Multiple-Object XOR Auctions with Buyer Preferences and Seller Priorities*

Chaitanya Bandela[†] Yu Chen[‡] Andrew B. Kahng[§]
Ion I. Măndoiu[‡] Alexander Zelikovsky[*]

Abstract. Auctions and exchanges are one of the most important market mechanisms for price determination and allocation of goods. In this paper we consider the case when each buyer has a limited budget and wishes to buy at most one item in multi-item auctions. We show the limitations of two known mechanisms – sequence of single-item auctions and recently introduced XOR double auctions – and introduce a new mechanism, so called *XOR (double) auction with buyer preferences* (XOR-(D)ABP), which avoids these limitations.

In the proposed mechanism buyers specify preferences on the items on which they bid. We seek allocations of the items to the buyers which are stable with respect to buyer's preferences, i.e., items which are preferable to the item allocated to a buyer are sold for a price higher or equal to what she offered for them. In the case of double auctions, the allocation should also ensure fairness to the sellers: if an item received a bid with a higher value than the allocated

*A.Z. acknowledges financial support from the U.S. National Science Foundation (Grant CCR-9988331), from the Moldovan Research and Development Association and the U.S. Civilian Research & Development Foundation for the Independent States of the Former Soviet Union (Award No. MM2-3018), and from the State of Georgia's Yamacraw Initiative.

[†]Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA. E-mail: {bandela,alexz}@cs.gsu.edu. Tel: +1-404-651-0676. Fax: +1-404-651-2246.

[‡]Department of Computer Science, University of California at Los Angeles, Los Angeles, CA 90095, USA. E-mail: yuchen@cs.ucla.edu. Tel: +1-310-825-1872, Fax: +1-310-825-7578.

[§]Department of Computer Science & Engineering, University of California at San Diego, La Jolla, CA 92093, USA. E-mail: {abk,mandoiu}@cs.ucsd.edu. Tel: +1-858-822-5003, Fax: +1-858-534-7029

price then the buyer who placed that bid gets a more or equally preferable item. We first show that in an XOR auction with no ties in buyer preferences and bid values both buyers and sellers are better off than in an XOR auction. Second, we show that finding stable allocations with maximum revenue or buyer satisfaction can be done efficiently in an XOR-DABP without ties, and that the problem is NP-hard problem when ties are allowed. We propose a practical heuristic for finding maximum stable allocations in the presence of ties, and report promising experimental results. Third, we consider the special case when all bids for an item have the same value and give an efficient algorithm based on maximum bipartite matching. We also show that in this case stable allocations form a greedoid.

We also propose a new mechanism, so called an *XOR auction with seller priorities* (XOR-ASP), in which the seller assigns a priority to each item, and seeks allocations in which any item is allocated only if all items with higher priorities are also allocated. We show that the seller is better off using an XOR-ASP rather than a series of simple auctions, and give an efficient algorithm for finding a feasible allocation with maximum value/surplus, based on maximum-weight perfect matchings. Feasible allocations form a Gaussian greedoid, and therefore the maximum value/surplus allocation can be found even more efficiently when every buyer bids the same value on all acceptable items.

1 Introduction

Auctions and exchanges are one of the most important market mechanisms for price determination and allocation of goods. They are becoming even more important as the Internet creates the opportunity for an increasing number of consumers and businesses to participate. Traditional auction and exchange formats (English and Dutch auctions, stock exchanges, etc.) allow the participants to bid for *a single item at a time*. Recent research on combinatorial auctions (Andersson, Tenhunen, and Ygge, 2000; Fujishima, Leyton-Brown, and Shoham, 1999; Lehmann, O'Callaghan, and Shoham, 1999; Nisan, 2000; Rothkopf, Pekec, and Harstad, 1998; Sandholm, 1999; Sandholm and Suri, 2000) has attempted to extend traditional auction formats by allowing bids on bundles of items. Another attractive feature of combinatorial auctions is the possibility to place multiple mutually exclusive (XOR) bids which more accurately express buyer's true valuations and lead to more efficient item allocations compared to sequential auctions (Nisan, 2000; Sandholm and Suri, 2000). Unfortunately, in combinatorial auctions buyers cannot express any preferences induced by their limited budget.

Consider for example a sealed-bid single-round auction in which one or multiple builders offer for sale several houses. Each buyer is interested in buying a single house from a set of acceptable choices. Each buyer b has its own private value, $private(b, h)$, for each house h. A buyer with unlimited budget will simply place mutually exclusive bids, one for each acceptable house, with bid values chosen so as to equalize buyer utility. The situation is drastically changed in presence of budget constraints. Note that for real estate the budget limit may also depend on the house appraisal value, i.e., a buyer b may have a different budget $budget(b, h)$ for each house h. In this case the buyer can no longer assign bid values that would make all choices equally acceptable. Indeed, assume that b is interested in buying one of two houses, h_1 and h_2, such that

$$private(b, h_1) - budget(b, h_1) > private(b, h_2) - budget(b, h_2) > 0$$

In a second price auction a rational buyer should bid the maximum possible value for each house, i.e., $budget(b, h_i)$ for the house h_i, $i = 1, 2$. The utility derived by b, $private(b, h_i) - budget(b, h_i)$, is larger when b wins house h_1 rather than h_2. Therefore, it may be better to bid for h_1 only, and do no bid for h_2.

