AkBA: A Progressive, Anonymous-Price Combinatorial Auction

Peter R. Wurman
Computer Science
North Carolina State University
Raleigh, NC 27695-7535 USA
wurman@csc.ncsu.edu

Michael P. Wellman
Computer Science and Engineering
University of Michigan
Ann Arbor, MI 48109-2110 USA
wellman@umich.edu

ABSTRACT

The allocation of discrete, complementary resources is a fundamental problem in economics and of direct interest to e-commerce applications. Combinatorial auctions account for complementarities by optimizing over offers expressed in terms of bundles. Progressive versions of combinatorial auctions alleviate the burden on bidders of expressing offers for all bundles of interest by providing interim feedback based on partial sets of bids. Feedback in terms of hypothetical prices is particularly useful, as it directs bidders toward those bundles potentially yielding the greatest surplus. For a general class of discrete resource allocation problems with free disposal, we establish by construction the existence of competitive equilibrium prices on bundles that support the efficient allocation. We introduce AkBA, a family of progressive auctions that use these equilibrium bundle prices. We examine a particular instance of the family, called A1BA, and present some empirical data on its performance.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous

Keywords

Combinatorial auctions, equilibrium prices, progressive auctions.

1. INTRODUCTION

In many potential e-commerce applications, agents have complementary preferences for items in the marketplace. Consider a factory that builds configurable products, such as computers. Customers have different valuations for different computer configurations—some are willing to pay more for a larger disk, a better graphics card, or a faster CPU. The factory has a limited supply of components, and would like to custom build the set of computers that generates the most total revenue. This example is an instance of the general allocation problem characterized by heterogeneous, discrete resources (e.g., components) and complementarities in agent preferences (e.g., over complete computer configurations).

Discrete resource allocation has been the focus of auction theory since Vickrey's 1961 article presenting the second-price auction [25]. The allocation of single items, or multiple homogeneous items, is well understood [6, 14, 17], however a general solution to the allocation of heterogeneous items has proved elusive.

The principal difficulty lies in the fact that efficient allocations may not be supportable by price equilibria. A simple example presented by McAfee and McMillan [15], reproduced in Table 1, illustrates the point. One agent values the pair of items at 3 and gets no benefit from the items singly. The second agent values either item at 2, but gets no added benefit from getting both items. The *socially efficient* solution assigns both items to agent 1. For any assignment that allocates items to agent 2, both agents can be made better off by an exchange of money from agent 1 with the items from agent 2. Yet there exist no equilibrium prices supporting the socially efficient (or any other) solution. In order to exclude the second agent from the allocation, the price of each item individually must be greater than or equal to 2. However, these prices put the cost of the pair at 4—above the first agent's valuation.

	A	B	AB
Agent 1	0	0	3*
Agent 2	2	2	2

Table 1: An example without equilibrium prices for individual items. Here and throughout, asterisks mark the socially efficient allocation.

In the example in Table 1, agent 1's preferences are *superadditive*, that is, it gets more value from the pair of items than simply the sum of the values it gets from each item individually. Superadditive preferences can lead to *complementarity*—an increase in price of one item associated with a decrease in demand of the other. It is well-known that complementarities can prevent the existence of equilibrium in price systems.

Recently, both economists and computer scientists have explored mechanisms to deal with complementarities. *Combinatorial auctions* allow agents to submit bids that directly express offers to buy a combination of items. This helps alleviate the *exposure problem* in which an agent desires a

EC'00, October 17-20, 2000, Minneapolis, Minnesota.

bundle but can make offers only for individual elements. In order to avoid getting stuck with only a subset of its desires, the agent may behave in an exceedingly cautious manner. However, combinatorial auctions often suffer from the *free rider* problem: two agents together value disjoint subsets of a bundle more than a third values the entire bundle, but each bids little hoping the other will shoulder more of the burden of displacing the third agent.

To characterize some distinguishing features of combinatorial auctions, we make use of a general parametrization of the auction design space [26]. The parameters determine the auction's behavior for each of its three main tasks: (i) admitting bids, (ii) (optionally) generating intermediate information—often in the form of hypothetical *price quotes*, and (iii) determining an allocation, referred to as *clearing* the auction.

In a *progressive* combinatorial auction (PCA), the bidding rules require that each iteration's bids be an improvement (in a well defined sense) over the previous. This promotes steady progress toward a final result, eliminating the possibility of oscillation or deadlock. The process ends when some termination criterion is reached, such as passage of a round with no participant submitting a new bid.

The task of clearing involves two steps. First, the auction must identify the bids that will be part of the transaction set. This step is referred to as *winner determination*. In general, solving the winner determination problem in a combinatorial auction is NP-complete, although tractable special cases exist [21]. Fujishima et al. [8] and Sandholm [22] have developed specialized algorithms for combinatorial winner determination that combine clever pruning techniques with depth-first search. These algorithms have performed well on problems with thousands of bids. More recently, several authors [1, 5, 18] have explored the application of standard linear and integer programming techniques to the winner determination problem. Importantly, these investigations illustrate how very dependent computational results are on assumptions about the distributions of bids.

Given the set of winning bids, the second step of auction clearing determines the payments associated with the transactions. We call this step *payment determination*. We use the term "payment" because it is more inclusive than the term "price". Typically, item j's price, p_j, defines the per-unit measure of the rate of exchange. Thus, an agent's payment is simply the sum of the prices of each item it receives. *Nonlinear pricing* permits prices to vary with quantity. In the case of heterogenous items, this corresponds to permitting $p_{jk} \neq p_j + p_k$, where p_{jk} is the *bundle price* associated with purchasing the combination. Prices are also classically *anonymous*, that is, they do not depend upon an individual's identity—all participants can buy or sell the item at the posted price. The exception is *discriminatory pricing*, which allows prices to vary depending upon some attribute of the agent, such as its identity or its current or historical bids.

The common benchmark against which combinatorial mechanisms are compared is the Generalized Vickrey Auction (GVA) [13]. The GVA can be viewed as a sealed-bid, combinatorial auction that computes discriminatory payments.[1]

[1]The monetary transactions determined by the GVA are referred to as payments because they are computed per agent, and not per item, or combination of items. The GVA does not specify the costs of unassigned items or bundles.

It is attractive because it is incentive compatible, individually rational, and efficient, but these desirable properties come at a significant computational cost for both the auctioneer and the participants [2]. The auctioneer must solve the winner determination problem to determine the optimal allocation of items, and then solve a winner determination problem once for each agent to determine payments. In addition, the GVA requires that each agent submit its utility function which, in the worst case, specifies $2^n - 1$ values (where n is the number of items), each of which may be computationally expensive for the agent to determine.

