An Auction for Scheduling Delivery of Highly Demanded Products

Sameer Korrapati & Peter R. Wurman
North Carolina State University
Department of Computer Science
Raleigh, North Carolina
sameer.korrapati@gmail.com, wurman@ncsu.edu

Abstract

We present a simple auction protocol for the sale of over-demanded goods with a delivery schedule. Under some reasonable assumptions, our mechanism is incentive compatible, collects only partial information about bidder's valuations, and empirical evidence suggests it converges to the efficient allocation.

1 Introduction

Many situations arise in which the supply of a product does not meet the initial demand, and in which there is some real or perceived value accrued by obtaining the object earlier rather than later. A recent example was the introduction of the Sony PlayStation 2 (PS2), a highly anticipated new product whose introduction in the U.S. was timed to coincide with the Christmas 2000 holiday shopping season. Component supply problems caused Sony to have only half of its target one million units available [3], which was already projected to be well under the pent-up demand. All of this played out in the media, which meant that consumers were well aware that demand far exceeded supply when the product was released on October 26. Thus, the evening before the introduction, consumers lined up outside of stores, and most retailers sold out within minutes of opening the next morning. A great many of these purchasers turned around and sold the products on eBay for three to four times retail price. Prices were highest closest to the product introduction but continued to demonstrate higher-than-retail prices right through the Christmas holiday.

The scenario that played out with the PS2's is an extreme example of a common phenomenon seen with the release of game consoles, automobiles, and other luxury items. Of interest here is that fact that a highly liquid market was created on eBay in which the PS2's were resold at their true market prices, and Sony captured none of the value. Instead, the difference between the retail price of $299 and the market price was captured by high school and college students who had the time to camp out overnight in front of retail outlets.[1] While there are many factors that go into Sony's marketing decisions, not the least of which is that consoles are sold at a loss and profits made by selling games, it is interesting to ask how a manufacturer could have captured more of the actual demand in the market.

In this paper we study a model of highly demanded products in which buyers are willing to pay more to acquire the object earlier. We assume that the supplier has a finite number of units that can be delivered each day, and that buyers have greater utility for objects delivered sooner rather than later. Our objectives are to create an iterative mechanism which maximizes the social value created by the allocation while being incentive compatible, having a simple interface, and not requiring buyers to compute and communicate their values for every possible delivery day. An efficient allocation is one in which the goods are allocated to the persons who value them the most. Thus, reaching an efficient allocation implies that the participants have no reason to re-trade the goods among themselves, as was the case with the PS2.

One way of achieving social efficiency would be to conduct a sealed bid auction and then compute the best allocation and prices. However, this is unlikely to work in a case like Sony's because it relies upon bidders being able to accurately establish their valuation for acquiring the product on every delivery day in the horizon without feedback from the marketplace. An alternative design would be to conduct a different auction for each of the delivery days, where each auction could be run as an English auction. While there are many variations of this design (say, with progressive auction closings, conditional bids, etc.), in general this approach is cumbersome to operate and to bid in, and is not guaranteed to find an efficient allocation.

Our design is an iterative auction in which the auctioneer sells a supply of multiple units of otherwise homogeneous

[1] Although this is a generalization, the second author still asks students in his classes if they participated in this secondary market and invariably finds at least one.

goods differentiated by the fact that they are delivered on different days. Bidders have only one bid in the auction at a given time, and the auctioneer announces prices in such a way that every bidder can unambiguously determine on which day they will take delivery. Moreover, prices are set in such a way as to be incentive compatible under some assumptions on bidder valuations. Our design is simple for the bidders–they make myopic decisions in response to price announcements–and easier to conduct for the auctioneer, who needs only keep a sorted list of all bids.

2 Model

The setting we consider is a company that has a limited number of indivisible units available to be auctioned. It can supply these goods spread across different days, with possibly multiple units on each day. We wish to assign these goods to bidders who demand at most one unit. We also wish to price the goods anonymously, meaning that all the bidders who receive a good on a given delivery day will pay the same price for it. Anonymous prices make the auction transparent to the bidders and engender a sense of fairness. The bidders have their own private valuations for these goods, and typically, different values for delivery on different days. A reasonable assumption that we make is that valuations decrease for delivery on later days. Thus, we expect that the price of the goods determined by the auction decreases with the delivery days.

