Bid Determination in Simultaneous Auctions

An Agent Architecture

Justin Boyan
ITA Software
One Kendall Square, Bldg 400
Cambridge, MA 02139
justin@boyan.com

Amy Greenwald
Department of Computer Science
Brown University
Providence, RI 02912
amygreen@cs.brown.edu

1. INTRODUCTION

Suppose you want to buy a used Canon AE-1 SLR camera and flash at an on-line auction. At last count, over 4000 links to on-line auction sites were available at `advocacy-net.com`. It would be quite a daunting task to manually monitor prices and make bidding decisions at all sites currently offering the camera—especially if the flash accessory is sometimes bundled with the camera, and sometimes auctioned separately. But for the next generation of automated trading agents, this will be a routine task.

Simultaneous auctions, or *parallel auctions*, present a challenge to bidders, particularly when complementary and substitutable goods are on offer. Complementary goods are items such as a flash, a tripod, and a case, that would complement a camera—but a bidder desires any of the former only if s/he is certain to acquire the latter. Substitutable goods are goods such as the Canon AE-1 and the Canon A-1—a bidder desires one or the other, but not both. In contrast to *combinatorial auctions*, in which bids may be placed for combinations of items (e.g., "camera and flash for $295"), simultaneous auctions require separate bids to be placed for each individual item. In combinatorial auctions, the problem of choosing a set of winning bids that maximizes revenue—the so-called winner determination problem (WD)—falls in the hands of the auctioneer; in simultaneous auctions, the complexity burden lies with the bidders.

In this paper, we present an agent architecture for intelligent bidding in simultaneous auctions. Our architecture is centered on a class of problems we call *bid determination* (BD): determining what bids to place in simultaneous auctions, given current market conditions and utilities on combinations of items. We present two theoretical results: 1. BD in double auctions, where goods can be sold as well as bought, can be formally reduced to the problem of BD in single-sided auctions. 2. BD problems in simultaneous auctions are isomorphic to common variants of WD in combinatorial auctions. In a longer version of this paper, we propose and experiment with algorithmic approaches to BD.

EC'01, October 14-17, 2001, Tampa, Florida, USA.

2. BID DETERMINATION

The key challenge that bidding agents face in simultaneous auctions is determining how to bid on complementary and substitutable goods. Complementary goods are goods with superadditive utilities: *i.e.*, $u(A\bar{B}) + u(\bar{A}B) < u(AB)$. For example, in TAC, the utility of airline tickets without hotel reservations (or of hotel reservations without airline tickets) is zero, whereas the utility of complete travel packages is strictly positive. Substitutable goods are goods with subadditive utilities: $u(A\bar{B}) + u(\bar{A}B) > u(AB)$. For example, in TAC, the utility of both a theater ticket and a symphony ticket for the same night is bounded above by the higher of the individual utilities of the two separate events. It does not make sense to assign individual utilities to complementary goods (which are worthless in isolation) or substitutable goods (which are worthwhile only in isolation). Thus, simple bidding strategies such as "for each good x in auction x, bid up to its utility" are inapplicable in simultaneous auctions.

Instead, bidding agents must reason directly about *sets* of goods—the utilities of which are well-defined. While making bidding decisions in simultaneous auctions, a bidding agent may pose and solve questions such as the following:

- "Given only the set of goods I already hold, what is the maximum utility I can attain?" We call this the *allocation* problem, since the agent's utility is determined by the choice of how to allocate its set of held goods into useful subsets.
- "Given the set of goods I already hold, and given market prices and supply in all open auctions, what set of additional goods should I acquire so as to maximize my utility less purchase costs?" This more general problem, which we term *acquisition*, provides a foundation for bidding strategies in single-sided auctions.
- "Given the set of goods I already hold, and given market prices, supply, and *demand*, on what set of additional goods should I place bids or *asks* so as to maximize my utility plus profits less costs?" This yet more general problem, *completion*, provides a foundation for bidding strategies in settings with simultaneous single-sided and double auctions.

2.1 Agent Architecture

An agent bidding in simultaneous auctions must decide (i) on what goods to bid, (ii) for how many to bid, (iii) at what price to bid, and (iv) when to bid—and likewise for asks. A completion directly answers the first two of these

(A) While some auction remains open, do:

1. **Update** current prices and holdings for each auction.
2. **Estimate** clearing prices, supply & demand of goods.
3. Run **completion** to determine the quantity of each good that is ultimately desired; compute the difference between the optimal solution and current holdings.
4. Place **bids** and **asks** strategically (with respect to current time and the auction mechanisms) to buy and sell goods to reach the desired quantities.

