Evaluating Bidding Strategies for Simultaneous Auctions

Teddy Candale
University of Tulsa
teddy-candale@utulsa.edu

Sandip Sen
University of Tulsa
sandip@utulsa.edu

ABSTRACT

Bidding for multiple items or bundles on online auctions raises challenging problems. We assume that an agent has a valuation function that returns its valuation for an arbitrary bundle. In the real world all or most of the items of interest to an agent is not present in a single combinatorial auction. We study the problem of bidding for multiple items in a set of simultaneous auctions, each of which sell only a single unit of a particular item. Hence an agent has to bid in multiple auctions to obtain preferred item bundles. While an optimal bidding strategy is known when bidding in sequential auctions, only suboptimal strategies are available when bidding for items sold in auctions running simultaneously. To decide on an agent's bid for simultaneous auctions, we investigate a multi-dimensional bid improvement strategy, which is optimal given an infinite number of restarts, . We provide a comparison of this algorithm with existing ones, both in terms of utilities generated and computation time, along with a discussion of the strengths and weaknesses of these strategies.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Coherence and coordination, Multiagent systems, Intelligent agents*

General Terms

Algorithm, Performance, Experimentation

Keywords

bidding, simultaneous auctions

1. INTRODUCTION

Auction theory has received significant attention from researchers following the development of electronic auctions on the Internet. Researchers are interested both in designing auctions with desirable properties [5, 6] and designing automated agents to represent interests of human users [3, 4, 7].

In a multi-auction setting, multiple single-item auctions are run concurrently or sequentially. A potential bidder needs to estimate closing prices of such auctions to compute optimal bids. The bidder incurs the computational cost of estimating closing prices and calculating optimal bids given its valuation for items and the expected closing prices. The problem of computing optimal bids is complex. While an optimal bidding strategy is known for sequential auctions [3, 8], none is known for simultaneous auctions.

We have developed a multidimensional bid improvement strategy for bidding in simultaneous auctions. We use an optimization techniques to determine bids that maximize the expected utility function. The focus of this paper is a comparative evaluation of our proposed algorithm with existing ones in terms of time complexity and the quality of the solutions generated.

2. RELATED WORK

Bidding for bundles in simultaneous auctions has received a lot of attention partly due to the Trading Agent Competition (TAC)[1]. Stone *et al.* and Greenwald and Boyan have studied the possibility to bid in multi-auctions in the context of the TAC [4, 2, 8, 7], which resulted in the design of two top-scoring agents. Their approaches, however, were fundamentally different from the one we have developed [1]. Our multi-dimensional bid improvement algorithm is motivated by optimization techniques, where, given any bid vector, we generate the next bid vector by sequentially optimizing the bid for each item while assuming the bid for the other items are held constant. The approaches used by both Stone *et al.* and Greenwald involve assigning valuations to individual items based on expected marginal utilities of the items. The marginal utility of an item i is the extra-profit generated by the acquisition of i at price equal to zero. The idea behind the marginal utility is that one is willing to pay for an item up to the benefit of getting it. We provide a more detailed description of Stone *et al*'s and Greenwald and Boyan's algorithms in Section 4 and offer a comparison of the effectiveness of their algorithms and ours in Section 5.

3. SIMULTANEOUS AUCTION MODEL

We consider a bidder which plans to acquire items from the set $\mathcal{I} = \{1, \ldots, N\}$. A valuation function ϑ represents the bidder's preference by assigning a value the bidder is

[1]http://www.sics.se/tac/.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.

willing to pay for each bundle. Item i is available in only one single-item single-unit auction a_i. An auction is modeled by the cumulative probability distribution F_i of the closing prices in the range $[\underline{p_i}, \overline{p_i}]$. We assume these distributions to be continuous, independent, and known to the bidder. When a_i closes, the bidder gets the item if it has placed a bid b_i greater than or equal to the closing price p_i ($b_i \geq p_i$) and the payment is equal to the closing price. All auctions are run in parallel and their closing times are not known by the bidder. $B = (b_1, \ldots, b_N)$ represents the bids placed simultaneously in all auctions by the bidder and $P = (p_1, \ldots, p_N)$ are the closing prices. We assume the bidder to be rational, i.e., it wants to maximize its expected utility from the purchases, if any. Hence it tries to find a bid vector B^* such that $B^* = \underset{B \in \mathcal{B}}{\operatorname{argmax}}\ \bar{\alpha}(B)$ where $\mathcal{B}$ is the bid space and $\bar{\alpha}(B)$ is the expected utility of bidding B.