The auction can be conducted anytime before the actual delivery. Bidders register with the auctioneer to participate in the auction. The auction proceeds in multiple iterations and terminates after a finite number of iterations when no participating bidder changes her bid, indicating that the bidders are satisfied with the outcome reached. The price vector declared by the auctioneer at the end of the last iteration establishes the prices of the items for each of the delivery days. The allocation is the one reached in this last iteration.

To formalize the auction, we introduce the following notation: Let B be the set of bidders who wish to participate in the auction. Let $d = 1, 2, ...m$ denote the delivery days, in sequence. Let Q be the total quantity of items (discrete units) available to be auctioned off; these items are available in limited quantities on each of the delivery days. Let q_d denote the number of units available on day d, such that $\sum_{d=1}^{m} q_d = Q$. The auction must decide the allocation of the goods among the bidders and determine a price, p_d, for each delivery day, d. Because our pricing scheme is anonymous, all bidders who win an item on day d pay price p_d.

Each bidder j demands at most one unit of the base good. Since a bidder's value for a particular unit of the good depends on the day on which it is delivered, the goods available on different days are treated as heterogeneous. Each bidder j has a valuation profile, v_j, with $v_j(d)$ being the valuation assigned by bidder j for receiving an item on day d. Without loss of generality, we assume that the item will become available for its list price on day $m+1$, and the bidders' valuations represent the excess monetary value they are ready to pay to get the item early. For normalization purposes, we define $v_j(d) = 0$ for $d > m, \forall j \in B$.

We make the standard assumptions that bidders have quasi-linear utility, risk neutrality, and private values. Further, we make several assumptions that are unique to our problem domain. First, bidders have valuations that are monotonically decreasing in delivery days. That is, for every bidder j and all delivery days, d, $v_j(d) > v_j(d+1)$. Second, when deriving the incentive compatibility results, we assume that bidders have non-crossing valuations. Non-crossing valuations means that the rank order of bidders, when sorted by their valuations, is the same on all days. That is, $v_j(d) > v_{j'}(d) \Rightarrow v_j(d') > v_{j'}(d')$, for all $j, j' \in B$ and $d, d' \in 1...m$.

Our objective is to find the efficient solution, that is, the one that achieves

$$\begin{aligned} \max \quad & \sum_j \sum_d v_j(d) x_{jd} && (1) \\ \text{s.t.} \quad & \sum_j x_{jd} \le q_d \quad \forall d, \\ & \sum_d x_{jd} \le 1 \quad \forall j. \end{aligned}$$

In (1), $x_{jd} \in \{0, 1\}$ is the decision variable that takes on the value one if bidder j obtains an object delivered on day d, and zero otherwise. Let f be a feasible solution, and $f*$ be a solution that satisfies equation (1). Let $f(j)$ be the day assigned to bidder j by solution f.

3 Auction Design

The interface between the auctioneer and the bidders is through prices and bids. In our mechanism, the auctioneer computes and reveals a tuple $\{p_d^{\text{bid}}, p_d^{\text{ask}}\}$ for each delivery day. p_d^{bid} is the price the bidder will pay for a good delivered on day d. p_d^{ask} is the minimum price a bidder needs to offer to win an item on day d. In this sense, the pricing model is akin to $(M+1)$st pricing. We construct bid and ask prices such that they are non-increasing in delivery days. That is, $p_d^{\text{bid}} \ge p_{d+1}^{\text{bid}}$ and $p_d^{\text{ask}} \ge p_{d+1}^{\text{ask}}$. For convenience, we define $p_0^{\text{ask}} = \infty$ and $p_{m+1}^{\text{ask}} = 0$.

