Combinatorial Auctions with Decreasing Marginal Utilities

[Extended Abstract] *

Benny Lehmann
Digital Fuel
benny@digifuel.com

Daniel Lehmann
School of Computer Science and Engineering The Hebrew University Jerusalem, Israel
lehmann@cs.huji.ac.il

Noam Nisan
School of Computer Science and Engineering The Hebrew University Jerusalem, Israel
noam@cs.huji.ac.il

ABSTRACT

In most of microeconomic theory, consumers are assumed to exhibit decreasing marginal utilities. This paper considers combinatorial auctions among such buyers. The valuations of such buyers are placed within a hierarchy of valuations that exhibit no complementarities, a hierarchy that includes also OR and XOR combinations of singleton valuations, and valuations satisfying the gross substitutes property. While we show that the allocation problem among valuations with decreasing marginal utilities is NP-hard, we present an efficient greedy 2-approximation algorithm for this case. No such approximation algorithm exists in a setting allowing for complementarities. Some results about strategic aspects of combinatorial auctions among players with decreasing marginal utilities are also presented.

1. INTRODUCTION

1.1 Background

Recent years have seen a surge of interest in combinatorial (also called combinational) auctions, in which a number of non-identical items are sold concurrently and bidders express preferences about combinations of items and not just about single items. Thus, for example, a bidder may offer \$40 for the combination of a Tournedos Rossini and a bottle of Chateau Lafitte, but offer only \$10 for each of those items alone. Similarly, a bidder may make an offer of \$10 for a blue and for a red shirt, but not be willing to pay more than \$10 even if he gets both shirts. In general, a combinatorial auction allows bidders to express complementarities – where the value of a combination of packages of items is worth more than the sum of the values of the separate packages – and substitutabilities – where the value of a combination of packages is less than the sum of the values of the separate packages. Such combinatorial auctions have been suggested for a host of auction situations such as those for spectrum licenses, pollution permits, landing slots, computational resources, and others. See [2] for a survey.

Implementation of combinatorial auctions faces several challenges including the representational question of succinctly specifying the values of the different packages, the algorithmic challenge of efficiently solving the resulting, NP-hard, allocation problem, and the game-theoretic questions of bidders' strategies. These questions have been recently approached by a host of researchers both in the general case and in several interesting special cases [12, 23, 17, 20, 4, 22, 6, 10, 15].

Somewhat surprisingly, the special case that is most natural from an economic sense has received very little attention from the computational point of view. In most of microeconomic theory, consumers are assumed to exhibit diminishing marginal utilities and they expect bulk discounts. In particular, such consumers exhibit *no complementarities*. In fact many papers dealing with allocation in combinatorial auctions focused on the dual case of no substitutes, i.e., much computational research assumes that a buyer places bids for packages of items and is not interested in a sub-package. In contrast, economists who dealt with auctions did mostly consider auctions in which players expressed no complementarities. For example, for multi-unit auctions, Vickrey's seminal paper [24] assumes *downward sloping* valuations for buyers. Recent papers dealing with combinatorial auctions such as [7, 8, 9, 1] usually assume the *gross substitutes* property. Each of these notions implies lack of complementarities.

In this paper, we study the three notions for lack of complementarity found in the literature. In increasing level of restrictiveness these notions are, informally:

- **No Complementarities**: The value of a combination of bundles is no more than the sum of the bundle values.
- **Decreasing Marginal Utilities**: The marginal value of an item decreases as the set of items already acquired increases.
- **Gross Substitutes**: The demand for an item does not decrease when the price of other items increases.

We pay special attention to the case of *decreasing marginal utilities*, which is equivalent to submodularity of the valuation functions.

*Supported by grants from the Israeli Ministry of Science and the Israeli Academy of Sciences

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

1.2 Representation and Hierarchy

We first consider the question of how to represent bids (i.e., valuations) that have no complementarities. They can be represented by valuations constructed out of atomic valuations that offer a price for a single item, (*singleton valuations*), using the operations of OR and XOR. Using restricted classes of such bidding languages we are able to obtain a structural hierarchy of families of complement-free valuations.