4. BIDDING STRATEGIES

We review some prominent bundle bidding strategies before discussing our approach.

4.1 Marginal utility bidding ($\overline{MU}$)

We first present the optimal bidding strategy for bundles in sequential auctions[2, 8]. In the remainder of this paper, we will refer to this method as *expected marginal utility bidding* or $\overline{MU}$. When bidding for the i^{th} item, the bidder places the bid $b_i = \bar{\mu}(i, I_h, I_r)$ in auction a_i. I_h contains items held by the bidder, I_r contains items to be auctioned, $\bar{\mu}(i, I_h, I_r)$ is the expected marginal utility of item i which can be viewed has the extra-profit due to the acquisition of i at zero-cost.

$\overline{MU}$ is optimal when bidding in sequential auctions as shown by Greenwald and Boyan in [2]. $\overline{MU}$ is suboptimal for simultaneous auctions. In particular, when items are substitutable, the bidder may acquire two items it may not desire to acquire together. Another downside of $\overline{MU}$ is its complexity. The calculation of the marginal utility is exponential since it requires the knowledge of the profit generated by each possible bundle.

To adopt $\overline{MU}$ for simultaneous auctions, a bidder has to compute all bids simultaneously. In that case, $\mathcal{I}_h = \emptyset$. The bidder bids for each item as if it is the first item to be auctioned. In other words, the bidder places $b_i = \bar{\mu}(i, \emptyset, \mathcal{I} \setminus \{i\})$ in auction a_i.

4.2 Expected Value Marginal Utility bidding (EVMU)

In this section, we present a variant of $\overline{MU}$ introduced by Greenwald and Boyan [2]. This variant tries to overcome the issue raised by leaving out of consideration some bidder preferences. To prevent the bidder from obtaining two "similar" items, a set of preferred items is precomputed. We refer to this set as the *acquisition set* $\mathcal{I}_{evmu}$. The acquisition set is the optimal set of items to be obtained assuming that items are auctioned at deterministic prices equal to the expected closing prices for these items. The bidder places a bid for all items in $\mathcal{I}_{evmu}$ and the bids are equal to the expected marginal utility of the items. Like $\overline{MU}$, the computation of marginal utility in EVMU is exponential.

4.3 Multi-Dimensional Bid Improvement (MDBI)

Now we present an approach to bidding in simultaneous auctions based on an incremental optimization technique [1]. Assume that the bidder has decided, by some means, to bid the vector B. We try to improve B replacing b_i by $\beta_i(B_{-i})$ where $\beta_i(B_{-i})$ is the best bid for item i holding other bids constant ($\beta_i(B_{-i}) = \underset{b_i \in [\underline{p_i}, \overline{p_i}]}{\operatorname{argmax}}\ \bar{\alpha}(b_i \vee B_{-i})$). We exclude the proof due to space limitations but observe that $\beta_i(B_{-i})$ is such that $\frac{\partial \bar{\alpha}}{\partial b_i}(\beta_i(B_{-i}) \vee B_{-i}) = 0$, i.e., $\beta_i(B_{-i})$ is the *marginal expected welfare* of bidding B_{-i}.

To use this approach the bidder has to choose an initial bid vector B_0 by some means, e.g., randomly. Let B_t be the bid after t iterations. B_{t+1} is obtained from B_t by sequentially improving bids for each of the N items in turn. While considering the improvement of the bid for an item, we keep the bids for other items constant in B_t. We call this $N-$*step* perturbation an $N-$*sequential* improvement and the algorithmic process a multi-dimensional bid improvement (MDBI) scheme. The process is stopped when $||B_{t+1} - B_t|| < \varepsilon$ where ε is a positive constant defined by the user and $||\cdot||$ is any vectorial norm.