This implies that the limited budget forces a buyer to prefer one house over another. If a buyer would place XOR bids on her choices, current combinatorial auction mechanisms will probably force her to buy one of the most expensive houses on her list, regardless of her surplus for that house. In fact, she may end up with the least preferable house, i.e., house that give her the least surplus. When this is the case, buyers may be better off by not bidding on all acceptable choices, see, e.g., Example 2. To encourage buyers to accurately express their wishes via XOR bids, the mechanism for determining the winning bids should take into account buyer's preferences between XOR bids.

The revenue of the auction (for each item separately) would increase if more bids are placed. Therefore it is in the auctioneer interests to let each buyer to place bids in the order preferable by this buyer. Indeed, if the order is changed then the rational buyer's behavior is not to bid for items which will be sold before the item which gives the most gain for the buyer.

In this paper we give winner determination algorithms which observe buyer's preferences for some restricted types of combinatorial auctions. In our setting each buyer wants to buy a single item. Together with bid values buyers specify preferences (possibly including ties) on the items on which they bid. We seek allocations of the items to the buyers that are *stable with respect to buyer's preferences* in the sense that items which are preferable to the item allocated to a buyer are sold for a price higher or equal to what she offered for them. In the case of double auctions, the allocation should also ensure fairness to the sellers: if an item received a bid with a higher value than the allocated price then the buyer who placed that bid gets a more or equally preferable item.

The stable item allocations can be chosen according to one of the following objectives.

- *Maximum Revenue/Surplus:* find a stable allocation maximizing the sum of prices paid by the buyers, or the sum of prices paid by the buyers minus the sum of reserve prices for the sold items.

- *Maximum Buyer Satisfaction:* find, if it exists, the stable allocation in which each buyer gets the most preferable item among all items that she can get in a stable allocation.

Finding the stable allocations with either the maximum total value/surplus or maximum buyer satisfaction can be done efficiently when there are no ties in buyer preferences and bid values. As soon as buyers have ties in their preferences, i.e., if they do not differentiate between two or more items that they bid on, or if the bid values have ties, i.e., two buyers happen to bid the same value on the same item, stable allocations with maximum buyer preference may no longer exist, and finding a stable allocation with maximum value/surplus becomes NP-hard.

We further consider the important special case of XOR auctions with buyer preferences in which all bids for an item have the same value. This models, e.g., the situation in which the parties involved do not assign bid values, but only express interest in starting bilateral negotiations. For example, consider a government agency having a certain number of projects. Various independent contractors bid on these projects, each giving her partial order of preferences for projects that she bids on. The objective of the agency is to assign the maximum number of these projects to various contractors with a constraint that a contractor is assigned a project that is less preferable to her only when all projects more preferable to her are assigned to someone else.

In this case the stability condition becomes weaker: buyers are guaranteed to get the most preferable item among those not taken by others. We show that stable allocations form a greedoid when the seller does not distinguish between items, e.g., all items have the same reserve price. This implies that the maximum size stable allocation can be computed efficiently.

In an *XOR auction with seller priorities* (XOR-ASP), the seller assigns a priority to each item, and seeks allocations in which any item is allocated only if all items with higher priorities are also allocated. We show that the seller is better off using an XOR-ASP rather than a series of simple auctions, and give an efficient algorithm for finding a feasible allocation with maximum value/surplus, based on maximum-weight perfect matchings. Feasible allocations form a Gaussian greedoid, and therefore the maximum value/surplus allocation can be found even more efficiently when every buyer bids the same value on all acceptable items.

The paper is organized as follows. In the next section we introduce the maximum stable allocation (MSA) problem for XOR-DABP. In Section 3 we show the advantages of XOR-DABP without ties over XOR double auctions. Then, in Section 4, we give practical exact and approximation algorithms for the MSA problem, and report promising experimental results. In Section 5 we study weakly stable allocations for XOR-ABPs and give an efficient algorithm for finding maximum size weakly stable allocations. In Section 6 we show that weakly stable allocations form a greedoid, and investigate its properties. Finally, in Section 7, we formally introduce XOR-ASP, compare it with a series of simple auctions, and give efficient algorithms for finding feasible allocations with maximum value or surplus.

2 XOR Double Auctions with Buyer Preferences

In this section we introduce XOR double auctions with buyer preferences and define stable allocations for them. Consider an XOR double auction with a set B of buyers and a set I of items for sale. Each buyer b is interested in buying a single item from a subset I_b of I. We assume that buyer b places mutually exclusive bids on the items in I_b. The value offered by b for item $i \in I_b$ is denoted by $v(b,i)$.

In an *XOR Double Auction with Buyer Preferences (XOR-DABP)*, buyers have preferences for the items on which they bid. We write $i \prec_b j$ when buyer b strictly prefers item $i \in I_b$ to item $j \in I_b$, and $i \preceq_b j$ when b does not strictly prefer j to i. When $\prec_b$ is a total order on I_b we say that b has *strict preferences*.