(B) After all auctions have closed, run **allocation**.

Table 1: A high-level architecture for bidding agents in simultaneous auctions.

questions, and it provides a partial answer to third question, namely bid (ask) no more (less) than the market price. (The fourth question, which depends on the specific auction mechanisms, is addressed in [3].) Thus, a natural architecture for a bidding agent is to repeatedly compute estimates of market clearing prices, run a completion algorithm to determine target holdings, and bid/ask accordingly. This architecture is outlined in Table 1 for a simultaneous auction setup in which there is no advantage to allocating goods early.

The remainder of this paper focuses on the computational challenges of completion, acquisition (the special case of completion applicable to single-sided auctions), and allocation. Two other components of our architecture—estimation (step A2) and bid/ask tactics (step A4)—are equally challenging but problem-dependent, and are not addressed here. (For some insights, see [2, 3, 7]).

2.2 Formal Problem Statement

We now formally define allocation, acquisition, and completion, starting with acquisition. Consider an agent bidding in simultaneous auctions for goods g in the set G. For each good g, we are given the agent's current holdings of g, the market supply, and the prices of additional copies of g. We represent these holdings and estimates by a single vector $\vec{p}_g$, which we call a *buyer priceline*, in which the nth component, p_{gn}, stores the cost the agent would incur upon acquiring the nth copy of g. For example, if the agent currently holds four units of a good $\hat{g}$, and predicts that two additional units of $\hat{g}$ could be won at costs of \$20 and \$30, respectively, the corresponding buyer priceline is given by $\vec{p}_{\hat{g}} = \langle 0, 0, 0, 0, 20, 30, \infty, \infty, \ldots \rangle$. The leading zeroes indicate that the agent may "acquire" the four goods it already holds at no cost. The tail of infinite costs means that the agent may acquire no more than six units of $\hat{g}$ in total. We assume buyer pricelines are nonnegative and nondecreasing.

In acquisition, the agent's goal is to determine how best to augment its holdings with new purchases so as to maximize the sum of the package utilities it can achieve less the total cost of the goods. Let a package be represented by a vector of quantities, one for each good, plus a unique identifier:[1] $\vec{q} = \langle q_1, \ldots, q_{|G|}; q_{\text{id}} \rangle$. The package's utility is denoted $u(\vec{q})$.

[1]The unique identifier serves to guarantee that the utility function is indeed a function even in the case where multiple packages contain the same goods but have different utilities.

Given a set of buyer pricelines $P = \{\vec{p}_g \mid g \in G\}$ and a set of packages S, we define the utility and cost of S as follows:

$$\text{Util}(S) = \sum_{\vec{q} \in S} u(\vec{q}) \qquad (1)$$

$$\forall g, \quad \text{Used}(S, g) = \sum_{\vec{q} \in S} q_g \qquad (2)$$

$$\forall g, \quad \text{Cost}_g(S, P) = \sum_{n=1}^{\text{Used}(S,g)} p_{gn} \qquad (3)$$

$$\text{Cost}(S, P) = \sum_{g \in G} \text{Cost}_g(S, P) \qquad (4)$$

Definition 2.1. ***ACQ***(P, Q, u). ***Inputs****: a set of buyer pricelines* $P = \{\vec{p}_g \mid g \in G\}$*; a set of packages* Q*; a utility function* $u : Q \to \mathbb{R}^+$. ***Output****: an optimal subset of packages* $S^* \in \arg\max_{S \subseteq Q} (\mathit{Util}(S) - \mathit{Cost}(S, P))$.

Definition 2.2. ***Allocation*** *is simply a special case of the acquisition problem in which no additional goods can be purchased. That is, all entries* p_{gn} *in the buyer pricelines are assumed to be either* 0 *or* ∞.

Completion generalizes acquisition to double-sided auction scenarios in which the agent may sell some of the goods it holds, if it so chooses, rather than using them in packages. Let us represent the estimated market demand for each good g the agent owns as a vector $\vec{\pi}_g$, which we call a *seller priceline*. Much like a buyer priceline, the nth component of a seller priceline stores the profit the agent could earn by selling its nth copy of g on the open market. For example, returning to the sample good $\hat{g}$ above, suppose the agent estimated that it could find buyers for two of its four copies of $\hat{g}$ at prices of \$10 and \$5, respectively. Its seller priceline would be given by: $\vec{\pi}_{\hat{g}} = \langle 10, 5, 0, 0 \rangle$ where the trailing zeroes reflect that there is no market demand for the third and fourth copies of $\hat{g}$. Note that seller pricelines have length equal to the quantity of the good held by the agent. We assume that seller pricelines are nonnegative and nonincreasing.