One motivation for investigating PCAs is the computational burden the GVA places on agents. Because a PCA iterates, an agent need not prepare bids for every combination of items of potential interest; rather, it can wait to see based on the feedback which combinations are likely to be relevant [19]. Since combinatorial auctions, by definition, involve a combinatorial number of potential bundles, this can lead to a tremendous saving of effort for the agents. In addition, the GVA requires that participants place a great deal of trust in the auctioneer if they are going to completely reveal their utility function. In a PCA, the participants need reveal only that portion of their utility functions that they deem relevant to the allocation.

The PCA we propose here achieves these general benefits of progressive mechanisms by using an anonymous, nonlinear pricing scheme. There are several potential benefits to anonymous pricing. First, when prices are determined based on identity (or bidding behavior), agents may have a further incentive to bid strategically to misrepresent this information. Second, we expect that participants in combinatorial auctions will desire an (admittedly incompletely-defined) *fairness* in the payment scheme. In particular, participants will presume that if another buyer purchased a bundle of items, $\boldsymbol{b}$, for a price, say $\pi_{\boldsymbol{b}}$, during the auction, then all agents should have had the opportunity to purchase $\boldsymbol{b}$ for $\pi_{\boldsymbol{b}}$. Though weaker than the *no regret* property [3]—which says that, regardless of whether prices are anonymous or discriminatory, an agent would not change its strategy after observing the other agents' bids—this property does say that an agent would not want to purchase someone else's bundle once it learns the final prices.

In the next section, we present the formal model of the allocation problem and define mechanism properties of interest. Section 3 shows how the anonymous, nonlinear prices are generated and proves that these prices have the desired properties. Section 4 introduces a new progressive combinatorial auction based on this price setting method and presents the results of our preliminary experiments with the mechanism. We compare our mechanism to other combinatorial auctions and discuss other related work in Section 5.

2. MODEL

We consider the task of allocating a set of n heterogeneous items $\mathcal{J}$ to a set of m agents $\mathcal{I}$. The agents are indexed by i and the items by j. There are $2^n - 1$ different combinations of items (excluding the empty set). Let $\boldsymbol{b} \in \{0,1\}^n$ where $b^j = 1$ implies that item j is an element of the bundle $\boldsymbol{b}$. For two bundles, $\boldsymbol{b}$ and $\boldsymbol{c}$, we use the superset notation $\boldsymbol{b} \supset \boldsymbol{c}$ to indicate that for all j, $b^j \geq c^j$. The set of all possible bundles, denoted $\mathcal{B}$, forms a finite lattice.

The value that an agent derives from a bundle is given by $v_i(\boldsymbol{b})$. We assume that agents have quasilinear utility, are risk neutral, and have no budget constraint. Further, we invoke free disposal, which allows us to assume valuations increase monotonically as items are added to a bundle. Formally, $\boldsymbol{b} \supset \boldsymbol{c}$ implies $v_i(\boldsymbol{b}) \geq v_i(\boldsymbol{c})$.

A *solution* to the allocation problem is a feasible assignment of bundles to agents. Let $x_{i\boldsymbol{b}} \in \{0,1\}$ take the unit value only when $\boldsymbol{b}$ is assigned to i. The value of the socially efficient allocation is given by

$$\begin{aligned} \max \quad & \sum_i \sum_{\boldsymbol{b}} v_i(\boldsymbol{b}) x_{i\boldsymbol{b}} & (1) \\ \text{s.t.} \quad & \sum_{\boldsymbol{b}} x_{i\boldsymbol{b}} \leq 1, \; i = 1, \ldots, m, \\ & \sum_i \sum_{\boldsymbol{b}} b^j x_{i\boldsymbol{b}} \leq 1, \; j = 1, \ldots, n, \\ & x_{i\boldsymbol{b}} \in \{0,1\}. \end{aligned}$$

The first constraint implies that each agent receives at most one bundle. The second ensures that no item is allocated more than once.

Let $\boldsymbol{f}^* \equiv \{x^*_{i\boldsymbol{b}}\}$ be a solution to (1). The value of the solution is simply

$$V(\boldsymbol{f}^*) = \sum_i \sum_{\boldsymbol{b}} v_i(\boldsymbol{b}) x^*_{i\boldsymbol{b}}.$$

Let $\boldsymbol{f}^*_i$ be agent i's allocation in $\boldsymbol{f}^*$.

Let $\boldsymbol{p}$ be a vector of prices, one for each element of $\mathcal{J}$. In a *price equilibrium*, agents behave as if prices are determined exogenously and are unaffected by the agent's own actions. Under quasilinear preferences, a solution, $\boldsymbol{f}^*$, and price vector, $\boldsymbol{p}$, form a price equilibrium if, for all i,

$$v_i(\boldsymbol{f}^*_i) - \boldsymbol{p} \cdot \boldsymbol{f}^*_i = \max_{\boldsymbol{b} \in \mathcal{B}} [v_i(\boldsymbol{b}) - \boldsymbol{p} \cdot \boldsymbol{b}]. \tag{2}$$

If (2) holds for a price vector $\boldsymbol{p}$, we say $\boldsymbol{p}$ *supports* the equilibrium. However, as the example in Table 1 illustrates, price equilibria do not always exist in the problem domain under consideration. Thus we relax the definition of a price equilibrium to account for nonlinear prices.

A complete specification of nonlinear prices associates a price, $\pi_{\boldsymbol{b}}$, with every bundle, $\boldsymbol{b}$. The aggregation of these prices forms the lattice $\boldsymbol{\pi}$. Under quasilinear preferences, the solution $\boldsymbol{f}^*$, and price lattice $\boldsymbol{\pi}$, form a *nonlinear-price equilibrium* if, for all i,

$$v_i(\boldsymbol{f}^*_i) - \pi_{\boldsymbol{f}^*_i} = \max_{\boldsymbol{b} \in \mathcal{B}} [v_i(\boldsymbol{b}) - \pi_{\boldsymbol{b}}]. \tag{3}$$

If (3) holds for lattice $\boldsymbol{\pi}$, we say $\boldsymbol{\pi}$ *supports* the equilibrium.

We further require that the price of a bundle be at least as great as any of its subsets (a natural consequence of free disposal). Each agent faces the same price lattice, that is, prices are anonymous.