Each bidder makes an offer of the form $\{r_j, \delta_j\}$, where r_j is the amount that bidder j is willing to pay for delivery on day d, and δ_j is the amount more than r_j that the bidder is willing to pay to upgrade to delivery on day $d-1$. In other words, $r_j(d-1) = r_j(d) + \delta_j$. Note that d does not have to be stated explicitly because it is implied by the current ask prices. The implied day is the day for which

$p_d^{\text{ask}} \leq r_j < p_{d-1}^{\text{ask}}$. When a bid $\{r_j, \delta_j\}$ is received, the auctioneer will infer the implied delivery day and record it with the bid in the form $\{r_j, \delta_j, \overrightarrow{d_j}\}$. Importantly, $\overrightarrow{d_j}$ is the day that is implied *at the time the bid was made*; if prices change, r_j may no longer be in the range of ask prices that corresponds to $\overrightarrow{d_j}$. For the boundary condition where $d = 1$, we set $\delta_j = 0$.

We now consider the policy by which the auctioneer computes prices, and a myopic bidding strategy that bidders may use and which, we show in Section 4, is incentive compatible.

3.1 Auctioneer's Policy

The auctioneer keeps only the most recent offer from each bidder. After each new bid, the auctioneer computes the state of the auction (price vectors and winning bidders), and announces the prices to the bidders. Prices are all zero at the start of the auction.

The general procedure for computing the state of the auction is to sort the bids decreasing in r_j. The auctioneer then assigns bidders to days, with the highest bidders going to the earliest days, until all of the objects have been assigned. Ties are broken first by the larger δ_j, then in favor of the earlier bid. In this process, the auctioneer ignores $\overrightarrow{d_j}$.

Once the bids are sorted, and the boundaries between days established, the auctioneer computes the bid and ask prices. The computation of the ask prices is designed to communicate the information necessary to the bidders for them to know the mapping between offer prices and delivery days. The computation of the bid prices is inspired by the Generalized Vickrey Auction.

Let $H(d)$ and and $L(d)$ be the highest bidder and lowest bidder, respectively, assigned to day d. Except in a special case when there is only one item available on a particular delivery day, the ask price is simply the lowest offer on that day. That is,

$$p_d^{\text{ask}} = r_{L(d)}$$

The bid price is only slightly more complex to compute:

$$p_d^{\text{bid}} = \sum_{k=d+1}^{m+1} \delta_{H(k)}. \tag{2}$$

In words, the bid price is simply the sum of the delta's of highest bidders on each day subsequent to d. Note that this computation includes day $m + 1$, which is past the horizon. It is necessary to include the highest bidder excluded from any allocation in order to accomplish our goal of generating GVA prices. We elaborate on the relationship between equation (2) and GVA prices in Section 4. Although this computation of the bid price may seem indirect, it is completely hidden from the bidders and produces a meaningful price that correctly guides their bidding decisions.

Bidder	Day 1	Day 2
b_1	100	90
b_2	90	82
b_3	83	74
b_4	70	66
b_5	60	58

Table 1. Example valuations.

Bidder	Day 1	Day 2
b_1	100	
b_2	(+8)	82
b_3	(+9)	74
b_4	(+4)	66
b_5		(+58)

Table 2. Example bids.

Notice that in general, the auctioneer is not respecting the implied delivery date, $\overrightarrow{d_j}$ and thus making assumptions about the bidder's valuations that may not be correct. For example, the very first bidder to bid will see prices of zero and may make an offer for day 1. However, after many other bidders have made offers, the first bidder's offer may now be scheduled for day 10, and for the auctioneer to make price announcements based upon the stated value is clearly incorrect. We allow this behavior because it is an *out-of-equilibrium* effect that disappears when the bidders reach equilibrium. Although the prices are temporarily misleading, once the first bidder has a chance to bid again, she will revise her bid with respect to the current ask prices and put in a more appropriate offer.

There is one special case that, when handled properly, allows the mechanism to work for a wider range of valuation functions. In particular, it is possible that the best allocation assigns bidders to days in a manner that is inconsistent with their sorted order even if their valuation functions do not cross. Table 1 shows a case in which there are five bidders and two days with two items available each day. The best allocation is to give the day 1 objects to bidders b_1 and b_3, and the day 2 objects to b_2 and b_4.

Suppose at an intermediate state the auctioneer had collected the information in Table 2, where the non-parenthetical number is r_j placed on the implied delivery day, and the parenthetical number is δ_j placed on $\overrightarrow{d_j} - 1$. Sorting by r_j and making assignments gives bidders b_1 and b_2 objects delivered on day 1, and b_3 and b_4 objects on day 2. Further, the auctioneer computes day 1 prices $\{67, 82\}$ and day 2 prices $\{58, 66\}$. Given these prices, b_2 receives a surplus of 23 from the day 1 object, and a surplus of 24 from a day 2 object. Thus, b_2 would prefer to purchase a day 2 object, but has already expressed his day 2 value and cannot further influence the auction. Nor is b_3 motivated to change his bid because days 1 and 2 provide her the same surplus.

