hotel rooms in TAC. Assume that N risk-neutral agents wish to buy 1 unit of a certain good each. The agents have valuations (utilities) u_i which are i.i.d. with probability function $F(u)$. To this effect each agent i submits a bid v_i. An independent seller sells m units of the desired good in an $m-th$ price auction, i.e. the goods are sold to the agents which submitted the m highest bids at a price equal to the $m-th$ highest bid's price for all winners. The auction either closes immediately after one round of bids with probability p or with probability $(1-p)$ the agents are allowed to submit one more round of bids. Let Q be the starting price at the auction, i.e. all bids submitted must exceed this value. We compute a Bayes-Nash equilibrium strategy $g(u)$ that maps utilities u_i to bids v_i.

When $m = 1$ good is sold, the starting price is $Q \geq 0$ and the bidding lasts for exactly one round ($p = 1$) the equilibrium strategy is

$$g(u_i) = u_i - \frac{\int_Q^{u_i} \big(F(\omega)\big)^{N-1} \cdot d\omega}{\big(F(u_i)\big)^{N-1}} \quad (1)$$

When $m = 1$ good is sold, the starting price is $Q = 0$ and a second round of bidding exists with probability $(1-p)$ ($p \neq 0, 1$) the equilibrium strategy is the solution of the differential equation

$$\big(u_i - g(u_i)\big) \cdot \frac{\Phi'(u_i)}{g'(u_i)} = \Phi(u_i) \cdot \Psi(g(u_i)) \quad (2)$$

where $\Phi(x) = \big(F(x)\big)^{N-1}$ and $\Psi(x) = 1 + \frac{1-p}{p} \cdot \big(F(x)\big)^{N-1}$, and the boundary condition is $g(0) = 0$. There is no known closed form solution to this equation, but a numerical solution can be easily calculated.

When $m > 1$ goods are sold, the starting price is $Q \geq 0$ and the bidding lasts for exactly one round ($p = 1$) the equilibrium strategy is

$$g(u) = u - \frac{e^{\int_Q^u \frac{-Y'(\omega)}{\Phi(\omega)-Y(\omega)} \cdot d\omega}}{\Phi(u) - Y(u)} \cdot \int_Q^u \frac{\Phi(z) - Y(z)}{e^{\int_Q^z \frac{-Y'(\omega)}{\Phi(\omega)-Y(\omega)} \cdot d\omega}} \cdot dz \quad (3)$$

where $\Phi(x) = \sum_{i=0}^{m-1} C(N-1, i) \cdot \big(F(x)\big)^{N-1-i} \cdot \big(1 - F(x)\big)^i$ and $Y(x) = \sum_{i=0}^{m-2} C(N-1, i) \cdot \big(F(x)\big)^{N-1-i} \cdot \big(1 - F(x)\big)^i$. Note that $C(K, i)$ are the number of possible combinations when choosing i items from a set of K items.

When $m > 1$ goods are sold, the starting price is $Q = 0$ and a second round of bidding exists with probability $(1-p)$ ($p \neq 0, 1$) the equilibrium strategy is the solution of the differential equation

$$\big(u_i - g(u_i)\big) \cdot \frac{\Phi'(u_i)}{g'(u_i)} = \big(\Phi(u_i) - Y(u_i)\big) \cdot \Psi(u_i, g(u_i)) \quad (4)$$

where $\Phi(x) = \sum_{i=0}^{m-1} C(N-1, i) \cdot \big(F(x)\big)^{N-1-i} \cdot \big(1 - F(x)\big)^i$, $Y(x) = \sum_{i=0}^{m-2} C(N-1, i) \cdot \big(F(x)\big)^{N-1-i} \cdot \big(1 - F(x)\big)^i$ and $\Psi(u_i, Q) = 1 + \frac{1-p}{p} \cdot \frac{\Phi(Q)-Y(Q)}{\Phi(u_i)-Y(u_i)} \cdot e^{\int_Q^{u_i} \frac{-Y'(\omega)}{\Phi(\omega)-Y(\omega)} \cdot d\omega} \cdot \big(\Phi(u_i) + \int_Q^{u_i} \frac{Y(\omega) \cdot \Phi'(\omega)}{\Phi(\omega)-Y(\omega)} \cdot d\omega\big)$, and the boundary condition is $g(0) = 0$. There is no known closed form solution to this equation, but a solution does exist and can be computed numerically.

4. REFERENCES

[1] P. Anthony, W. Hall, V. Dang, and N. R. Jennings. Autonomous agents for participating in multiple on-line auctions. In *Proc IJCAI Workshop on E-Business and Intelligent Web*, pages 54–64, 2001.

[2] A. R. Greenwald and J. Boyan. Bid determination for simultaneous auctions. In *Proc of the 3rd ACM Conference on Electronic Commerce (EC-01)*, pages 115–124 and 210–212, Oct. 2001.

[3] T. Ito, N. Fukuta, T. Shintani, and K. Sycara. Biddingbot: a multiagent support system for cooperative bidding in multiple auctions. In *Proceedings of the Fourth International Conference on MultiAgent Systems*, pages 399 – 400, July 2000.

[4] C. Preist, C. Bartolini, and I. Phillips. Algorithm design for agents which participate in multiple simultaneous auctions. *In Agent Mediated E-Commerce III (LNAI), Springer-Verlag, Berlin*, 2001.

[5] P. Stone, R. Schapire, M. Littman, J. Csirik, and D. McAllester. Attac-2001: A learning, autonomous bidding agent. In *Agent Mediated E-Commerce IV. LNCS, volume 2531. Springer Verlag, Berlin.*, 2002.

[6] I. A. Vetsikas and B. Selman. A methodology and equilibria for design tradeoffs of autonomous trading agents (forthcoming).

[7] I. A. Vetsikas and B. Selman. A principled study of the design tradeoffs for autonomous trading agents. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 473–480, 2003.

[8] M. P. Wellman, P. R. Wurman, K. O'Malley, R. Bangera, S. de Lin, D. Reeves, and W. E. Walsh. Designing the market game for TAC. *IEEE Internet Computing, April*, March/April 2001.

[3] This is important since the performance of each strategy depends on the strategies used by the other agents. In previous work, we did not present a clear methodology for what sets of experiments to run, but we clarify this process in this paper.