An *item allocation* L is a set of pairs (b,i), $b \in B$, $i \in I_b$, such that each buyer $b \in B$ and item $i \in I$ appears in at most one pair of L. When $(b,i) \in L$ we say that b and i are *matched* by L. We denote by $B(L)$ and $I(L)$ the set of buyers, respectively items, that are matched by L. For each

$b \in B(L)$ ($i \in I(L)$) we denote by $L(b)$ ($L(i)$) the unique item (buyer) to which b (resp. i) is matched by L. The *allocation value* of item $i \in I(L)$ is $V_L(i) = v(L(i), i)$.

Definition 1 *An item allocation L is* stable *if, for each buyer $b \in B$ and item $i \in I_b$, $(b, i) \notin L$ implies that $L(b) \preceq_b i$ or $v(b, i) \leq v(L(i), i)$.*

Stable allocations are simultaneously fair to buyers and sellers in the following sense:

(1) A buyer cannot complain that she got a less preferable item (or no item at all) since more preferable items were sold for a price higher or equal to what she offered for them, and

(2) A seller cannot complain that she got less money for an item (or that the item has not been sold) since every buyer that bids a larger (resp. any) price for such an item gets a more or equally preferable item.

Theorem 2 *Stable item allocations always exist.*

Proof: A stable allocation can be found by arbitrarily breaking ties in buyer preferences and bid prices, and then running the Gale-Shapley algorithm (Gale and Shapley, 1962) extended to handle incomplete lists (Gusfield and Irving, 1989; see Figure 1). Since the allocation computed by the Gale-Shapley algorithm is stable under the strict preferences obtained after breaking ties, it will also be stable under the original (non-strict) preferences. ■

In general, an XOR-DABP admits more than one stable allocation. In this paper we focus on the problem of finding stable allocations with maximum revenue, formally defined as follows:

Maximum Stable Allocation (MSA) Problem. Given an instance of XOR-DABP, find a stable allocation L with maximum revenue $\sum_{i \in I(L)} V_L(i)$.

1. For each $i \in I$, $L(i) \leftarrow b$, where b is the buyer that bids the largest value on i
2. While there exist items $i, i' \in I$ s.t. $L(i) = L(i') = b$ do

   If $i' \prec_b i$, then swap i and i'

   $L(i') \leftarrow b'$, where b' is the buyer that bids the *next largest* value on i'
3. Output allocation L

Figure 1: Seller-optimal Gale-Shapley allocation algorithm for strict preferences $\prec_b$, $b \in B$, and no ties in bid values.

An alternative objective is to find, if it exists, the stable allocation with maximum buyer satisfaction, i.e., the stable allocation in which each buyer gets the most preferable item among all items that she can get in a stable allocation When items have reserve prices, another objective is finding a stable allocation with maximum total surplus, $\sum_{i \in I(L)} (V_L(i) - r(i))$, where $r(i)$ denotes the reserve price of item i. Note that the problem of maximizing total surplus is not identical to MSA, since XOR-DABPs may admit stable item allocations of different cardinalities (Manlove et al.). (This contrasts the classical result for the Stable Marriage and Hospitals/Residents problems that all stable matchings have the same cardinality and the sets of matched men/women are the same over all stable matchings).

Theorem 3 *The MSA problem is NP-hard under either one of the maximum value or maximum surplus objectives. The problem remains NP-hard even when all buyers have strict preferences, or when all bid values are distinct.*

Proof: The proof follows by a reduction from the problem of finding a maximum cardinality stable marriage with incomplete preference lists and ties (Max-Cardinality SMTI), which was recently proved to be NP-hard by (Manlove et al.). Given an instance of Max-Cardinality SMTI, we construct an MSA instance as follows: items correspond to men, buyers correspond to women with the same preference lists, and the value of the bid placed by buyer w on item m is set to $1 + k\varepsilon$, where k is the rank of w in m's preference list. For small enough ε any maximum stable allocation for the MSA instance gives a maximum cardinality stable marriage.

The NP-hardness of the restricted cases of MSA follows in the same way from the NP-hardness of correspondingly restricted versions of Max-Cardinality SMTI (Manlove et al.). ■

In the next section we will show that the MSA problem is polynomial-time solvable in the case when there are no ties in bid values and buyers have strict preferences. In Section 5 we will give an efficient algorithm for the important special of the MSA problem when all bids the same value, i.e., finding a maximum cardinality stable allocation. Another case known to be polynomial-time solvable is when buyers have no preferences, i.e., XOR double auctions. In this case finding the MSA reduces to computing the maximum-weight matching in a bipartite graph representing all bids.

3 XOR-DABP without Ties

In this section we consider the case when there are no ties in bid values and buyer preferences. Note that we can always break ties in an XOR-DABP by giving preference to bids placed earlier. In this case, stable allocations for XOR-DABP correspond to stable matchings in a stable marriage instance with incomplete lists. Therefore, all stable allocations assign the same set of items to the same set of buyers. Furthermore, the lattice structure of stable matchings (Gusfield and Irving, 1989) implies the following two properties of maximum stable allocations:

(1) There exists a unique stable allocation, called the *seller-optimal* allocation, which simultaneously maximizes the total value and the total surplus. Every item receives under this allocation the maximum price over all stable allocations.

(2) There exists a unique stable allocation, called the *buyer-optimal* allocation, with maximum buyer satisfaction. Under this allocation each buyer gets the most preferable item among all items that she can get in a stable allocation.