Theorem: Denote by CF – the class of complement-free valuations; SM – the class of valuations with decreasing marginal utilities (submodular valuations); GS – the class of valuations that satisfy the gross substitutes property; XOS – the class of XOR-of-OR-of-singleton valuations; OXS – the class of OR-of-XOR-of-singleton valuations. Then:

$$OXS \subset GS \subset SM \subset XOS \subset CF.$$

Further more, all these containments are strict.

1.3 Allocation

We then focus our attention on the question of allocation among players with submodular valuations. We should first stress the difference between this case and the more restrictive and well-studied case of allocation among valuations satisfying the gross substitutes property. In the latter case, it is known that a Walrasian equilibrium exists [9], while in the former Walrasian equilibria do not necessarily exist [7]. Walrasian equilibria may be found in polynomial time and are the basis of almost all known computationally efficient allocation algorithms. One may thus say that the submodular case is the first hard case from a computational point of view[1].

We first show that the optimal allocation problem remains NP-hard even among players of submodular types. Our main positive algorithmic result is a simple greedy algorithm that produces an allocation that is a 2-approximation, i.e., an allocation whose value is at least half of the optimal one. This is in sharp contrast to the general case where it remains NP-hard to find even a $n^{1/2-\epsilon}$-approximation [19, 10].

Greedy Algorithm: Enumerate the items in arbitrary order, and allocate each item to the bidder with highest marginal valuation in the current partial allocation.

Theorem: If all bids are submodular then this algorithm produces an allocation whose value is at least half the optimal one.

We do not know if a better approximation ratio is possible, nor do we know whether a polynomial time approximation algorithm exists that works for all complement-free valuations. As mentioned, the restricted case of valuations with gross substitutes has a polynomial-time optimal solution based on linear programming. The more restricted case of OXS valuations has an optimal solution based on bipartite matching. We note that for both of these algorithms, appropriate (and non-standard) access to the valuation functions is needed.

[1] A similar phenomena exists from the information transfer point of view: any optimal allocation algorithm among submodular valuations requires exponential communication, while if the valuations satisfy the gross substitutes property then polynomial communication suffices [16].

1.4 False Name Bids

Our next concern is with the issue of *false-name bids* that was identified and analyzed in a sequence of papers [18, 25, 26, 27]. This concerns the following problem that is inherent in combinatorial auctions that use the Vickrey-Clark-Groves payment rules [12] – the only choice, if one requires incentive compatibility and efficiency. These rules dictate that a player that is allocated a set S of items pays the external cost to society. A disturbing observation made in [18] is that in many cases a bidder can manipulate a VCG combinatorial auction and reduce his payment by splitting his bid and placing two separate bids under *false-names*. For example, if two items A and B are offered and my valuation for the pair $\{A, B\}$ is 6, while another bidder values the pair at 5, the VCG rules will set my payment to 5. If, instead, I place two separate bids, $\{A\}$ for 4 and $\{B\}$ for 4, then one may easily calculate that each of these *false-name bidders* pays $1 = 5 - 4$ for a total payment of 2. In [18] it is shown that, in the case of single-item multi-unit auctions, this type of savings cannot occur when all bids are downward sloping. It is claimed in there (without proof and without precise definitions) that this generalizes to combinatorial auctions when all bidders have no complementarities. We show that this is not exactly the case[2], but rather the generalization to combinatorial auctions requires that the combined valuation of other bidders has decreasing marginal utilities.

Theorem: *False-name bids* cannot reduce payments in a combinatorial auction using the VCG rules, whenever the combined valuation of all other players is submodular.

There is no requirement on the valuation of the player placing false-name bids. The requirement of submodularity is on the combined valuations of all other players. We provide an example showing that requiring each player be of a submodular type is not sufficient. In particular, we observe, this result implies the result of [18] regarding single-item multi-unit auctions.

1.5 Paper Structure

In section 2 we present the basic definitions of the classes of valuations that we discuss. Section 3 discusses representations of valuations using OR and XOR expressions, and provides a structural hierarchy of subclasses of complement-free valuations. Section 4 provides many examples of complement-free and submodular valuations, both natural ones, and various negative examples. Section 5 discusses allocation (winner determination) algorithms. Section 6 discusses false-name bids. Finally, section 7 shortly mentions an extension to valuations with bounded complementarities.