The auctioneer, however, has enough information to recognize that the preferred allocation has b_2 getting an object on day 2. Ignoring the other bidders and looking only at the two bidders on the border between day 1 and day 2, the auctioneer determines that giving day 1 to b_2 and day 2 to b_3 results in a combined value of $(82 + 8) + 74 = 164$, whereas the contrary allocation produces a combined value of $82 + (74 + 9) = 165$.

Thus, to facilitate moving from a near equilibrium situation into the true equilibrium, before announcing new prices we solve equation (1). Because the auctioneer has very limited information (i.e., two valuations from each bidder), it is sufficient to check only the day boundaries for binary swaps that can improve the allocation, and only when $f(j) \neq \overrightarrow{d_j}$ for one of the two bidders at a particular boundary.

More formally, let $i = L(d)$ and $h = H(d + 1)$. If the best allocation does not agree with the sorted order of the agents, that is, if $(r_i + \delta_i) + r_h < r_i + (r_h + \delta_h)$, then the auctioneer should swap the positions of i and h in the sorted list of bidders and adjust the price computations.

$$
\begin{aligned}
p_d^{\text{bid}} &= \delta_i + \sum_{k=d+1}^{m+1} \delta_{H(k)}, \\
p_d^{\text{ask}} &= r_i + \delta_h - \delta_i.
\end{aligned}
$$

In the example from Table 2, the allocation adjustment would result in b_3 winning day 1 and b_2 winning day 2, with day 1 prices $\{66, 83\}$. At these new prices, b_2 prefers day 2, and b_3 is indifferent between day 1 and day 2. Note that it is important to apply this adjustment from day m back to day 1 in order to ensure the integrity of the bid prices that depend upon later days.

The auction terminates when no bidders wish to change their current bids. In the simulations reported in Section 5, we iterate through all of the bidders in a random fashion until all of the agents are happy with the current allocation. In practice, we would expect the auction to have a rolling horizon, with delivery day d closing before the day on which those products must ship. Further, bidders may not be paying full attention to the auction at all times, and so more flexible closing criteria are needed.

In summary, the auctioneer's algorithm is:

1. Initialize the prices to zero for all d.

2. Declare the current prices. On receiving a new bid from bidder j, perform steps 3-6.

3. Sort the bidders according to r_j, δ_j, and time. Allocate products to bidders.

4. Adjust the allocation of the boundary bidders as necessary.

5. Calculate the new prices.

6. If the termination conditions are not met, return to step 2. Otherwise, announce final allocation and exit.

3.2 Bidder's Policy

In our simulations, the bidders follow a simple, myopic best-response strategy. In Section 4 we show that, in equilibrium, our method generates incentive compatible prices, from which we argue that this myopic strategy is a realistic strategy if it leads to the equilibrium.

A bidder's potential surplus from selecting day d is the difference between the bidder's valuation for d and the current bid price for d. In the myopic strategy, a bidder finds the day for which, given the current prices, her surplus is maximized, and for which her valuation is within the current ask price range.[2] More formally,

$$
\begin{aligned}
d^* \quad &= \quad \arg\max_d v_j(d) - p_d^{\text{bid}}, \\
&\text{s.t.} \quad p_d^{\text{ask}} \leq v_j(d) < p_{d-1}^{\text{ask}}.
\end{aligned}
$$

After identifying day d^* as the one that maximizes her payoff, the bidder offers her true value, $v_j(d^*)$, and her true upgrade value $\delta_j = v_j(d^* - 1) - v_j(d^*)$.

If the bidder's current desired delivery day is among those that maximize her utility, she sticks with it. If none of the payoffs are positive, bidder j places a bid of $\{0, v_j(m)\}$, that is, r_j = 0 and δ_j equal to her value for the mth day.