The stable allocation with maximum value/surplus can be computed efficiently using the seller-optimal version of the Gale-Shapley algorithm with incomplete preferences (Gusfield and Irving, 1989; see Figure 1). The stable allocation with maximum buyer satisfaction can also be computed efficiently using a buyer-optimal version of the algorithm. Thus we have:

Theorem 4 *When there are no ties in bid values and buyers have strict preferences, the MSA problem for XOR-DABP is polynomial time solvable under either one of the maximum value/surplus or maximum buyer satisfaction objectives.*

XOR auctions have the attractive property that revenues increase with increasing number of buyers. Unfortunately, as shown by the following example, when a new buyer joins an XOR *double* auction, any individual seller may be worse off (either may get a smaller price for her item, or may not sell the item at all).

Example 1: Consider an XOR double auction in which two items, i_1 and i_2, are sold by different sellers. A buyer b_1 bids $2x - \varepsilon$ monetary units on item i_1 and x monetary units on i_2. In the absence of any other bids, item i_1 is allocated to b_1, for a price of $2x - \varepsilon$. If another buyer b_2 joins the auction and bids x monetary units on i_1, the maximum value allocation would assign i_1 to b_2 for a price of x and i_2 to b_1 also for a price of x. Thus, the value of i_1 goes down from $2x - \varepsilon$ to x when b_2 is added to the auction. ■

We next prove that revenue monotonicity still holds for XOR-DABP. The following lemma, which holds for either one of the three MSA objectives, follows from Theorem 1.4.3 in (Gusfield and Irving, 1989):

Lemma 5 *Let L be the maximum stable allocation for an XOR-DABP, and let L' be the maximum stable allocation after buyer b adds a new item i to I_b such that $j \prec_b i$ for $j \in I_b$, $j \neq i$. Then*

(1) For every item i, $v_{L'}(i) \geq v_L(i)$.

(2) $L'(b) \prec_b L(b)$

Corollary 6 *Adding new buyers to an XOR-DABP cannot decrease the price of any allocated item.*

The following example shows that any individual buyer may be better off by not revealing all her acceptable alternatives in an XOR auction.

Example 2: Consider an XOR auction with two items for sale, i_1 and i_2. Buyer b_1 considers both i_1 and i_2 acceptable choices, and assigns them a value of $2x - \varepsilon$, respectively x. However, b_1 prefers i_1 to i_2. Buyer b_2 has only one acceptable choice, i_1, and bids a value of x on it. Assume that there are no other bids on i_1 and i_2. If b_1 bids only on item i_1 then she gets it for a price of $2x - \varepsilon$, while b_2 doesn't get anything. On the other hand, if b_1 places the bids on both i_1 and i_2, the maximum value allocation would assign i_1 to b_2 and i_2 to b_1, for a price of x each. By bidding on all her acceptable choices, bidder b_1 has worsened her outcome: she ends up getting her last preference although her first choice is sold for half the price that she offered. ■

The following corollary to Lemma 5 shows that buyers are always better off by revealing their complete lists of preferences in an XOR-DABP.

Corollary 7 *Regardless of the bids of the other buyers, the best strategy for each buyer in an XOR-DABP is to reveal truthfully (i.e., in the true order of preference) all her acceptable choices.*

4 Allocation Algorithms for XOR-DABP

4.1 Exact Solution Based on Integer Linear Program

Below we give an integer program formulation for the MSA problem. This formulation can be used with available commercial MIP solvers (e.g., Cplex 6.5) to compute optimum solutions for MSA instances of moderate size. The integer program sets the variable x_{bi} to 1 if item i is allocated to buyer b, and to 0 otherwise. The constraints enforce that every item is allocated to at most one buyer (c1), that every buyer gets at most one item (c2), and that the resulting allocation is stable (c3).

$$
\begin{aligned}
\max \quad & \sum_{b \in B} \sum_{i \in I_B} x_{bi} v(b,i) & & \\
\text{s.t.} \quad & \sum_{b \in B : i \in I_b} x_{bi} \leq 1, & i \in I & \quad (c1)\\
& \sum_{i \in I_b} x_{bi} \leq 1, & b \in B & \quad (c2)\\
& \sum_{b' \in B : i \in I_{b'}} x_{b'i} v(b', i) & & \\
& \qquad \geq v(b,i)(1 - \sum_{j : j \preceq_b i} x_{bj}), & b \in B, i \in I_b & \quad (c3)\\
& x_{bi} \in \{0,1\} & b \in B, i \in I_b &
\end{aligned}
$$

Remark: It is known that the constraints $x_{bi} \in \{0,1\}$ become unnecessary in case when there are no ties, i.e., the MSA is given by the solution to a linear, not integer, program. This gives another proof that the problem is polynomial time solvable in this case.