Given a set of seller pricelines $\Pi = \{\vec{\pi}_g \mid g \in G\}$ and a set of packages S, we now define profit analogously to cost:

$$\forall g, \quad \text{Unused}(S, g, \Pi) = \max\{|\vec{\pi}_g| - \text{Used}(S, g), 0\} \qquad (5)$$

$$\forall g, \quad \text{Profit}_g(S, \Pi) = \sum_{i=1}^{\text{Unused}(S,g)} \pi_{gn} \qquad (6)$$

$$\text{Profit}(S, \Pi) = \sum_{g \in G} \text{Profit}_g(S, \Pi) \qquad (7)$$

Definition 2.3. ***COM***(P, Π, Q, u). ***Inputs****: a set of buyer pricelines* $P = \{\vec{p}_g \mid g \in G\}$*; a set of seller pricelines* $\Pi = \{\vec{\pi}_g \mid g \in G\}$*; a set of packages* Q*; a utility function* $u : Q \to \mathbb{R}^+$. ***Output****: an optimal subset of packages* $S^* \in \arg\max_{S \subseteq Q} (\mathit{Util}(S) - \mathit{Cost}(S, P) + \mathit{Profit}(S, \Pi))$.

2.3 Reducing Completion to Acquisition

Perhaps surprisingly, completion problems (for double-sided simultaneous auctions) can be cast as acquisition (for single-sided auctions). We demonstrate this equivalence in two ways: by mapping the seller pricelines into dummy packages, or by augmenting the pricelines with the information contained in the seller pricelines.

In the first mapping, we treat the market as an additional source of utility: each nonzero number π_{gn} on each seller priceline translates into a new package $\vec{e}_{gn}$ containing only the single item g, with corresponding utility $u(\vec{e}_{gn}) = \pi_{gn}$. For example, given the example seller priceline $\vec{\pi}_{\hat{g}} = \langle 10, 5, 0, 0 \rangle$, we would create two dummy packages, $\vec{e}_{\hat{g}1}$ and $\vec{e}_{\hat{g}2}$, with utilities 10 and 5, respectively.

THEOREM 2.4. *Given P, Π, Q, u, $COM(P, \Pi, Q, u) = ACQ(P, Q \cup Q', u \cup u') \cap Q$, where $Q' = \{\vec{e}_{gn} \mid g \in G, n = 1 \ldots |\vec{\pi}_g|\}$ and $u' = \{\vec{e}_{gn} \mapsto \pi_{gn} \mid g \in G, n = 1 \ldots |\vec{\pi}_g|\}$.*

In the second mapping from completion to acquisition, the package-utility pairs are unchanged. Instead, we merge the seller pricelines into the buyer pricelines. The intuition here is that in a double-auction scenario, an agent cannot use even the goods it holds at no cost; rather, the agent incurs *opportunity costs* for not selling those goods on the market.

Combining a buyer priceline $\vec{p}_g$ and a seller priceline $\vec{\pi}_g$ is a simple matter of replacing the leading $|\vec{\pi}_g|$ entries of $\vec{p}_g$—which are originally all zero, since they represent the cost of acquiring the agent's own holdings—with the contents of $\vec{\pi}_g$ in reverse. Consider the sample buyer priceline $\vec{p}_{\hat{g}}$ and sample seller priceline $\vec{\pi}_{\hat{g}}$ shown earlier. Combining these two pricelines produces the following result: $\vec{p}_{\hat{g}} + reverse(\vec{\pi}_{\hat{g}}) = \langle 0, 0, 5, 10, 20, 30, \infty, \infty, \infty, \ldots \rangle$. The resulting priceline should again be nonnegative and nondecreasing. (Otherwise, there would exist an arbitrage opportunity, which could be exploited in a preprocessing step.)

The interpretation of the new priceline is as follows: the agent can use one or two copies of good $\hat{g}$ without penalty, since in either case it will still be able to sell at most two further copies of $\hat{g}$. If it uses a third copy, though, it must charge itself \$5 in lost profits. If it uses all four of its held copies, it must charge itself \$5+\$10=\$15. If it uses five, it incurs the lost opportunity cost of \$15, plus the \$20 cost of purchasing an additional copy on the open market; for six, it incurs cost \$65. Finally, lack of market supply (represented by ∞) prevents the agent from using more than six copies of the good in total. We can now state our second theorem.