2. DEFINITIONS

2.1 Preliminaries

In this paper we consider a combinatorial auction of non-identical goods: There is a set X of items for sale by a single auctioneer in a single combinatorial auction. We will denote the number of items $|X| = m$. There are n bidders who all desire these items. Each bidder i has its own private

[2] We have contacted the authors of this paper and they have confirmed a bug in their unpublished proof.

valuation function, v_i, that specifies his valuation for each possible subset of items that he may get: i.e. for a subset $A \subseteq X$ of items, $v_i(A)$ is the amount of money at which bidder i values this subset A of items. For a singleton set $\{x\}$ we will use $v_i(x)$ as a shorthand for $v_i(\{x\})$. This notation makes two important assumptions:

- **Quasi-linearity:** Bidders' utilities can be measured in terms of "money", i.e. are linear in the "money".
- **No Externalities:** The bidders' valuation depends only on the set of items he wins: the valuation function v_i is $v_i = v_i(A)$, where A is the set of items won by bidder i, not on the identity of the bidders who get the items not in A.

In addition, we assume that these valuations all satisfy the following conditions:

- **Free disposal:** Items have non-negative value. Thus v_i satisfies $v_i(A) \leq v_i(B)$, whenever $A \subseteq B$.
- **Normalization:** $v_i(\emptyset) = 0$.

The auctioneer's aim is to find an optimal *allocation.*

DEFINITION 1. *An* Allocation *is a partition of X into pair wise disjoint sets of items $S_1 ... S_n$. An allocation is* optimal *if it maximizes $\sum_i v_i(S_i)$.*

The auction rules must also define the payments received from each bidder.

2.2 Marginal Valuations

A central notion that we will be using is that of the marginal valuation. It describes how a player would value sets of remaining items if he were already given some items.

DEFINITION 2. *Given a valuation v on a set X of items and a set $W \subseteq X$ of items, the marginal valuation of a set $A \subseteq X - W$ given W is defined by:*

$$v(A|W) = v(A \cup W) - v(W).$$

One may consider the marginal valuation of a single element, $v(x|\cdot)$, as a discrete analog for the partial derivative of v in direction x.

2.3 No Complementarities

DEFINITION 3. *A valuation v is called* complement-free *if it satisfies:*

$$v(A) + v(B) \geq v(A \cup B)$$

for all sets A, $B \subseteq X$. The class CF is the set of all complement free valuations.

While this notion is clearly natural, it turns out that valuations with this property can still have "hidden complementarities". We could expect consumers to exhibit no complementarities even once they have acquired some items. Example 4 shows that even if v is in CF, the valuation $v(\cdot|W)$ need not be – i.e. once a set of items W is already acquired, complementarities surface. Indeed, as example 4 shows, this condition of complement-freedom is not the right analog of downward-sloping valuations. It is therefore only natural to turn to those valuations that have no such "hidden complementarities". It turns out that these are exactly the submodular valuations.

2.4 Decreasing Marginal Utilities – Submodular Valuations

DEFINITION 4. *A valuation v is called* submodular *if for every two sets of items $S \subseteq T$ and element x, $v(x|T) \leq v(x|S)$. Submodular valuations are also called valuations with* decreasing marginal utilities. *The class SM is the set of all submodular valuations.*

Thus we require that the marginal utility of an element decreases as the set of items already acquired increases. Submodular functions are well-known and heavily used in combinatorics [14]. If we extend the analogy between the marginal valuation and the derivative discussed in section 2.2, then we see why submodular valuations are considered a discrete analog of convex functions – the "derivative" $v(x|\cdot)$ is decreasing. Here, we consider only valuations: monotone, positive submodular functions. In the literature, submodular functions have been generally been considered in a wider setting. Many equivalent characterizations of decreasing marginal utilities are well-known.

THEOREM 1. *(see for example [13, 14]) A valuation v is submodular if and only if any one of the following equivalent propositions holds.*

- *For any $x, y \in X$ and $S \subseteq X$: $v(x|S) \geq v(x|S \cup \{y\})$.*
- *For any $S, T, V \subseteq X$, such that $S \subseteq T$: $v(V|S) \geq v(V|T)$.*
- *For any $A, B \subseteq X$: $v(A) + v(B) \geq v(A \cup B) + v(A \cap B)$.*

It follows, in particular, from the last characterization that any valuation with decreasing marginal utilities has no complementarities.