To maintain nonlinear pricing, it may be necessary to have some regulator enforce the allocation of one bundle per agent. Otherwise, if $\pi_{\boldsymbol{b} \cup \boldsymbol{c}} > \pi_{\boldsymbol{b}} + \pi_{\boldsymbol{c}}$, an agent could obtain $\boldsymbol{b} \cup \boldsymbol{c}$ at advantageous terms by purchasing $\boldsymbol{b}$ and $\boldsymbol{c}$ separately and assembling the bundle. Similarly, if $\pi_{\boldsymbol{b} \cup \boldsymbol{c}} < \pi_{\boldsymbol{b}} + \pi_{\boldsymbol{c}}$, two agents may have an incentive to collude in order to purchase the combined set at a discount, and reallocate the items later.

3. EQUILIBRIUM PRICES

We now consider the existence of nonlinear-price equilibria that support the efficient allocation when agents have valuations that are monotone on the finite lattice $\mathcal{B}$. We show that such equilibrium bundle prices *always* exist by constructing them from the agents' valuations.

3.1 Construction

The construction proceeds in three steps. First, find the efficient allocation. Next, compute prices on the bundles that are allocated in $\boldsymbol{f}^*$. Finally, compute prices on the unallocated bundles.

Step 1. Solve (1) and compute $\boldsymbol{f}^*$ and $V(\boldsymbol{f}^*)$. For example, one can employ any of the aforementioned winner-determination algorithms for this step.

Step 2. Let $\mathcal{I}_\ominus$ be the set of agents who receive items in $\boldsymbol{f}^*$, that is, $\mathcal{I}_\ominus = \{i \mid \boldsymbol{f}^*_i \neq \emptyset\}$. The set $\mathcal{I}_{\neg\ominus}$ denotes the agents who receive nothing, $\mathcal{I}_{\neg\ominus} = \mathcal{I} \setminus \mathcal{I}_\ominus$. Let $\mathcal{B}_\ominus$ be the subset of bundles that are assigned in $\boldsymbol{f}^*$, and $\mathcal{B}_\emptyset$ be the unassigned bundles. For each agent $i \in \mathcal{I}_{\neg\ominus}$, introduce a null item ϕ_i to represent the agent's null allocation. Let $\Phi \equiv \{\phi_i | i \in \mathcal{I}_{\neg\ominus}\}$. Every agent has a valuation of zero for every element of Φ.

Let $G = \mathcal{B}_\ominus \cup \Phi$, and $g \in G$. G, along with the corresponding agent valuations, describes an assignment problem [24]. We refer to it as the assignment subproblem because in the process of formulating it, we have discarded information about the original allocation problem. Specifically, we have omitted all of the bundles that are not part of the solution. Note also, that we already have the solution to the subproblem, $\boldsymbol{f}^*$, from Step 1.

We now compute π_g, for all g, that support the solution to the assignment subproblem. To accomplish this, we employ the dual program used by Leonard [12] to compute minimal prices for assignment problems. Let $\mathrm{LP}_{\mathrm{lower}}$ be the following linear program:

$$\begin{aligned} \min \quad & \sum_g \pi_g \\ \text{s.t.} \quad & s_i + \pi_g \geq v_i(g), \forall i, g, \\ & s_i, \pi_g \geq 0, \\ & \sum_i s_i + \sum_g \pi_g = V(\boldsymbol{f}^*). \end{aligned}$$

The last constraint ensures that the first constraint is satisfied at equality for the optimal assignment.

The s_i term represents the surplus achieved by agent i. $\mathrm{LP}_{\mathrm{lower}}$ maximizes each agent's surplus within the range of equilibrium prices that support the optimal solution to the assignment subproblem. Note that the introduction of the items in Φ is simply a trick to include the agents in $\mathcal{I}_{\neg\ominus}$ in the assignment subproblem. In the solution, $s_i = 0$ and $\pi_{\phi_i} = 0$ for all $i \in \mathcal{I}_{\neg\ominus}$.

$\mathrm{LP}_{\mathrm{lower}}$ has a complementary program, $\mathrm{LP}_{\mathrm{upper}}$, which computes upper bound prices:

$$\begin{aligned} \min \quad & \sum_i s_i \\ \text{s.t.} \quad & s_i + \pi_g \geq v_i(g), \forall i, g, \\ & s_i, \pi_g \geq 0, \\ & \sum_i s_i + \sum_g \pi_g = V(\boldsymbol{f}^*). \end{aligned}$$

Clearly the price vector produced in finding a solution to either $\mathrm{LP}_{\mathrm{lower}}$ or $\mathrm{LP}_{\mathrm{upper}}$ is a price equilibrium for the assignment subproblem—the first constraint in both formulations requires that an agent cannot gain any more surplus from another assigned bundle than it receives from the one allocated to it.

Step 3. Given a solution to either $\mathrm{LP}_{\mathrm{lower}}$ or $\mathrm{LP}_{\mathrm{upper}}$, the next step is to set prices on the bundles in $\mathcal{B}_\emptyset$. This can be done in a straightforward manner using the surpluses calculated in Step 2. For all $\boldsymbol{b} \in \mathcal{B}_\emptyset$,

$$\pi_{\boldsymbol{b}} = \max_i [v_i(\boldsymbol{b}) - s_i]. \tag{4}$$

3.2 Properties

THEOREM 1. *The bundle prices, $\underline{\boldsymbol{\pi}}^*$, computed by LP_{lower} and (4), support the efficient allocation.*

PROOF. The construction begins by identifying an optimal allocation, $\boldsymbol{f}^*$. The question remains whether the constructed bundle prices, $\underline{\boldsymbol{\pi}}^*$, *support* $\boldsymbol{f}^*$. A solution to $\mathrm{LP}_{\mathrm{lower}}$ has the property that $\boldsymbol{f}_i^*$ maximizes i's surplus among the bundles in G. Koopmans and Beckmann (1957) show that an integer solution to the assignment problem always exists. By the Duality Theorem of Linear programming, we can support this integer solution with prices. For all other bundles, we set the price such that $\underline{\pi}_{\boldsymbol{b}}^* \geq v_i(\boldsymbol{b}) - s_i$ (4). Therefore, no bundle provides more surplus to i than its assignment. □

THEOREM 2. *The bundle prices, $\overline{\boldsymbol{\pi}}^*$, computed by LP_{upper} and (4), support the efficient allocation.*

PROOF. Same as for Theorem 1 with $\mathrm{LP}_{\mathrm{upper}}$ in place of $\mathrm{LP}_{\mathrm{lower}}$. □

These bundle prices satisfy the same monotonicity constraint imposed on valuations.