THEOREM 2.5. *For all P, Π, Q, u, $COM(P, \Pi, Q, u) = ACQ(P', Q, u)$, where $P' = \{\vec{p}_g + reverse(\vec{\pi}_g) \mid g \in G\}$.*

2.4 Winner Determination

The BD problems formulated above are isomorphic to variants of the winner determination (WD) problems that arise in combinatorial auctions. In combinatorial auctions (see, for example, [1, 6]), an auctioneer collects bids on combinations (*i.e.*, packages) of goods, and then seeks the most profitable subset of those bids—the "winners"—that can be fulfilled from the set of available goods. Combinatorial bids correspond exactly to our notion of package utilities in the allocation problem. Similarly, the bidding agent's goal in allocation and the auctioneer's goal in WD are identical: to choose a subset of bids/packages that maximizes utility while respecting the constraint on total goods available. Thus, allocation is isomorphic to WD in (multi-unit) combinatorial auctions.

More generally, the tasks of acquisition and completion are equivalent to *winner determination with reserve prices*. In this latter problem, it is assumed that there is some price below which the auctioneer prefers not to sell each good. Thus, the auctioneer seeks to maximize the difference between profits and reserve prices. Similarly, in acquisition and completion tasks, bidding agents seek to maximize the difference between utilities and market prices. WD problems are themselves equivalent to maximum weighted set-packing (and maximum weighted clique), and are therefore NP-complete [5]. Hence, allocation, acquisition, and completion are NP-complete.

As an aside, we note that implementations of winner determination have typically been evaluated on randomly generated datasets [4], since data from large-scale combinatorial auctions is scarce. (One obvious exception is the FCC spectrum auction.) The equivalence between BD and WD makes new datasets available for testing by the combinatorial auction community—for example, the data generated by the Trading Agent Competition (see, for example, [3]).

3. CONCLUSION

In this paper we have: (i) proposed an architecture for intelligent bidding in simultaneous auctions; and (ii) defined a class of "bid determination" problems at the core of the proposed architecture. The principal advantage of our architecture is its modular design. Moreover, its core components, allocation and completion, are well-defined with objective evaluation criteria. While BD problems form the core of the proposed architecture, the other major components—namely, estimation and bidding—are by no means trivial. Estimation requires predicting market dynamics on the basis of historical bid trajectories (taken over sets of goods, not only goods in isolation), and possibly modeling opponents' behavior. Likewise, the strategic placement of bids and asks, particularly with regard to their timing, depends on analysis of the auction mechanism and opponent modeling. Moreover, even if all three major problems (estimation, completion, and strategic bidding/asking) could be solved optimally, the resulting behavior still need not be optimal. Most egregiously, the architecture as shown does not explicitly plan for uncertainty in the future dynamics of the auctions. An architecture that would be superior in this respect would bid based on distributions over estimated clearing prices, rather than on expected values. We see this as an exciting direction for future research in this area.

4. REFERENCES

[1] Y. Fujishima, K. Leyton-Brown, and Y. Shoham. Taming the computational complexity of combinatorial auctions. In *Proceedings of Sixteenth International Joint Conference on Artificial Intelligence*, pages 548–553, August 1999.

[2] A. Greenwald and J. Boyan. Roxybot: An autonomous bidding agent for simultaneous auctions. In *Proceedings of Third ACM Conference on Electronic Commerce*, to appear, October 2001.

[3] A. Greenwald and P. Stone. Autonomous bidding agents in the Trading Agent Competition. *IEEE Internet Computing*, April 2001.

[4] K. Leyton-Brown, M. Pearson, and Y. Shoham. Towards a universal test suite for combinatorial auction algorithms. In *Proceedings of Second ACM Conference on Electronic Commerce*, pages 66–76, October 2000.

[5] M. H. Rothkopf, A. Pekeč, and R. M. Harstad. Computationally manageable combinatorial auctions. *Management Science*, 44(8), 1998.

[6] T. Sandholm and S. Suri. Improved algorithms for optimal winner determination in combinatorial auctions and generalizations. In *Proceedings of AAAI*, pages 90–97, 2000.

[7] P. Stone, M. L. Littman, S. Singh, and M. Kearns. `ATTac`-2000: An adaptive autonomous bidding agent. In *Fifth International Conference on Autonomous Agents*, 2001.