4 Auction Properties

In this section, we show that in equilibrium our mechanism produces incentive compatible prices, which means that no bidder has an incentive to deviate. This result also suggests that bidders will be willing to play the myopic strategy throughout the auction if doing so leads them to the efficient solution. Since we do not have a convergence proof yet, we support our claim that the myopic strategy converges with the experimental results in Section 5.

Vickrey [8], first established that second-price auctions are incentive compatible. Since then, many researchers have studied variations of the Vickrey auction, including work that is focused on multi-item scenarios with unit demand [4, 6] . It is now well established that the incentive compatibility feature continues to hold in the multi-item, unit-demand setting. Shapley and Shubik [7] observe that

[2] The bidder eliminates days for which her valuation is below the ask price because her truthful bid will not result in her winning the day under consideration.

the model always has an equilibrium and that the smallest price vector among all the equilibrium prices is unique, and "best" from the point of view of the bidders.

To prove that, at equilibrium, our auction is incentive compatible and achieves the efficient allocation, we leverage existing literature in the areas of economics and mechanism design. Bikchandani et al. [1] survey various auction environments and examine the connection between linear programming and Vickrey payments in auctions. Leonard [6] considered the assignment problem and derived the requirements for the final solution to be incentive compatible.

Equation (2) gives the pricing scheme in our auction, where all bidders winning on day d pay p_d^{bid}. It is not difficult to see the relationship to the pricing scheme of the GVA.

Consider a slightly simplified allocation problem with m items to be assigned to m delivery days, one per day. Let us assume, without loss of generality, that the bidders, when sorted in the decreasing order of their valuations over all these days, are $\{1, 2, ...m, m+1, ...n\}$. Because we restrict bidders to monotonically decreasing, non-crossing valuations, the sorted order of bidders remains the same on all the days. Further assume that in the efficient allocation, bidder j wins the item on day j. This means that H_j and L_j will be bidder j, as there is only one item on each day. When we consider the marginal economy without bidder j, and re-calculate the optimal allocation, bidder j's position will be taken by bidder $j+1$.

Recall that $f(j)$ is bidder j's allocation in f and $v_j(f(j))$ is bidder's valuation for her assignment. Let f^* be the efficient allocation that solves equation (1). Define

$$
\begin{aligned}
V(B) &= \max_f \sum_{j \in B} v_j(f(j)), && (3)\\
V(B\backslash j) &= \max_f \sum_{k \in B\backslash\{j\}} v_k(f(k)). && (4)
\end{aligned}
$$

Equation (4) defines the *marginal economy* without bidder j. That is, we consider the economy in which bidder j is not present, and solve the allocation problem. In the GVA, the payment by bidder j is given by

$$\pi_j = V(B\backslash j) - [V(B) - v_j(f^*(j))]. \tag{5}$$

With the assumption that bidder j wins the item on day j, equation (3) reduces to

$$V(B) = \sum_{d=1}^{m} v_d(d).$$

If we drop bidder j from the economy, equation (4) reduces to

$$V(B\backslash j) = \sum_{d=1}^{j-1} v_d(d) + \sum_{d=j+1}^{m+1} v_d(d-1),$$

because, on re-assignment, every bidder following j moves up one day, and bidder $m+1$ gets the last item. Recall that, with truthful bidding, $v_j(d-1) = v_j(d) + \delta_j$. Substituting, we get

$$V(B\backslash j) = \sum_{d=1}^{j-1} v_d(d) + \sum_{d=j+1}^{m+1} [v_d(d) + \delta_d].$$

Equation (5) becomes

$$
\begin{aligned}
\pi_j &= \left[\sum_{d=1}^{j-1} v_d(d) + \sum_{d=j+1}^{m+1} [v_d(d) + \delta_d]\right]\\
&\quad - \left[\sum_{d=1}^{m} v_d(d)\right] + v_j(j). && (6)
\end{aligned}
$$

After eliminating the common terms, the price paid by bidder j according to the GVA scheme is

$$p_j = \sum_{d=j+1}^{m+1} \delta_d.$$

Bidder d is H_d, thus the above equation is the same as equation (2).