COROLLARY 1. *A valuation with decreasing marginal utilities is complement-free.*

The converse is not true and in example 4 below we exhibit a valuation that has no complementarities yet its marginal utilities are not decreasing. It turns out that valuations with decreasing marginal utilities are exactly the valuations without any "hidden" complementarities.

LEMMA 1. *A valuation v is submodular if and only if for every subset of items R, the marginal valuation function $v(\cdot|R)$ has no complementarities.*

PROOF. For the *if* direction, use the third characterization from theorem 1, so we need to show that for all A, B: $v(A \cup B) + v(A \cap B) \leq v(A) + v(B)$. Now set $R = A \cap B$, $A' = A - R$, and $B' = B - R$, and we have, $v(A) = v(A'|R) + v(R)$, $v(B) = v(B'|R) + v(R)$, $v(A \cup B) = v(A' \cup B'|R) + v(R)$, and $v(A \cap B) = v(R)$. Thus the condition $v(A \cup B) + v(A \cap B) \leq v(A) + v(B)$ is equivalent to $v(A' \cup B'|R) \leq v(A'|R) + v(B'|R)$, which is true due to $v(\cdot|R)$ being complement-free.

For the *only if* direction, we now need to prove that for all R and $A', B' \subseteq R^c$: $v(A' \cup B'|R) \leq v(A'|R) + v(B'|R)$. Now set $A = A' \cup R$, and $B = B' \cup R$, and so again, $v(A) = v(A'|R) + v(R)$, $v(B) = v(B'|R) + v(R)$, $v(A \cup B) = v(A' \cup B'|R) + v(R)$, and $v(A \cap B) = v(R)$. Thus the condition $v(A' \cup B'|R) \leq v(A'|R) + v(B'|R)$ is equivalent to $v(A \cup B) + v(A \cap B) \leq v(A) + v(B)$ which is true due to v being submodular. □

In particular we get the corollary.

COROLLARY 2. *If v is submodular, then for every subset R of items the marginal valuation function $v(\cdot|R)$ is submodular.*

PROOF. All the marginal valuation functions of $v(\cdot|R)$ are also marginal valuation functions of v and thus have no complementarities. □

2.5 Gross Substitutes

In order to define the gross substitutes property we first need to consider the effect of putting prices on items. We think of what happens to the valuation of v of a set S, when the price of all items in S must be paid. All the definitions and theorems in this subsection are from [9, 7] (we have sometimes slightly modified the terminology and notation).

DEFINITION 5. *Given a vector of real item prices $\vec{p} = (p_1...p_m)$, the* surplus *of a set of items S relative to these prices is defined as $v(S|\vec{p}) = v(S) - \sum_{i \in S} p_i$. A set S is a* preferred set *of v at prices $\vec{p}$ if $v(S|\vec{p}) = \max_T v(T|\vec{p})$. i.e. S maximizes the surplus. The* demand set *of v at prices $\vec{p}$ is the set of all preferred sets $D(v|\vec{p}) = \{S \mid v(S|\vec{p}) = \max_T v(T|\vec{p})$.*

The gross substitutes property mandates that increasing the price of an item will not decrease the demand for any other item. I.e. if an item i is in a preferred set at prices $\vec{p}$, then increasing p_j for some $j \neq i$, will still have item i in a preferred set.

DEFINITION 6. *A valuation v is said to satisfy the* gross substitutes *property if for any item i, any price vector $\vec{p}$ and any price vector $\vec{q} \geq \vec{p}$ (point wise comparison) with $p_i = q_i$, we have that if $i \in A$ where $A \in D(v|\vec{p})$, then there exists $A' \in D(v|\vec{q})$ such that $i \in A'$. The class GS is set of all valuations that satisfy the gross substitutes property.*

An equivalent condition is called the *single improvement property*. This condition states that a non-preferred set can always be improved (in terms of its surplus) by deleting at most one element from it and inserting at most one element into it.

LEMMA 2. *[7] A valuation v satisfies the gross substitutes property if and only if for any $\vec{p}$ and $A \notin D(v|\vec{p})$, there exists A' such that $|A' - A| \leq 1$ and $|A - A'| \leq 1$ and $v(A'|\vec{p}) > v(A|\vec{p})$.*

The gross substitutes property is stronger than submodularity.