THEOREM 3. *The price lattice computed by LP_{lower} and (4), or by LP_{upper} and (4), is monotone.*

PROOF. Condition (4) implies that, for all i, $\boldsymbol{b} \in \mathcal{B}_\emptyset$,

$$\pi_{\boldsymbol{b}} \geq v_i(\boldsymbol{b}) - s_i,$$

which can be written

$$s_i + \pi_{\boldsymbol{b}} \geq v_i(\boldsymbol{b}). \tag{5}$$

The first constraint in $\mathrm{LP}_{\mathrm{lower}}$ ($\mathrm{LP}_{\mathrm{upper}}$) imposes the same condition on all $\boldsymbol{b} \in \mathcal{B}_\ominus$.

Therefore, (5) holds for all i and $\boldsymbol{b}$. Let h be the agent that maximizes $\pi_{\boldsymbol{b}}$. That is, $h = \arg\max_i [v_i(\boldsymbol{b}) - s_i]$. Thus, $s_h + \pi_{\boldsymbol{b}} = v_h(\boldsymbol{b})$. When valuations are monotone, $\boldsymbol{c} \supset \boldsymbol{b}$ implies that $v_h(\boldsymbol{c}) \geq v_h(\boldsymbol{b})$. Therefore

$$s_h + \pi_{\boldsymbol{c}} \geq v_h(\boldsymbol{c}) \geq v_h(\boldsymbol{b}) = s_h + \pi_{\boldsymbol{b}},$$

which reduces to

$$\pi_{\boldsymbol{c}} \geq \pi_{\boldsymbol{b}}.$$

□

	A	B	C	AB	AC	BC	ABC
Agent 1	6	6	5*	10	7	8	12
Agent 2	3	3	2	8*	5	6	11
Agent 3	4	2	1	7	6	5	10

Table 2: An example with three agents.

Let $\overline{\boldsymbol{\pi}}^*$ be a lattice of prices calculated by $\mathrm{LP}_{\mathrm{upper}}$ and (4). Similarly, $\underline{\boldsymbol{\pi}}^*$ is the price lattice resulting from solving $\mathrm{LP}_{\mathrm{lower}}$ and applying (4). We now consider a range of equilibrium bundle prices that support the optimal allocation.

THEOREM 4. *For all $k \in [0,1]$, $k\overline{\boldsymbol{\pi}}^* + (1-k)\underline{\boldsymbol{\pi}}^*$ is a nonlinear-price equilibrium.*

PROOF. Let $\boldsymbol{b}$ be (one of) agent i's most preferred bundles at $\overline{\boldsymbol{\pi}}^*$, and $\boldsymbol{c}$ be some other bundle. Because $\overline{\boldsymbol{\pi}}^*$ and $\underline{\boldsymbol{\pi}}^*$ both support the same efficient allocation, i must not prefer $\boldsymbol{c}$ to $\boldsymbol{b}$ at $\underline{\boldsymbol{\pi}}^*$. Formally,

$$v_i(\boldsymbol{b}) - \overline{\pi}_{\boldsymbol{b}}^* \geq v_i(\boldsymbol{c}) - \overline{\pi}_{\boldsymbol{c}}^*,$$

and

$$v_i(\boldsymbol{b}) - \underline{\pi}_{\boldsymbol{b}}^* \geq v_i(\boldsymbol{c}) - \underline{\pi}_{\boldsymbol{c}}^*.$$

In both cases, the relation is invariant to positive scalar transformations. Thus, for $k \in [0,1]$,

$$k[v_i(\boldsymbol{b}) - \overline{\pi}_{\boldsymbol{b}}^*] \geq k[v_i(\boldsymbol{c}) - \overline{\pi}_{\boldsymbol{c}}^*],$$

and

$$(1-k)[v_i(\boldsymbol{b}) - \underline{\pi}_{\boldsymbol{b}}^*] \geq (1-k)[v_i(\boldsymbol{c}) - \underline{\pi}_{\boldsymbol{c}}^*].$$

Adding the two equations and simplifying gives

$$v_i(\boldsymbol{b}) - [k\overline{\pi}_{\boldsymbol{b}}^* + (1-k)\underline{\pi}_{\boldsymbol{b}}^*] \geq v_i(\boldsymbol{c}) - [k\overline{\pi}_{\boldsymbol{c}}^* + (1-k)\underline{\pi}_{\boldsymbol{c}}^*].$$

□

We refer to auctions that set prices in accordance with Theorem 4 as *k-bundle auctions*. The k parameter here is directly analogous to the parameter used in the k-double auction [23]. $\overline{\boldsymbol{\pi}}^*$ and $\underline{\boldsymbol{\pi}}^*$ bound the range of anonymous bundle prices for which supply equals demand. Reducing any price in $\underline{\boldsymbol{\pi}}^*$ would create excess demand for some of the items. There is a slightly weaker analogy for the upper bound. We cannot raise the price of any $\boldsymbol{b} \in \mathcal{B}_\ominus$ without disequilibrating supply and demand. However, in some cases we can increase the prices of bundles in $\mathcal{B}_\emptyset$ without adverse affects.

	AB	C	ϕ_3
Agent 1	10	5	0
Agent 2	8	2	0
Agent 3	7	1	0

Table 3: The assignment subproblem.

	A	B	C	AB	AC	BC	ABC
$\overline{\pi}^*$	4	4	3	8	6	6	11
$\underline{\pi}^*$	4	2	1	7	6	5	10

Table 4: Equilibrium price lattices.

3.3 Example

Consider the example in Table 2. The efficient allocation is to assign AB to agent 2 and C to agent 1. Agent 3 gets nothing. This allocation has a social welfare of 13.

In order to construct equilibrium bundle prices, we introduce the null item ϕ_3. Now, solve the linear programs $\mathrm{LP_{lower}}$ and $\mathrm{LP_{upper}}$ for the assignment subproblem in Table 3.

The solution to $\mathrm{LP_{lower}}$ for this problem is $\pi_{AB} = 7$, $\pi_C = 1$, and of course $\pi_{\phi_3} = 0$. This leaves $s_1 = 4$, $s_2 = 1$, and $s_3 = 0$. The final upper and lower equilibrium price lattices are given in Table 4.