In the above derivation, we considered the simplified case where there is just one item per day. The generalization to the case of multiple items per day is straightforward. Recall that, in our scheme, we consider the δs of the "border bidders" only. Identically, in the GVA, when an arbitrary bidder is removed from the system only those other bidders who are the highest on their respective days move forward one day and affect the computation of the payment. We conclude that our pricing scheme is the same as the GVA pricing scheme.

5 Experimental Results

We conducted experiments to analyze the performance of our mechanism. In our simulations, the auctioneer proceeds in rounds during which a bid is elicited from every bidder, in a random order. The state of the auction is recomputed after each new bid is placed. We considered test cases in which the valuation functions converge with increasing delivery days, diverge with days, and combinations of the two. In our experiments, the mechanism always reached the equilibrium, lending support to our claim that the mechanism converges with non-crossing valuations.

One of our design objectives was to create an auction that computes the efficient outcome with as little information from the bidders as possible. To measure this, we tracked the average number of days for which a bidder reveals her valuation to the auctioneer. Note that in our auction, when

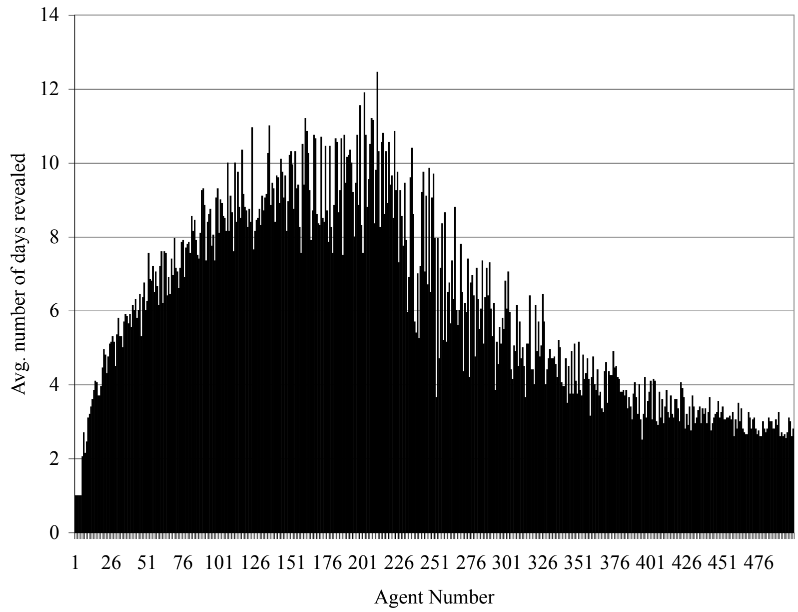

Figure 1. Average number of day's valuations exposed by each bidder.

a bidder places her bid for a particular day, she also reveals a δ, effectively revealing her valuation for two days with a single bid.

We constructed a scenario with 500 bidders in an auction to allocate 225 items spread across 45 days. We ran the auction 15 times and computed the average number of days of each bidder's valuations that are revealed. Out of 45 days, the bidders revealed their valuations for relatively few days, varying between 1 and 14, with an average of 6. Figure 1 shows the number of days revealed for all 500 bidders, with the bidders numbered from left to right by maximum valuation.

Observe that the highest bidders and the lowest bidders reveal relatively little information while the bidders in between expose more of their valuation functions. The lowest bidders, as they are not able to beat the ask-prices, are quickly pushed out of the auction. The highest bidders, once they reach their optimal places, are satisfied and do not change their bids, revealing very little information about later days. However, the bidders in the middle reveal relatively more information before they settle at their optimal positions. Figure 2 shows a typical pattern in the bids of some of the bidders in the middle. Note that the myopic best response behavior of these bidders causes them to pursue multiple different options before they settle at their optimal positions.

We also measured the number of iterations that it takes for our auction to converge. Our simulations produced very encouraging results, shown in Table 3. We simulated our auction with different problem sizes and observed the average number of rounds taken to converge to the equilibrium, which, in essence, is the average number of bidding opportunities for each bidder.

Bidders	Days	Items	Rounds
20	5	16	6
50	10	26	11
100	20	40	12
500	45	225	15
1000	100	500	28
1100	1000	1000	65

Table 3. Average number of rounds to converge with different problem sizes.