LEMMA 3. *[7] A valuation that satisfies the gross substitutes property is submodular.*

The key property that makes valuations with the gross substitutes property so convenient is that in an auction with such valuations, Walrasian equilibria exist. A Walrasian equilibrium is a vector of prices on the items and an allocation such that every bidder receives a preferred set at these prices.

DEFINITION 7. *A Walrasian equilibrium in an auction with valuations $v_1...v_n$ is a price vector $\vec{p}$ and an allocation $A_1...A_n$ such that for all bidders j: $A_j \in D(v_j|\vec{p})$.*

THEOREM 2. *[9] Any auction with valuations $v_1...v_n$ that satisfy the gross substitutes property has a Walrasian equilibrium.*

It turns out that this is essentially also a necessary condition.

THEOREM 3. *[7] If v_1 does* not *satisfy the gross substitutes property then there exist valuations $v_2...v_n$ that do satisfy the gross substitutes property[3] and the auction with $v_1...v_n$ does* not *have a Walrasian equilibrium.*

It is easy to verify the "first Welfare theorem" in this context: any Walrasian equilibrium gives an optimal allocation, i.e. one that maximizes $\sum_j v_j(A_j)$.

3. REPRESENTATION AND STRUCTURE

3.1 Elements of Representation

The question of representation of valuations, i.e., how to succinctly describe a valuation without listing explicitly the values for each of the 2^n subsets of items, must be addressed before valuations can be efficiently treated in any allocation algorithm. How can we represent complement-free valuations succinctly? We may consider representing such valuations by combining *simple* valuations by suitable operators.

The basic syntactic elements from which valuations are usually constructed, sometimes called atomic bids, are a declaration of a certain price for a specific set of elements. They correspond to the single-minded valuations of [10]. The only single-minded valuations in CF are valuations in which only a single item is valued at a positive value. It is therefore expected that those *singleton* valuations will play a central role in describing CF valuations.

DEFINITION 8. *For an item $x \in X$ and a price p, the singleton valuation e_x^p is a valuation giving the same value p to all sets containing x and the value 0 to all sets that do not include x.*

The operators commonly used for representing valuations are those introduced in [20]: OR and XOR. They are often understood as prescriptions concerning the compatibility of bids, but the following presentation, as operations on valuations, is more convenient.

DEFINITION 9 ([15]). *Let v_1 and v_2 be two valuations on the set X of items. The valuations $v_1 \oplus v_2$ (XOR) and the valuation $v_1 \vee v_2$ (OR) are defined by:*

$$(v_1 \oplus v_2)(S) = max(v_1(S), v_2(S)),$$

$$(v_1 \vee v_2)(S) = \max_{T \subseteq S}(v_1(T) + v_2(S - T)).$$

The OR ($\vee$) of two valuations represents the valuation of an agent representing both valuations, bidding on their behalf and sharing the result between them. An OR of two valuations is the valuation obtained by partitioning the set of goods obtained optimally between these valuations. The XOR ($\oplus$) of two valuations represents the valuation of a single bidder capable of choosing between two possible but incompatible personalities after the auction. His value for a set S is the largest of the two values of the component valuations. The operations OR and XOR are obviously associative and commutative.

[3]In fact, [7] prove that $v_2...v_n$ may be unit demand functions – an even more restricted notion.

3.2 A Syntactic Hierarchy

It is now natural to study the hierarchy of valuations obtained by combining singleton valuations by the OR and XOR operations. Here are the levels of the hierarchy

1. Singleton valuations.
2. $OS - valuations$: The family of valuations that can be described by OR of singletons is precisely the family of (separately) *additive valuations*: those valuations which value any set at the sum of the values of its elements. They exhibit no complementarity and no substitutability.
3. $XS - valuations$: The family of valuations that can be described by XOR of singletons is usually called the family of *unit-demand valuations* [7]: those valuations that value any set of elements at the value of the element of the set that is most valued. This family of types has been considered central by Vickrey and since.
4. $OXS - valuations$: The family of OR's of XS valuations represents the aggregated valuation of multiple unit-demand players.
5. $XOS - valuations$: The family of XOR's of additive valuations turns out to subsume the whole hierarchy. This follows from the simple observation that OR distributes over XOR:

   $$(a \oplus b) \vee (c \oplus d) = (a \vee c) \oplus (a \vee d) \oplus (b \vee c) \oplus (b \vee d).$$

   Therefore XOS is closed under OR (it is obviously closed under XOR). It follows that any expression that uses combinations of OR and XOR operations on singleton valuations can be represented as a simple XOS expression.