Figure 1 shows the binding linear constraints for this example projected into the space $\pi_{AB} \times \pi_C$. The constraints pictured correspond to:

$$\pi_{AB} \geq 7 \tag{6}$$

$$\pi_{AB} \geq \pi_C + 5 \tag{7}$$

$$\pi_{AB} \leq \pi_C + 6 \tag{8}$$

$$\pi_{AB} \leq 8 \tag{9}$$

The first constraint ensures that agent 3 doesn't want to buy AB, while (7) and (8) ensure that agent 1 prefers C to AB, and agent 2 prefers AB to C, respectively. The last constraint ensures that the price of AB does not exceed agent 2's valuation. The white region is the space of equilibrium prices, and points on the dashed line between $\overline{\pi}^*$ and $\underline{\pi}^*$ are k-bundle prices.

3.4 Discussion

To our knowledge, the only other study of nonlinear-price equilibria in combinatorial auctions is by Bikhchandani and Ostroy [4]. B&O analyzed the competitive equilibrium properties of the *package assignment* model. In their formulation, buyers and sellers exchange packages of items with the restriction that buyers are allowed to request only one package from each seller, and sellers are allowed to earmark only one package for each buyer.

B&O show that for the single-seller model, third-order Walrasian equilibrium exist, but second-order equilibrium may not. In their terminology, third-order equilibrium prices allow a separate price vector for each buyer (i.e., are discriminatory), whereas second-order pricing has one nonlinear vector presented to all parties and is equivalent to what we call anonymous, nonlinear prices. An example from their paper is shown in Table 5.

Equilibrium in the package assignment model requires that the seller choose to sell the combination of bundles that maximizes its revenue. In this example, in order for the seller to

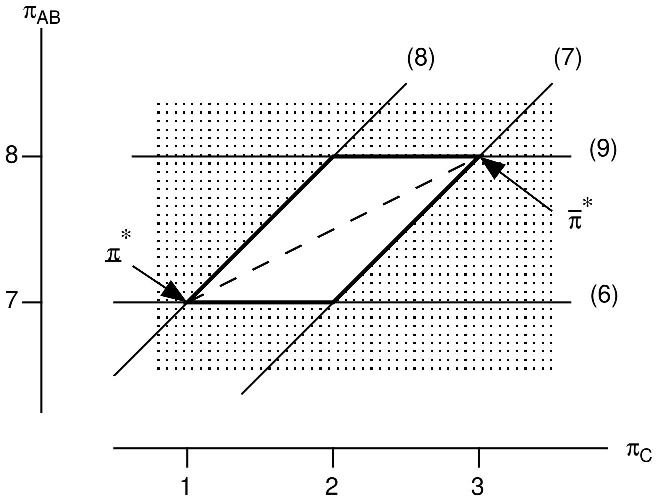

Figure 1: The space of possible equilibrium prices for the example in Table 2. The dashed line indicates the range of k-bundle prices.

be maximizing its revenue with respect to the prices, separate price vectors are needed for each buyer, labeled π^1 and π^2 in Table 5.

B&O's results seem at odds with ours. The discrepancy is explained by a difference in the allowable actions in the two models. In contrast to B&O's model, our model does not allow the seller to earmark bundles for buyers. Instead, we assume the seller has consigned all the items to the auctioneer, who then redistributes them to the buyers. For the example in Table 5, $(3.4, 3.6, 3.4, 6.65, 6.4, 6.9, 8.4)$ is an anonymous price lattice, computed using our method, that supports the assignment where agent 1 receives AB and agent 2 receives C. We have fixed the price for C at 3.4 to make the prices comparable to B&O's. Notice that the seller receives less revenue (10.05 versus 10.3) under our scheme than B&O's. In fact, were the seller in our scheme free to choose how to sell its bundles, it would prefer to sell AC and B, generating an apparent revenue of 10.5. In some sense, this loss of revenue to and loss of control by the seller is the cost of producing anonymous prices.

The construction in Section 3.1 proves that any optimal allocation is supportable by some anonymous, nonlinear-price equilibrium. However, it should also be noted that by relaxing the pricing scheme from linear to nonlinear, we admit equilibria that support nonoptimal allocations. For example, the price vector $(2, 2, 2.5)$ supports a competitive equilibrium for both the problem in Table 1 and the problem in Table 6. However, in the latter case the allocation it supports is inefficient—agent 1 gets both items. For this example, the price vector $(2, 2, 3)$ supports the efficient allocation.[2] Clearly, one of the challenges in developing an efficient allocation mechanism will be avoiding these inefficient equilibria.

[2]We present the price vector that is computed by our algorithm, but in this problem, the efficient solution actually has an equilibrium that is supported by linear prices.

	A	B	C	AB	BC	AC	ABC
Agent 1	4	4	4.25	7.5*	7	7	9
Agent 2	4	4.25	4*	7	7	7.5	9
π^1	3.5	3.4	3.75	6.9	6.5	6.5	9
π^2	3.5	3.75	3.4	6.5	6.5	6.9	9

Table 5: An example from Bikhchandani & Ostroy. The top two rows specify the agents' valuations. The bottom two rows display third-order equilibrium prices. π^1 and π^2 are the price vectors that govern trades with agent 1 and 2, respectively.

	A	B	AB
Agent 1	0	0	3
Agent 2	2*	2	2
Agent 3	2	2*	2

Table 6: An example with multiple equilibrium bundle price vectors.

4. ASCENDING k-BUNDLE AUCTIONS

Although we have developed the price setting policy under the assumption that the mechanism knows the agents' valuations, the procedures derived can be applied to bids and used in the quote or clearing steps of an iterative auction. Thus, we can combine the k-bundle price policy with a wide variety of other auction rules [26], giving potentially hundreds of new auction types to explore. In this section, we present the class of Ascending k-Bundle Auctions (abbreviated AkBA), and examine one particular member where $k = 1$ (abbreviated A1BA).

Without loss of generality, we assume that agent i's bid, r_i, is expressed as a collection of mutually exclusive offers on bundles of the form $r_i(\boldsymbol{b})$, where $r_i(\boldsymbol{b}) \in \mathcal{R}_+$.[3] $r_i(\boldsymbol{b})$ is not necessarily equal to $v_i(\boldsymbol{b})$. An agent can bid on any or all bundles, but will be assigned at most one bundle.

4.1 AkBA Rules

AkBA accepts bids in the form described above, and computes a tentative optimal allocation and a quote, in the form of a price lattice, by using the k-bundle price procedure. AkBA applies the ascending rule and the beat-the-quote bidding rules [26]. The combination of these rules requires that agent i's new bid, $\hat{r}_i$, satisfies $\hat{r}_i(\boldsymbol{b}) \geq r_i(\boldsymbol{b})$ for all $\boldsymbol{b}$, and $\hat{r}_i(\boldsymbol{b}) \geq \pi_{\boldsymbol{b}} + \delta$ for at least one $\boldsymbol{b}$, where δ is the minimum bid increment. The auction clears when a period of bidding inactivity concludes.