To improve the bid for item i, we need to calculate β_i. The formula, however, requires exponential computation. It is possible to approximate β_i by price sampling. The following presents the pseudo-code to approximate β_i, where K is the number of price samples generated from the closing price distributions for all the items.

$\beta \leftarrow 0$
for $k = 1..K$ **do**
  $P \leftarrow$ generatePriceSamples$(F_1, \ldots, F_N)$
  $\beta \leftarrow \beta + \left(\vartheta\left(\mathcal{I}_{ac}(\overline{p_i} \vee B_{-i}, P)\right) - \vartheta\left(\mathcal{I}_{ac}(\underline{p_i} \vee B_{-i}, P)\right)\right)$
end for
return $\frac{\beta}{K}$.

Three variants of MDBI are available: **Random Start Bid Improver (RSMDBI):** RSMDBI starts with a randomly chosen bid vector and does not use restarts. **Random Start Bid Improver With Restart (RSMDBIWRn):** RSMDBIWRn restarts the improvement process with random bid vectors $n-1$ times and outputs the bid vector with the highest expected utility. **Valuation Start Bid Improver (VSMDBI):** VSMDBI uses the initial bid $B_0 = (v_1, \ldots, v_N)$ where $v_i = \vartheta(\{i\})$ and does not use restarts.

The complexity of the approximation of $\beta_i(B_{-i})$ is linear given K, the number of samples. In MDBI, each N-sequential improvement requires N approximations of $\beta_i(B_{-i})$. The number of N-sequential improvements, $C(\varepsilon, N)$, correspond to the number of iterations of the improvement loop. Thus, the complexity of MDBI is $O(K\, N\, C(\varepsilon, N))$. Experimental results presented in Section 5 shows that the actual run-time complexity of MDBI is satisfactory.

5. EXPERIMENTAL RESULTS

Our experimentation goal is to compare the efficiency of different variants of the MDBI scheme with variants of MU bidding both in terms of the quality of the solution generated and time efficiency. To assess the absolute performances of these algorithms, we use an exhaustive search-based brute force algorithm (BF) which is asymptotically optimal

We ran our experiments in an environment containing four single-item single-unit auctions. All closing prices are drawn from selected discrete closing price distributions. A simulation consists of one bidder with the knowledge of all closing price distributions. A run of our experiment consists

Strategy	Score
RSMDBIWR10	1940.29
BF	1909.42
RSMDBIWR5	1907.25
VSMDBI	1859.65
RSMDBI	1856.85
EVMU	1708.18
$\overline{MU}$	1600.59

(a) SI

Strategy	Score
BF	7693.41
RSMDBIWR5	7693.32
RSMDBIWR10	7674.74
VSMDBI	7513.83
RSMDBI	7434.81
EVMU	7413.88
$\overline{MU}$	7202.51

(b) CI

Strategy	Score
RSMDBI	3748.42
VSMDBI	3731.57
RSMDBIWR10	3725.22
RSMDBIWR5	3724.76
$\overline{MU}$	3715.95
BF	3658.38
EVMU	3522.67

(c) NRI

Strategy	Score
RSMDBIWR10	6898.97
BF	6871.01
RSMDBIWR5	6828.67
RSMDBI	6239.84
EVMU	6224.37
VSMDBI	5946.33
$\overline{MU}$	5184.76

(d) RI

Table 1: Cumulative profits of different algorithms.

of seven simulations, one for each of the bidders RSMDBI, RSMDBIWRn with n=5 or 10, VSMDBI, BF, $\overline{MU}$, EVMU. For a run, bidders in each simulation share the same valuation function. We generate four kinds of valuation functions: (a) SI, where items are substitutable, (b) CI, where items are complementary, (c) NRI, where items are non-related, and (d) RI, where valuations for bundles are random.

5.1 Quality of solutions generated by algorithms

We present the cumulative average profit of the bidders using each algorithm in Table 1. The algorithms are ordered by the profits they generate. Algorithms with no statistical performance difference are grouped together. To obtain such groupings, we consider the average profits obtained by our algorithms as random variables and use the Wilcoxon test to identify the significance of the difference in their values.