3.3 Closure Theorems

Once we have defined the operations of OR and XOR it is natural to ask whether the classes of valuations defined semantically (CF, SM, and GS) are closed under these operations.

3.3.1 CF

THEOREM 4. *The class CF is closed under OR and XOR.*

PROOF. Let v_1 and v_2 be complement free then:

$$(v_1 \oplus v_2)(S \cup T) = max(v_1(S \cup T), v_2(S \cup T)) \leq$$

$$max(v_1(S) + v_1(T), v_2(S) + v_2(T)) \leq$$

$$max(v_1(S) + v_2(S)) + max(v_1(T) + v_2(T)) =$$

$$(v_1 \oplus v_2)(S) + (v_1 \oplus v_2(T)).$$

Similarly fix partitions $(S_1 : S_2)$ of S and $(T_1 : T_2)$ of T, such that $(v_1 \vee v_2)(S) = v_1(S_1) + v_2(S_2)$ and $(v_1 \vee v_2)(T)) = v_1(T_1) + v_2(T_2)$. Now,

$$(v_1 \vee v_2)(S \cup T) \leq v_1(S_1 \cup T_1) + v_2(S_2 \cup T_2) \leq$$

$$v_1(S_1) + v_1(T_1) + v_2(S_2) + v_2(T_2)) =$$

$$(v_1 \vee v_2)(S) + (v_1 \vee v_2)(T).$$

□

3.3.2 SM

The class of submodular valuations is *not* closed under neither OR nor XOR. It is not closed under XOR as it contains the class OS of additive valuations, but example 3 shows that it does not contain the class XOS which is XORs of additive valuations. The fact that it is not closed under OR is given by example 5.

3.3.3 GS

The class GS is not closed under XOR as it contains the class OS of additive valuations, but example 3 shows that even its superset SM does not contain the class XOS which is XORs of additive valuations. It is however closed under the OR operation.

THEOREM 5. *If u and w satisfy the gross substitutes property then so does $u \vee w$.*

PROOF. We will prove it indirectly applying theorems 3 and 2. Using theorem 3, it suffices to show that for all valuations $v_2 ... v_n$ that satisfy the gross substitutes property, an auction with bidders $u \vee w, v_2 ... v_n$ has a Walrasian equilibrium. Theorem 2 states that an auction with bidders $u, w, v_2, ..., v_n$ has a Walrasian equilibrium: vector of prices $\vec{p}$ with allocation $A_u, A_w, A_2, ... A_n$. We claim that $\vec{p}$ with the allocation $A_u \cup A_w, A_2, ... A_n$ is a Walrasian equilibrium for bidders $u \vee w, v_2 ... v_n$.

The only thing that needs to be verified is that $(A_u \cup A_w) \in D(u \vee w|\vec{p})$. This is so since for any set S, let S_u and S_w be a partition of S such that $(u \vee w)(S) = u(S_u) + w(S_w)$. Thus also $(u \vee w)(S|\vec{p}) = u(S_u|\vec{p}) + w(S_w|\vec{p})$. Now, since $A_u \in D(u|\vec{p})$ we have that $u(S_u|\vec{p}) \leq u(A_u|\vec{p})$, and similarly for w. Clearly $u(A_u) + w(A_w) \leq (u \vee w)(A_u \cup A_w)$, and so also $u(A_u|\vec{p}) + w(A_w|\vec{p}) \leq (u \vee w)(A_u \cup A_w|\vec{p})$. Thus

$$(u \vee w)(S|\vec{p}) = u(S_u|\vec{p}) + w(S_w|\vec{p}) \leq$$

$$\leq u(A_u|\vec{p}) + w(A_w|\vec{p}) \leq (u \vee w)(A_u \cup A_w|\vec{p}).$$

□

3.4 The Complete Hierarchy

We have seen by now five significant classes of valuations. Three of them were defined semantically: complement-free (CF), submodular (SM), and those with the gross substitutes property (GS). Two of them were defined syntactically: ORs of XORs of singletons (OXS) and XORs of ORs of singletons (XOS). We now can state the relationships between these classes.