As a practical matter, rather than having an agent specify a valuation for every member of the lattice in each bid, the auction stores the agent's previous offers (beginning at zero) and the agent's message is treated as an update. We assume that the auctioneer interprets these updates in a manner consistent with the assumption of monotone valuations; if agent i increases its offer on $\boldsymbol{b}$, the auction ensures that i's offer is raised on all $\boldsymbol{c} \supset \boldsymbol{b}$ for which the agent's previous offer was less than its new offer on $\boldsymbol{b}$. Formally, for all $\boldsymbol{c} \supset \boldsymbol{b}$,

$$\hat{r}_i(\boldsymbol{c}) = \max(r_i(\boldsymbol{c}), \hat{r}_i(\boldsymbol{b})).$$

[3] This is the standard XOR format described by others, which is fully expressive [18]. In particular applications, more concise bidding languages can be used.

	A	B	AB
Agent 1	5*	3	7
Agent 2	2	3*	6

Table 7: An example in which prices would decrease if agent 1 increased its offer on B.

Note that offers cannot be withdrawn even if they are not part of the current tentative best allocation.

The auction begins with a quote in which each price on the lattice is zero. Thereafter, it calculates a new price lattice each time a bid is admitted.[4] With $k = 1$, the quote is the price lattice $\boldsymbol{\pi}^*$ computed from the current set of bids. Because of the manner in which the prices are determined, the agent knows it is winning the bundle that maximizes $r_i(\boldsymbol{b}) - \pi_{\boldsymbol{b}}$. If more than one bundle satisfies this condition, the agent may not be able to tell from among these which it is winning. To remedy this, the auction informs each agent which bundle, if any, it is tentatively winning. With this enhancement, the quote becomes *separating*, that is, it accurately informs all agents of the contents and associated payment of their assignment in the tentative allocation. Note that the auction needs to announce prices on only those bundles that have received offers, since the agents can easily perform the inference to determine minimal prices for the rest of the bundles.

The auction design is directly applicable to situations where bids are restricted in order to admit polynomial time algorithms for computing $\boldsymbol{f}^*$ [21]. It is equally well-defined whether the bids are sparse or dense. In addition, we can straightforwardly substitute approximation algorithms into step 1 of the price setting procedure. In such cases, the auction can use the better of the previous best allocation and the newly computed approximate solution to the winner determination problem. This is a promising area for future research as it may lead to high performing auctions that are not constrained by the complexity of solving optimally the winner determination problem.

We classify AkBA as a progressive auction because through iteration it makes progress toward termination. The bidding rules require that each agent's bid ascend, that is, be an improvement over its previous bid. However, this does not necessarily mean that prices on the lattice monotonically increase across iterations. Table 7 illustrates an example of two agents bidding on three bundles. Based on these bids, A1BA would determine that the best allocation involves giving A to agent 1 and B to agent 2, and would compute a price lattice $(5, 3, 7)$. Suppose that agent 1 increased its bid on B to 4. The optimal allocation would remain the same, but the supporting price lattice would now be $(4, 3, 6)$. The auction decreases the price of item A so that agent 1 would still (weakly) prefer to buy it over any other bundle.

If agent 2 did not raise any of its offers, agent 1 could further increase its offer on B to \$5, resulting in its winning A for \$3. Thus, by increasing its offer on an item it is not winning—possibly above its actual valuation—the agent is able to reduce its payment. This illustrates one possible strategic behavior in the auction. However, bidding higher

[4] AkBA could also generate price quotes on a fixed schedule, or when a bid has been received from all of the participants. Similarly, variations of AkBA could terminate at a fixed time, or after a fixed number of rounds.

Solution Type	Percent of Experiments
{ABCDE}	1.5%
{ABCD} {E}	11.8%
{ABC} {DE}	6.5%
{ABC} {D} {E}	26.4%
{AB} {CD} {E}	16.3%
{AB} {C} {D} {E}	33.1%
{A} {B} {C} {D} {E}	4.4%

Table 8: Distribution of (normalized) solutions when $\beta = 1.5$.

than one's value for a item is a strategy that bears significant risk. If, for instance, after agent 1's second increase, agent 2 increased its offer on A to \$4. The best allocation would now be to give A to agent 2 and B to agent 1 at prices $(4, 4, 6)$.

4.2 A1BA Simulations with Myopic Agents

A complete game-theoretic analysis of A1BA is difficult. To gain insight into the auction's potential performance, we analyzed a protocol in which agents followed a straightforward best-response bidding strategy.

4.2.1 Simulation Design

The agents were assigned valuations on all possible bundles according the the following algorithm, parametrized by ℓ and $\beta > 0$.

1. Assign values to individual items from the uniform distribution of integers between one and ℓ. That is, $v_i(j) \in [0, \ell]$.
2. Starting with bundles of size 2, and progressively increasing the bundle size,

   Let $\underline{v}_{\boldsymbol{b}} = \max_{\boldsymbol{c} \subset \boldsymbol{b}} v_i(\boldsymbol{c})$.

   Let $\overline{v}_{\boldsymbol{b}} = \max_{\boldsymbol{c} \subset \boldsymbol{b}} v_i(\boldsymbol{c}) + v_i(\boldsymbol{b} \setminus \boldsymbol{c})$.

   Assign a value to $v_i(\boldsymbol{b})$ selected from a uniform distribution whose range is $[\underline{v}_{\boldsymbol{b}}, \underline{v}_{\boldsymbol{b}} + \beta(\overline{v}_{\boldsymbol{b}} - \underline{v}_{\boldsymbol{b}})]$.

The parameter ℓ determines the range of valuations for individual items. The parameter β controls the potential supermodularity of agent preferences. If $\beta = 0$, then an agent's valuations are extremely subadditive—an agent's valuation for the bundle is its maximal valuation for any element of the bundle. When $\beta > 1$, an agent has the potential for superadditive valuations. When $\beta = 2$, any given valuation assignment has a 0.5 probability of being superadditive.

This method of constructing random valuation functions was designed to produce problems which were likely to have both subadditive and superadditive components. The simulation consisted of 1000 randomly generated problems, each with five agents and five items, with $\ell = 10$ and $\beta = 1.5$.[5] The distribution of optimal solutions among the 1000 random problems (with items relabeled to normalize the solutions) is shown in Table 8.