4.2 A Greedy Tie-Breaking Heuristic for the MSA Problem

In this section we suggest a practical heuristic for the MSA problem. The heuristic breaks the ties in buyer preferences in non-increasing order of bid values (see Figure 2). As shown by (Manlove et al.), the ratio between the maximum and minimum cardinality of a stable allocation is at most two. This immediately gives the following upper-bound on the approximation factor of the greedy tie-breaking heuristic:

1. Break ties in non-increasing order of bid values (e.g., among bids in a tie, the bid with largest value becomes most preferable)
2. Break remaining ties arbitrarily
3. Find the seller-optimal stable matching using the Gale-Shapley algorithm (see Figure 1)

Figure 2: The Greedy Tie-Breaking Heuristic for the MSA Problem

Theorem 8 *The greedy tie-breaking heuristic has an approximation factor of at most 2ρ, where ρ is the ratio between the largest and the smallest bid values.*

The following example shows that the approximation guarantee established in Theorem 8 is tight up to a constant factor.

Example 3: Consider an XOR-DABP instance with:

- $B = \{b_1, b_2\}$
- $I = \{i_1, i_2, i_3\}$
- Bid values: $v(b_1, i_1) = L$, $v(b_1, i_2) = v(b_2, i_2) = \varepsilon$, $v(b_2, i_3) = 2\varepsilon$
- Preferences: $i_2 \prec_{b_1} i_1$; b_2 has no preference between i_2 and i_3.

Then the maximum value stable allocation is $\{(b_1, i_1), (b_2, i_2)\}$, with total value $L+\varepsilon$. The greedy tie-breaking heuristic breaks the tie such that $i_3 \prec_{b_2} i_2$, and returns the stable allocation $\{(b_1, i_2), (b_2, i_3)\}$, with total value 3ε. ■

4.3 Experimental Study of MSA Algorithms

In this section we report preliminary experimental results comparing the greedy tie-breaking heuristic for the MSA problem with optimum results computed using the MIP Solver from the Cplex 6.5 commercial optimization package and the integer linear program formulation given in Section 4.1. Our experiments were run on randomly generated XOR-DABP instances modeling the real estate application presented in the introduction.

The generator further allows the user to select the distribution of the number of bids per buyer and the method used to generate ties in buyer preferences. More importantly, the generator has provisions for generating XOR-DABP instances with a structure likely to be encountered in practical applications. Buyers and items are partitioned into a user specified number of classes, and the user may control what classes of buyers can bid on what classes of items, as well as control reserve and bid value distributions at class granularity. These parameters can be used to model, for example, differences in item popularity or buyer wealth.

Table 1 gives results for the greedy tie-breaking heuristic and the Cplex MIP solver on XOR-DABP instances with 200–2000 items divided into 3 classes, 200–8000 buyers divided into 2 classes, and 800–64000 bids. All reserve prices and bid surplus values were generated from normal distributions.

5 Weakly Stable Allocations

In this section we introduce weakly stable allocations for XOR-ABPs, give an efficient algorithm for finding the maximum value weakly stable allocation, and establish connections with greedoid theory.

Throughout this section we consider that there is a single seller, and hence fairness to sellers reduces to maximizing the total value (or, alternatively, total surplus over reserve prices) of sold items. An allocation is said to be *weakly stable* if, for any buyer b, there is no unallocated item that b prefers to the item she is allocated (in particular, if b does not get any item, then all items which she bids for must be allocated to other buyers). With the notations in Section 2, allocation L is weakly stable if

(1) For any $b \in B(L)$ and $k \in I - I(L) \cap I_b$, $L(b) \preceq_b k$, and

1. Find the maximum unweighted matching M between buyers and items
2. While there is $b \in B$ and $i \in I - M(I)$ such that $M(b) \succ_b i$ do
3. Find an item j which is the most preferable item for b in $I - M(I)$
4. Swap allocation of b, i.e., $M \leftarrow M - (b, M(b)) \cup (b, j)$
5. Output M

Figure 3: The Swapping Algorithm for the MWSA problem.

(2) For any $b \in B - B(L)$, $I_b \subseteq I(L)$.

Maximum Weakly Stable Allocation (MWSA) Problem. Given an instance of XOR-ABP, find a weakly stable allocation L with maximum total value.

The complexity of MWSA problem is open: we do not know if it is NP-hard, and we are not aware of a polynomial time algorithm either. Note that the maximum matching does not produce a weakly stable allocation and the greedy tie-breaking heuristic may result in an unbounded error (see Example 3). However, there is a non-trivial case of the MWSA problem for which we give an exact solution. If all the the bid values are the same, the the MWSA problems asks to maximize the number of allocated items. This problem can be solved efficiently by modifying a maximum unweighted matching between buyers and items (see Figure 3). In the next subsection we will show that weakly stable allocations form a greedoid, this yields another algorithm for finding a weakly stable allocation with maximum size.

Theorem 9 *The Swapping Algorithm (Figure 3) finds the maximum size weakly stable allocation for XOR-ABP.*

Proof: We need to show that the number of iterations in the loop 4 is polynomially bounded. Indeed, each time after performing such iteration, the buyer b

improves the preference of the allocated item. Therefore, in total, the number of iterations cannot exceed the number of bids. ■

Theorem 9 implies that we can use the Swapping Algorithm to approximately solve the MWSA problem for weakly stable allocations.

Theorem 10 *The Swapping Algorithm has an approximation factor of at most ρ, where ρ is the ratio between the largest and the smallest bid value.*

6 Greedoids and Weakly Stable Allocations

In this subsection we show that the set weakly stable allocations form a greedoid, which gives a more efficient algorithm for finding a maximum size stable allocation. We also show that the corresponding greedoids, so called ABP greedoids, do not have the exchange property. This implies (Korte, Lovász, and Schrader, 1991) that the maximum weight stable allocation cannot be found with a greedy algorithm.