In order to measure the sensitivity of the mechanism to the inputs, we explored the effects of changing the three parameters (number of bidders, number of items, and delivery days). We observed that the number of delivery days has the most influence on the number of iterations to convergence. Table 4 shows the results when we varied the number of delivery days while keeping fixed the number of bidders at 600 and total number of items at 400. These observations are averaged over 12 trials of the auction and suggest a sublinear relationship.

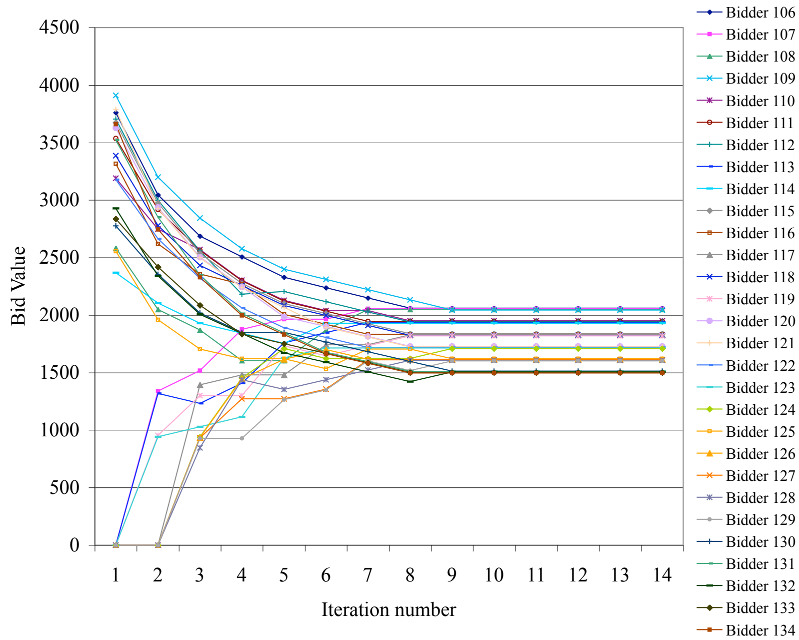

Figure 2. Selected agent's bids across iterations.

Number of days	Rounds
50	16
100	26
250	35
400	51

Table 4. Average number of rounds to convergence with 600 bidders and 400 items, as a function of number of delivery days.

6 Related Work

The linear programming model of our problem is very similar to the standard assignment model, which has been widely studied. Leonard [6] studied the problem of assigning individuals to positions under unit demand and proposed a procedure that elicits honest preferences and uses them to make an efficient assignment and to determine prices to be charged for the positions. He finds the prices that satisfy the Groves-Clarke [2, 5] definition of pricing, thus achieving incentive-compatibility. But this method requires the knowledge of the complete valuation profiles of all bidders. While our mechanism has theoretical similarity with this method, we describe a more practical auction design where requiring complete information revelation by distributed bidders is impractical. Our simulations have shown that our mechanism requires relatively little information from the bidders to compute the efficient allocation and the supporting prices.

Demange et al. [4] consider the assignment problem with multiple items in the unit demand case. They propose an ascending auction that implements the VCG outcome. The auctioneer collects the "demand sets" (set of all items giving a bidder the maximum payoff) at each iteration, and computes the minimally over-demanded set of items. The prices of these over-demanded items are progressively increased until the competitive equilibrium is reached–each bidder demands only one item, and each item is demanded by no more than one bidder–and the auction terminates. In this work, the authors assume that all prices and valuations are integers; we do not restrict the valuations (and prices) to be integral in our mechanism. Also, for their exact mechanism to work, it is necessary for the bidders to be very precise in their responses to the changing prices. As a consequence of the valuations being integral numbers of units, a bidder should report *all* the items that maximize her surplus, and cannot switch between items from one iteration to another. For this reason, the authors themselves admit that this might not be a very practical mechanism to implement in realistic situations. Our iterative mechanism needs each bidder to report only a couplet, and requires the auctioneer to keep only one triplet about each bidder.

Demange et al. [4] also propose a more flexible approximate mechanism that achieves the equilibrium prices within some error bounds. Unfortunately, the outcome of this mechanism depends on the order of placing bids. Our mechanism does not depend on the order of bid elicitation. In our simulations, we randomize the order in which the auctioneer elicits bids from the bidders in every round.