We highlight the following observations from Table 1: (1) RSMDBI performs similar to or better than previously known algorithms. (2) VSMDBI performs similar to RSMDBI except for RI valuations where performances are worse. (3) With few random restarts our algorithm always performs similar to BF which is asymptotically optimal. (4) RSMDBIWRn, $n = 5, 10$ has better performances than RSMDBI except for non-related items where performances are equal. (5) Every algorithm is optimal when items are non-related except EVMU, (6) EVMU performs better than $\overline{MU}$ except when items are non-related when its performance is worse.

5.2 Time efficiency of MDBI variants

We discussed the complexity of MDBI in Section 4.3. The expression of this complexity contains an unknown function $C(\varepsilon, N)$. We wanted to find out the complexity of MDBI variants in practice. We run experiments using the RSMDBI and VSMDBI by varying the number of items, N. The time complexity of VSMDBI and RSMDBI appears to be linear in N. Since the complexity of MDBI is equal to $O(K\, N\, C(\varepsilon, N))$, $C(\varepsilon, N)$ is constant for VSMDBI given N. We also collected the average value of $C(\varepsilon, N)$ from our experiments and can highlight the following observations: (1) Except for non-related items, the number of N-sequential improvements ($C(\varepsilon, N)$) increases very slowly. The largest range is [2, 4], (2) The number of N-sequential improvements is always better for VSMDBI, (3) $C(\varepsilon, N) = 1$ for VSMDBI and $C(\varepsilon, N) = 2$ for RSMDBI when items are non-related.

6. CONCLUSION

We compared the performance of an optimization based bidding algorithm for obtaining bundles of items in simultaneous auctions, MDBI, that we have recently developed with existing algorithms from literature. We discuss variants of the basic iterative bid-improvement algorithm: an approach with finite number of restarts and one with no restarts but a carefully chosen starting bid vector. Performance similar to or better than existing algorithms in literature, $\overline{MU}$ and EVMU, is achieved without restarts when items are substitutable, complementary or non-related. We additionally observe that a small number of restarts can be used to approximate optimal expected utility.

Another strong point of the MDBI variants is their polynomial time complexities whereas existing algorithms have exponential complexity. While exact calculation of bid improvement needs combinatorial number of operations, we approximate such calculations in linear time by a constant number of price samples. We are currently working on formally characterizing the process and the properties of the MDBI algorithms.

Acknowledgments: This work has been supported in part by an NSF award IIS-0209208.

7. REFERENCES

[1] T. Candale. Hill-climbing approach to bidding for bundles in simultaneous auctions. Master's thesis, University of Tulsa, August 2005.

[2] A. Greenwald and J. Boyan. Bidding under uncertainty: theory and experiments. In *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 209–216. AUAI Press, 2004.

[3] A. Greenwald and J. Boyan. Bidding algorithms for simultaneous auctions: A case study. *Journal of Autonomous Agents and Multiagent Systems*, 10(1):67–89, 2005.

[4] A. Greenwald and P. Stone. Autonomous bidding agents in the trading agent competition. *IEEE Internet Computing*, 5(2):52–60, 2001.

[5] T. Sandholm. emediator: a next generation electronic commerce server. In *AGENTS '00: Proceedings of the fourth international conference on Autonomous agents*, pages 341–348. ACM Press, 2000.

[6] S. Sen, T. Candale, and S. Basak. Profit sharing auction. In *Proceedings of the Twentieth International Conference on Artificial Intelligence (AAAI-2005)*, July 2005.

[7] P. Stone and A. Greenwald. The first international trading agent competition: Autonomous bidding agents. *Journal of Electronic Commerce Research*, 2003.

[8] P. Stone, R. E. Schapire, M. L. Littman, J. A. Csirik, and D. McAllester. Decision-theoretic bidding based on learned density models in simultaneous, interacting auctions. *Journal of Artificial Intelligence Research*, 19:209–242, 2003.