A set $I \in J(I)$ is called *feasible.* The family of independent sets in a matroid satisfy these requirements, so every matroid is a greedoid. One significant difference between matroids and greedoids is that every subset of an independent set is independent in a matroid, but a feasible set in a greedoid will have non-feasible subsets in general.

Definition: A *transversal* of a finite family $\mathcal{B} = \{I_1, \ldots I_n\}$ of subsets of a finite set I is a set $T \subseteq I$ for which a bijection $f : T \to \{1, \ldots, n\}$ exists such that $i \in I_{f(i)}$ for all $i \in T$. A *partial transversal* is a subfamily of $\mathcal{B}$.

Assume now that each $I_i = (I_i, \preceq_i)$ has a partial order on its elements (e.g., preferences). A *stable transversal* of $\mathcal{B}$, is a partial transversal T of $\mathcal{B}$ such that for any $i \in T$, if $x \prec_{f(i)} i$, then $x \in T$.

The following remark establishes the connection between weakly stable allocations and stable transversals.

Remark: Let I be the set of items and I_i be the set of items for which buyer

i place a bid. Then the bijection f of a stable transversal is a weakly stable allocation and vice versa.

Definition: A *greedoid* on the ground set I is a pair $(I, J(I))$ where J is a family of subsets of I satisfying the following two properties:

1. For every non-empty $I \in J$, there is an element $i \in I$ such that $I - \{i\} \in J$
2. For $I, K \in J$ with $\mid I \mid < \mid K \mid$, there is an element $k \in K - I$ such that $I \cup \{k\} \in J$

Theorem 11 *The set of all stable transversals $T(I)$ forms a greedoid with ground set I.*

Proof: To show that $(I, T(I))$ is a greedoid we need to show that

(i) For any $A \in T(I)$, there exists $a \in A$ such that $A - \{a\} \in T(I)$.

(ii) For any $A, B \in T(I)$ and $|B| > |A|$, there is an element $b \in B - A$ such that $A \cup \{b\} \in T(I)$.

Proof of (i). Consider arbitrary element $a_0 \in A$. If $A - a_0$ is not stable, then there is $y_1 \in M(A)$ such that $a_1 = M(y_1) \succ_{y_1} a_0$. Inductively, if $A - a_{i-1}$ is not stable, then there is $y_i \in M(A)$ such that $a_i = M(y_i) \succ_{y_i} a_{i-1}$, for $i = 1, 2, \ldots$. Since the set A is finite, there should be i such that either $A - a_i$ is stable or $a_i = a_j$ for some $j < i$.

Assume that M is a preferred stable matching for A. Consider the matching M' coinciding with M on all elements of A except a_k, $k = j, \ldots, i$, for which $M'(a_k) = M(a_{k+1})$, $k = j + 1, \ldots, i$, and $M'(a_j) = M(a_i)$. The matching M' is more preferable than M, therefore, M is not preferred.

Proof of (ii). Let $A, B \in T(I)$ and $|B| > |A|$, and let M' be a stable matching for B and let M be a stable matching for A with the minimum $|M' - M|$. Since

$|B| > |A|$, there exists $y \in M'(B) - M(A)$. We first prove that there exists $b \in B - A$ such that $b \preceq_y M'(y)$.

Indeed, let us assume that on the contrary, for any $b \in B - A$, $b \succ_y M'(y)$. Then $M'(y) \in B \cap A$ and the matching $M'' = M \cup (M'(y), y) - (M'(y), M(M'(y)))$ is stable. Indeed, for any $x \in E - A$, either $x \in B - A$ and $M'(y) \prec_y x$ or $x \in E - B$ and $M'(y) \preceq_y x$ since B is stable. Since $|M' - M''| = |M' - M| - 1$, we have a contradiction with the choice of M.

Now assume that $b \in B - A$ is the best w.r.t. y among all elements in $B - A$. We will show that $A \cup \{b\}$ is stable since $M \cup \{(b, y)\}$ is stable. Indeed, for any $x \in E - A - b$, either $x \in B - A$ and $b \preceq_y x$ by the choice of b, or $x \in E - B$ and $b \preceq_y M'(y) \preceq_y x$ since B is stable. ∎

Let us refer to greedoids $(I, T(I))$ as *greedoids for XOR auctions with buyer preferences* (ABP greedoids). The greedy algorithm for finding a maximum size feasible set in a greedoid starts with an empty set and then iteratively adds new elements keeping the set feasible until no more elements can be added. If we apply the greedy algorithm to the ABP greedoid then we will find a maximum size stable transversal, and, as remarked above, this corresponds to a maximum size weakly stable allocation.

Theorem 12 *The greedy algorithm for the ABP greedoid finds a maximum size weakly stable allocation for the corresponding XOR-ABP.*

If we assign weights to the elements of a greedoid, then, in general, the greedy algorithm does not find a maximum weight feasible subset. In (Korte, Lovász, and Schrader, 1991) it is proved that the greedy algorithm works for weights if it has the following exchange property: if $T_1, T_2 \in T(I)$ are any two bases and $| T_1 | = | T_2 |$, then for any $x \in T_1$ there exists $y \in T_2 - T_1$ such that $T_1 \setminus \{x\} \cup \{y\}$ is a feasible transversal. However, this is not the case for a ABP greedoid as seen from the following example.