Wellman et al. [9] consider some general classes of scheduling problems and investigate the existence and properties of equilibrium solutions with distributed protocols. They analyze an ascending auction protocol under various settings, including single unit resource allocation problems–a class of problems that contains our model. Wellman et al. derive error bounds on the sub-optimality of the solution that arises in the ascending auction.

One essential difference between the ascending auctions considered in the literature and our auction is in the application of the mechanism to a real world scenario. We consider the problem in a particular case where sellers need to distribute a limited supply of goods spread across different delivery days. We believe such situations are common in the real world. Pricing the goods consistently for products delivered on the same day is a reasonable and fair method that is likely to be acceptable to bidders. The existing ascending auctions do not explicitly deal with such issues.

7 Conclusions

We have designed a simple auction mechanism for efficient allocation and pricing of a limited supply of highly demanded, homogeneous goods, with different delivery days. An important application of this work is the initial release of popular products which the manufacturer can supply in limited quantities for some period of time. The bidders that participate in this auction can be categorized as self-interested, with private valuations for the goods. The iterative nature of our auction addresses the problem of computing the efficient outcome with incomplete information revelation by bidders. Under certain assumptions on the valuations, and unit demand, we have shown that our mechanism computes the outcome of the generalized Vickrey auction and is incentive compatible. Yet even when these assumptions are violated, the auction is simple enough to implement and use with a large number of bidders that we believe it can be applied to good effect.

The incentive compatibility feature of our auction means that the bidders do not have to think strategically. At the same time, we greatly reduced the *valuation problem*, that is, the bidders' task of determining their private valuations for the items; in our simulations, bidders needed to reveal only about 15% of their valuation function, and by implication did not need to compute exact values on all days in order to eliminate some from consideration.

Another feature of our auction is that bidders can join and/or leave the auction anytime during the process. This is a desirable feature for the practical situations where we envisage our auction being used. From the viewpoint of the auctioneer, while the entire bid history of each bidder can be stored, our mechanism does not require it. Only the most recent offer from each bidder is stored by the auctioneer, and it is enough information to compute the efficient allocation and the VCG prices. This is also a desirable feature when implementing large auctions.

Our simulations have produced encouraging results about the convergence of the mechanism to the equilibrium, and the number of iterations necessary to reach equilibrium. Analytically proving the convergence of the mechanism is the next step in validating our claims. The existing literature in this area points only to ascending auctions that implement the primal-dual algorithm. Our mechanism does not have an ascending auction design, and trying to further interpret the dynamics of our auction should yield more fruitful research.

There are other aspects of auction analysis that we have not yet addressed, such as potential collusion by bidders. Many existing auction mechanisms fail in the face of collusive behavior by bidders, including the Vickrey second price auction, and it would be interesting to analyze the robustness of our design against bidder misbehavior.

Acknowledgments

This research was funded by NSF CAREER award 0092591 and the E-Commerce program NC State. We wish to thank the members of the Intelligent Commerce Research Group at NCSU, and the members of the first author's MS thesis committee, Dr. Munindar Singh and Dr. Michael Young, for their comments.

References

[1] S. Bikhchandani, S. de Vries, J. Schummer, and R. V. Vohra. Linear programming and vickrey auctions, 2001.

[2] E. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.

[3] C. Crouch and T. Mainelli. Playstation 2 arrives–but it's going fast. *PCWorld.com*, October 26 2000.

[4] G. Demange, D. Gale, and M. Sotomayer. Multi-item auctions. *Journal of Political Economy*, 94(4):863–872, 1986.

[5] T. Groves. Incentives in teams. *Econometrica*, 61:617–631, 1973.

[6] H. B. Leonard. Elicitation of honest preferences for the assignment of individuals to positions. *Journal of Political Economy*, 91:1–36, 1983.

[7] L. Shapley and M. Shubik. The assignment game I: The core. *International Journal of Game Theory*, 1(2):111–130, 1972.

[8] W. Vickrey. Counterspeculation, auctions and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

[9] M. P. Wellman, W. E. Walsh, P. R. Wurman, and J. MacKie-Mason. Auction protocols for decentralized scheduling. *Games and Economic Behavior*, 35(1–2):271–303, 2001.