The agents implemented the following myopic bidding strategy. Let $\breve{\boldsymbol{b}}_i$ be agent i's tentative allocation, as asserted by the auction. A myopic agent behaves as if it can win any bundle that it is not currently winning by bidding $\pi_{\boldsymbol{b}} + \delta$. The agent's real surplus from purchasing its tentative allocation at the announced price is $v_i(\breve{\boldsymbol{b}}) - \pi_{\breve{\boldsymbol{b}}}$. The myopic agent's strategy is to bid on bundle, $\boldsymbol{b}'$, that maximizes its real surplus at the given prices,

$$\boldsymbol{b}' = \arg\max_{\boldsymbol{b}} \begin{cases} v_i(\breve{\boldsymbol{b}}) - \pi_{\breve{\boldsymbol{b}}} & \text{if } \boldsymbol{b} = \breve{\boldsymbol{b}}, \\ v_i(\boldsymbol{b}) - (\pi_{\boldsymbol{b}} + \delta) & \text{otherwise.} \end{cases} \tag{10}$$

Then, if the solution to (10) provides strictly more surplus than the agent's tentative allocation, the agent will increase its offer on $\boldsymbol{b}'$ to $\pi_{\boldsymbol{b}'} + \delta$.[6]

4.2.2 Results of the Simulation

A1BA was run for the 1000 random problems with agents using the myopic best-response bidding policy and $\delta = .5$. In 918 of the trials, the allocation reached by the protocol was optimal.[7] The average efficiency of solutions found by the protocol was 99.8%. On average, the seller (who we assumed had zero reserve prices) captured 80% of the social welfare generated. The lowest revenue percentage was 44%.

Although these results are encouraging, the numbers themselves should be viewed with some skepticism. As has been pointed out in similar studies, efficiency percentages can be inflated by adding a large constant to everyone's valuations—the ratio $\frac{1}{2}$ does not look as good as $\frac{1001}{1002}$. Also, the minimum bid increment, δ, affects both the convergence speed and the quality of the solution: large δs converge faster, but can prevent an agent from placing a winning bid that triggers the efficient allocation.

Further, the assumption that agents would follow the myopic best-response strategy in any PCA should be suspect. An examination of the trials in which the auction terminated at inefficient allocations reveals something about possible agent strategies in A1BA. In several of these trials, an agent won a large bundle, say $\boldsymbol{b}$, without ever bidding on a subset, $\boldsymbol{c} \subset \boldsymbol{b}$, that it would receive as part of the optimal solution. Because the agent did not bid on $\boldsymbol{c}$, the auction lacked the information necessary to find the efficient allocation.

Table 9 illustrates the type of problem in which this can occur. Suppose agent 1 bids on only the bundle AB and agent 2 bids myopically. Eventually, it will win this bundle at $\pi_{AB} = \$5 \pm \delta$. Thus, it will get a surplus of $\$2 \pm \delta$. If, instead, it bids on all of the bundles, a likely outcome is that it will pay $\$2 \pm \delta$ and win B (as it should in the efficient allocation). However, agent 1's surplus is less in the second scenario than in the first, and the agent therefore has an incentive to bid strategically.

[5]Problems of sizes up to 8 items and 8 agents were solved on a desktop computer. Because we completely specify each agents' valuation function, an 8x8 problem consists of up to 2048 offers.

[6]Note that $\boldsymbol{b}'$ is not necessarily unique. When $\breve{\boldsymbol{b}}$ is one of the bundles that maximizes the agent's real surplus, the bidding policy ensures that the agent won't change its offer. However, in other case where more than one bundle maximizes (10), agents must still choose among the potential candidates. In the experimentation, such ties were broken in favor of bundles that come earlier in a particular ordering based on the bundles' binary representations.

[7]In 222 of the trials with efficient outcomes, the auction reached a different allocation than the A* search algorithm we used to compute the optimal solution. This indicates that many of the problems had at least two efficient solutions.

	A	B	AB
Agent 1	5	3*	7
Agent 2	5*	2	5

Table 9: An example in which it is in agent 1's interest not to bid on B.

5. RELATED WORK

Under some well established conditions on the agents' preferences, price equilibria exist. For example, it has been shown that price equilibria exist if a *gross substitutability* condition holds [10], or if utility functions satisfy the *no-complementarities* condition [9]. Bikhchandani and Mamer [3] demonstrated that price equilibria exist iff the total value of the solution to the discrete allocation problem is equal to the value of the corresponding relaxed linear optimization problem.

There are a several notable alternatives to the GVA designed to allocate heterogeneous items with complementarities. The most well-known mechanism is the *Simultaneous Ascending Auction* (SAA) used by the FCC to allocate spectrum rights [15, 16]. Three other promising mechanisms have been studied: the *Adaptive User Selection Mechanism* (AUSM) introduced by Banks et al. [2], a combination of the SAA and AUSM called the *Resource Allocation Design* (RAD) auction [7], and *i*Bundle [19, 20].

These mechanisms differ in many subtle ways. Two dimensions of particular relevance to this discussion are the semantics of bids and the scope of items being priced. SAA accepts bids on individual items, and sets prices on the individual items. RAD accepts bids on bundles, but still computes prices exclusively for the individual items. AUSM accepts bids on bundles and makes those bids public, essentially announcing the prices of the bundles.[8]

The auction that most closely resembles A1BA is the *i*Bundle mechanism proposed by Parkes [19]. Both A1BA and *i*Bundle are progressive combinatorial auctions whose primary innovation is in the payment determination step. Both rely on existing algorithms to solve the winner determination problem, but the two auctions differ in the manner they compute bundle prices. *i*Bundle comes in three flavors: *i*Bundle(2) uses anonymous, nonlinear prices, *i*Bundle(3) has nonlinear discriminatory prices, and *i*Bundle(d) switches from anonymous to discriminatory prices when a single agent bids on disjoint bundles. Interestingly, the motivation for switching to discriminatory prices—to present a higher price to an agent that is not winning a bundle than the price presented to the bundle's current winner—is the same situation in which A1BA reduces the price on some bundles (namely, the bundles that the agent *is* winning). In fact, the essential difference between A1BA and *i*Bundle is the manner in which the auction handles bids that contain offers on disjoint bundles. In *i*Bundle, such a bid triggers discriminatory pricing against that agent for the duration of the auction. The same bid in A1BA will cause a downward adjustment in prices only when it is necessary to attract the agent to the bundle it is tentatively assigned.