Example 4: Consider a set $H = \{h_1, h_2, h_3, h_4, h_5, h_6\}$ with the total order $h_5 \preceq h_6 \preceq h_2 \preceq h_1 \preceq h_3 \preceq h_4$. A family of subsets over H as $\mathcal{B} =$

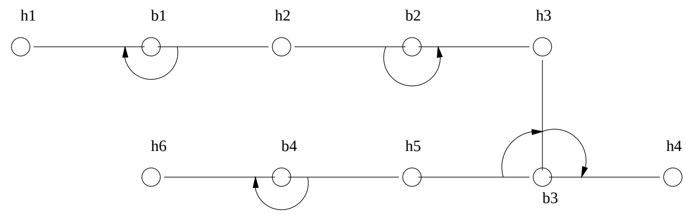

Figure 4: An ABP greedoid which does not have exchange property.

$\{B_1, B_2, B_3, B_4\}$ where $B_1 = \{h_2, h_1\}$; $B_2 = \{h_2, h_3\}$; $B_3 = \{h_5, h_3, h_4\}$; $B_4 = \{h_5, h_6\}$ (see Figure 4).

Consider a transversal base $T_1 = \{(h_2, B_1), (h_3, B_2), (h_4, B_3), (h_5, B_4)\}$. Consider another transversal base $T_2 = \{(h_1, B_1), (h_2, B_2), (h_5, B_3), (h_6, B_4)\}$. Now $\mid T1 \mid = \mid T2 \mid$. Therefore, by the exchange property, for every $h_i \in T_1$, there exists a $h_j \in T_2$ - T_1, such that T_1 - $\{h_i\}$ + $\{h_j\}$ is a feasible transversal. But for $h_3 \in T_1$, there exists no element in T_2 that satisfies this property. ■

7 XOR Auctions with Seller Priorities

In this section we introduce XOR auction with seller priorities, give algorithms for finding a maximum feasible allocation, and, finally, describe connections with greedoids.

In an *XOR auction with seller priorities* (XOR-ASP), the seller assigns priorities to the items for sale. We write $i \preceq j$ to denote the fact that the priority of i is higher than that of j, and $i \prec j$ if $i \preceq j$ and $j \not\preceq i$. We say that seller priorities are *strict* (respectively *total*) if for any two items i and j, either $i \prec j$ or $j \prec i$ (respectively, either $i \preceq j$ or $j \preceq i$). An item allocation L is *feasible* if for every two items i and j, $i \prec j$, $j \in I(L)$ implies $i \in I(L)$, i.e., L allocates an item only if all items with higher priorities are also allocated.

Maximum Feasible Allocation (MFA) Problem. Given an instance of XOR-ASP, find an allocation L maximizing the total value/surplus.

Clearly the maximum feasible allocation cannot be worse than an alloca-

1. Construct the bipartite graph $G = (B \cup I, E, w)$ where $(b, i) \in E$ iff buyer b bids on item i, and the weight $w(b, i)$ is the value of this bid
2. Let $i_1, \ldots, i_{|I|}$ be the items I sorted by decreasing priority
3. For $k = 1, \ldots, |I|$, find, if it exists, a maximum-weight perfect matching M_i in the subgraph G_k of G induced by $B \cup \{i_1, \ldots, i_k\}$
4. Output the matching M_i with maximum weight

Figure 5: The Iterated Perfect Matching Algorithm for the MFA Problem.

tion implied by the sequence of auctions, each for a single item, ordered with respect to priorities. The following example shows that, in general, the opposite is not true, i.e., the seller may be strictly better off using an XOR-ASP rather than a sequence of simple auctions.

Example 5C: onsider a seller auctioning items i_1, i_2 with a priority of selling item i_1 before i_2 i.e., $i_1 \prec i_2$. Assume that buyer b_1 bids a value of x on i_1 and of y on i_2 and buyer b_2 bids a value of $x - \varepsilon$ on i_1. If the two items are sold in a series of auctions in decreasing order of priority, i_1 is allocated to b_1 while i_2 is not sold. The total value of the sold items is x. On the other hand, if the two items are sold in an XOR-ASP, item i_1 is allocated to b_2 and item i_2 is allocated to b_1, with a total value of $x + y - \varepsilon$. ■

Theorem 13 *The Iterated Perfect Matching Algorithm (Figure 5) finds the maximum feasible allocation for the auction with strict seller priorities in* $O(|I| \cdot T_{PM}(n, m))$ *time, where* $T_{PM}(n, m)$ *is the time needed to compute a maximum weight perfect matching in a bipartite graph with* $n = |B| + |I|$ *vertices and* $m = \#bids$ *edges.*

Remark: The status of the MFA problem for auctions with total seller priorities is open.

7.1 Greedoids and Seller Priorities

Let $\mathcal{I} = \{B_1, \ldots, B_n\}$ be a finite family of subsets of a finite set B. Let $\prec$ be a strict order on $\mathcal{I}$, i.e., $B_1 \prec B_2 \prec \ldots \prec B_n$. A partial transversal T of $\mathcal{I}$ is called a *feasible* transversal if it transverses a subfamily $\{B_1, \ldots, B_k\}$, for some $k \leq n$.