[8]This is an oversimplification. The mechanism announces the prices of the winning bundles, and provides a secondary queue with which agents bidding on smaller bundles can coordinate their bids to displace larger bundles. The prices listed in this queue imply prices for their complements with respect to supersets that have bids.

Parkes has shown that, when discriminatory pricing is allowed, agents following a myopic strategy in *i*Bundle achieve the optimal allocation, and thus the B&O third-order equilibrium prices, within a factor that is linear in the bid increment [20]. Our experimentation with myopic agents in A1BA produced efficiency results similar to those found by Parkes [19], but we have not proved a theoretical bound on the solution quality.

There are also several differences between the bidding rules used by A1BA and those used by *i*Bundle, as we understand them. Both A1BA and *i*Bundle permit XOR expressions. *i*Bundle also permits non-exclusive (OR) offers within a bid, and defines a method for handling such offers that has different strategic implications than the natural XOR expression of the same preferences. In addition, *i*Bundle uses an ascending rule akin to that used in the SAA; an agent's new bid must repeat offers for whatever bundles the agent was tentatively winning in the previous round, and any new offers must beat the quote by δ. However, offers on bundles that the agent is not tentatively winning do not have to be repeated in the next round. Although the asking price remains fixed after the non-winning offer is withdrawn, the offer itself is no longer available to be used in later solutions. Finally, *i*Bundle has an option to take an "ϵ-discount", which is meant to be used by an agent when the asking price has exceeded its valuation. In general, the bidding rules in *i*Bundle are more complex than in A1BA, and seem to allow more avenues for strategic manipulation. However, more research is needed to understand the similarities and differences between the two auctions.

6. CONCLUSION

The allocation of discrete, heterogeneous resources when agents have complementarities in preferences is a very general problem that is likely to arise in many e-commerce applications. In this paper we have established that anonymous, nonlinear equilibrium prices that support the efficient allocation always exist. We accomplish this by solving a pseudo-assignment problem for the bundles that are assigned in the optimal allocation, and then constructing prices on the rest of the bundles using a simple rule. This technique, which we refer to as k-bundle pricing, leads to a range of equilibrium price lattices.

We have also shown how the k-bundle price technique can be used as a foundation for a progressive combinatorial auction. We believe that ascending combinatorial auctions that generate anonymous bundle prices are likely to be acceptable and explainable to real participants. Thus, we see A1BA as an important step in the quest for a practical mechanism that performs well in the face of complementary preferences. We intend to continue our investigation of the family of k-bundle auctions in general, and A1BA in particular.

Acknowledgments

This research was funded by DARPA grant F30602-97-1-0228 from the Information Survivability program and a Fellowship from the IBM Institute for Advanced Commerce. We thank Ennio Stacchetti, William Walsh, Jeffrey MacKie-Mason, Fredrik Ygge, and David Parkes for their helpful comments.

7. REFERENCES

[1] A. Andersson, M. Tenhunen, and F. Ygge. Integer programming for combinatorial auction winner determination. In *Fourth International Conference on Multiagent Systems*, pages 39–46, 2000.

[2] J. S. Banks, J. O. Ledyard, and D. P. Porter. Allocating uncertain and unresponsive resources: An experimental approach. *RAND Journal of Economics*, 20:1–22, 1989.

[3] S. Bikhchandani and J. W. Mamer. Competitive equilibrium in an economy with indivisibilities. *Journal of Economic Theory*, 74:385–413, 1997.

[4] S. Bikhchandani and J. M. Ostroy. The package assignment model. Technical report, University of California at Los Angeles, November 1998.

[5] S. de Vries and R. Vohra. Combinatorial auctions: A brief survey. Technical report, Northwestern University, Evanston, IL, December 1999.

[6] G. Demange, D. Gale, and M. Sotomayor. Multi-item auctions. *Journal of Political Economy*, 94:863–72, 1986.

[7] C. DeMartini, A. M. Kwasnica, J. O. Ledyard, and D. Porter. A new and improved design for multi-object iterative auctions. Technical report, California Institute of Technology, November 1998.

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

[9] F. Gul and E. Stacchetti. Walrasian equilibrium with gross substitutes. *Journal of Economic Theory*, 87:95–124, 1999.

[10] A. S. Kelso and V. P. Crawford. Job matching, coalition formation, and gross substitutes. *Econometrica*, 50:1483–1504, 1982.

[11] T. Koopmans and M. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25:53–76, 1957.

[12] H. B. Leonard. Elicitation of honest preferences for the assignment of individuals to positions. *Journal of Political Economy*, 91(3):461–479, 1983.

[13] J. K. MacKie-Mason and H. R. Varian. Generalized Vickrey Auctions. Technical report, University of Michigan, July 1994.

[14] R. P. McAfee and J. McMillan. Auctions and bidding. *Journal of Economic Literature*, 25:699–738, 1987.

[15] R. P. McAfee and J. McMillan. Analyzing the airwaves auction. *Journal of Economic Perspectives*, 10(1):159–175, 1996.

[16] J. McMillan. Selling spectrum rights. *Journal of Economic Perspectives*, 8(3):145–162, 1994.

[17] P. Milgrom. Auctions and bidding: A primer. *Journal of Economic Perspectives*, 3(3):3–22, 1989.

[18] N. Nisan. Bidding and allocation in combinatorial auctions. In *Second ACM Conference on Electronic Commerce*, 2000.

[19] D. C. Parkes. *i*Bundle: An efficient ascending price bundle auction. *Proceedings of the ACM Conference on Electronic Commerce*, pages 148–157, November 1999.

[20] D. C. Parkes and L. H. Ungar. Iterative combinatorial auctions: Theory and practice. In *Seventeenth National Conference on Artificial Intelligence*, pages 74–81, 2000.

[21] M. H. Rothkopf, A. Pekeč, and R. M. Harstad. Computationally manageable combinational auctions. *Management Science*, 44(8):1131–1147, 1998.

[22] T. Sandholm. An algorithm for optimal winner determination in combinatorial auctions. In *Sixteenth International Joint Conference on Artificial Intelligence*, pages 542–547, 1999.

[23] M. A. Satterthwaite and S. R. Williams. Bilateral trade with the sealed bid k-double auction: Existence and efficiency. *Journal of Economic Theory*, 48:107–133, 1989.

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

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

[26] P. R. Wurman, M. P. Wellman, and W. E. Walsh. A parametrization of the auction design space. *Games and Economic Behavior*, to appear.