Remark: Let B be the set of buyers and B_i be the subset of buyers which bid for the item $i \in I$. Then the bijection f of a feasible transversal is a feasible allocation and vice versa.

Example 2.14 from (Korte, Lovász, and Schrader, 1991) describes so called *medieval marriage greedoids*, in which feasible sets are exactly feasible transversals. This implies the following result.

Theorem 14 *The set of all feasible transversals $T(B)$ forms a* Gaussian *greedoid* $(B, T(B))$.

We will refer to these greedoids as *greedoids for auctions with seller priorities* (ASP greedoids).

Theorem 15 *(Korte, Lovász, and Schrader, 1991) ASP greedoids have the exchange property.*

This result implies that the greedy algorithm solves exactly the MFA problem in case when all bids from the same bidder has the same value.

8 Acknowledgments

We would like to thank Frank Olken and Amit Sahai for bringing the problem to our attention and fruitful discussions.

References

Andersson. A, M. Tenhunen, and F. Ygge. (2000). "Integer programming for combinatorial auction winner determination," In *Proc. ICMAS*, pp. 39–46, extended version available from http://www.enersearch.se/∼ygge.

Fujishima, Y., K. Leyton-Brown, and Y. Shoham. (1999). "Taming the computational complexity of combinatorial auctions: optimal and approximate approaches," In *Proc. IJCAI*, pp. 548–553, also available from http://robotics.stanford.edu/∼kevinlb.

Gale, D. and L.S. Shapley. (1962). "College admissions and the stability of marriage," *Amer. Math. Monthly* 69, pp. 9–15.

Gusfield, D. and R.W. Irving. (1989). *The Stable Marriage Problem: Structure and Algorithms.* Cambridge, MA: MIT Press.

Korte, B., L. Lovász, and R. Schrader. (1991). *Greedoids*, Algorithms and Combinatorics, vol. 4, Berlin: Springer-Verlag.

Lehmann, D., L. O'Callaghan, and Y. Shoham. (1999). "Truth revelation in rapid, approximately efficient combinatorial auctions," In *Proc. ACM-EC*, pp. 96–102, available from http://www.cs.huji.ac.il/∼lehmann/.

D. MANLOVE AND R.W. IRVING AND K. IWAMA AND S. MIYAZAKI AND Y. MORITA Hard Variants of Stable Marriage. *Theoretical Computer Science* 276 (2002), pp. 261–279.

Nisan, N. (2000). Bidding and allocation in combinatorial auctions. In *Proc. ACM-EC*, pp. 1–12.

Rothkopf, M., A. Pekec, and R. Harstad. (1998). "Computationally manageable combinatorial auctions," *Management Science 44*, pp. 1131–1147.

Sandholm, T. (1999). "An algorithm for optimal winner determination in combinatorial auctions," In *Proc. IJCAI*, pp. 543–547.

Sandholm, T. and S. Suri. (2000). "Improved algorithms for optimal winner determination in combinatorial auctions and generalizations," In *Proc. AAAI*, pp. 90–97.

#items	#buyers	#bids	Greed val.	CPU sec.	Cplex MIP val.	CPU sec.	Gap
200	200	800	75128257	0.03	78576249	2.46	4.39%
200	200	1600	74325339	0.06	84054224	285.45	11.57%
200	400	1600	88942384	0.08	89066527	0.93	0.14%
200	400	3200	89499018	0.17	89589954	4.21	0.10%
200	800	3200	90892284	0.27	90916681	1.83	0.03%
200	800	6400	91123664	0.55	91140024	11.86	0.02%
500	500	2000	182720456	0.15	200382640	20	8.81%
500	500	4000	188462442	0.3	213948423	1568.62	11.91%
500	1000	4000	221822991	0.48	223916856	5.61	0.94%
500	1000	8000	224921119	0.91	225496946	19.89	0.26%
500	2000	8000	223311871	1.68	223415781	11.47	0.05%
500	2000	16000	223996927	3.42	224104530	21.1	0.05%
1000	1000	4000	377607773	0.6	411979940	84.85	8.34%
1000	1000	8000	383358512	1.2	N.A.	N.A.	N.A.
1000	2000	8000	436211164	1.95	442829836	16.41	1.49%
1000	2000	16000	445455065	3.94	446120926	76.57	0.15%
1000	4000	16000	450705268	6.67	450921237	28.81	0.05%
1000	4000	32000	452143070	13.45	452334922	108.5	0.04%
2000	2000	8000	750054567	2.61	N.A.	N.A.	N.A.
2000	2000	16000	778591088	5.21	N.A.	N.A.	N.A.
2000	4000	16000	885838136	7.92	N.A.	N.A.	N.A.
2000	4000	32000	897567709	15.92	N.A.	N.A.	N.A.
2000	8000	32000	891690542	26.13	N.A.	N.A.	N.A.
2000	8000	64000	894947694	52.63	N.A.	N.A.	N.A.

Table 1: Results for the greedy tie-breaking heuristic and the Cplex MIP solver on XOR-DABP instances with normal distributed reserve prices and bid values.