THEOREM 6.

$$OXS \subset GS \subset SM \subset XOS \subset CF.$$

All containments are strict.

PROOF. We will prove the containments from left to right. The fact that they are strict will follow from the examples below, in Section 4.2.

1. $OXS \subseteq GS$: This follows from theorem 5 since GS contains all unit demand valuations (XS-valuations) [7], and thus also contains ORs of XS valuations.
2. $GS \subseteq SM$: This was shown in [7].

3. $SM \subseteq XOS$: each submodular valuation may be represented by a long XOR expression. For each permutation π of the items we will have an OR clause – the XOS expression will be the XOR of all these OR clauses. The OR clause for the permutation π will offer for each item i the marginal price of i assuming that all items preceding it in the permutation π have already been obtained. Formally, for $i = \pi(j)$, the price is $v(i|\{\pi(1), ..., \pi(j-1)\})$. To see that the XOS expression indeed represents the submodular valuation, consider a submodular v and a set A of items. For any permutation π of the items in which the items of A are placed first, the value given to A by the OR clause for π is exactly $v(A)$. All other OR clauses give to A a value that is smaller or equal, since v has decreasing marginal utilities.

4. $XOS \subseteq CF$ By Theorem 4.

□

4. EXAMPLES OF SUBMODULAR VALUATIONS

4.1 Natural Examples

The simplest and most common natural examples of valuations with no complementarities are the additive valuations (the class OS) and the unit demand valuations (the class XS) mentioned in section 3.2. These valuations are both trivially in subclasses of OXS and thus clearly satisfy the gross substitutes property, and are also submodular. We now mention some other natural valuations that are submodular.

4.1.1 Symmetric valuations

Symmetric valuations are ones where $v(S)$ depends only on the size of S, $\mid S \mid$. These cases fit auctions of multiple identical items. Symmetric valuations may be described by a sequence of *marginal* values: numbers $p_1 \ldots p_m$ where $p_i = v(T) - v(S)$ for any sets T and S of sizes i and $i-1$ respectively. We then have: $v(S) = \sum_i^{|S|} p_i$ for any set S. The following set of symmetric valuations has received considerable attention. For example Vickrey's multi-unit auction [24] only considers such valuations.

DEFINITION 10. *A symmetric valuation is downward sloping iff for all i, $p_{i+1} \leq p_i$.*

As expected, these are exactly the symmetric submodular valuations.

PROPOSITION 1. *A symmetric valuation is downward sloping if and only if it is submodular.*

PROOF. For a symmetric valuation v, for $x \notin S$, $v(x|S) = p_{|S|+1}$, and the equivalence is immediate. □

It is known [7] that downward sloping symmetric valuations also satisfy the gross substitutes property and in fact can be represented in the class OXS [15]. On the other hand, the complement-free symmetric valuations are a wider class – see below examples 4 and 3.

4.1.2 Additive valuations with a budget limit

Our second class of examples is composed of valuations that are additive-up-to-a-budget-limit. In these valuations, a set is valued at the sum of the values of its items, unless this sum is larger than a budget limit. In this last case it is valued at the budget limit.

DEFINITION 11. *A valuation v is called additive with a budget limit if there exists a constant b, the budget limit, such that for all sets S of items, $v(S) = min(b, \sum_{i \in S} v(\{i\}))$.*

PROPOSITION 2. *Every additive valuation with a budget limit is submodular.*

PROOF. Assume $S \subseteq T$ and $x \notin T$. We have

$$v(x|T) = min(b, v(T) + v(\{x\})) - v(T) =$$

$$min(b - v(T), v(\{x\})) \leq min(b - v(S), v(\{x\})) = v(x|S).$$

□

These type of valuations do not necessarily satisfy the gross substitutes property – see below example 2.

4.1.3 Valuations based an underlying measure

The third class of valuations we want to present consists of valuations based on some underlying set with a measure on it. Let us first start with a concrete example. Consider a combinatorial auction for a set of spectrum licenses in overlapping geographical regions. Each geographical region contains a certain population, and a reasonable valuation for a set of licenses is the total population in the geographic area covered by the regions in the set. It turns out that such valuations are submodular. More generally, the overlapping regions may be arbitrary sets in some underlying base set, and the population count may be replaced by any measure on the base set.

DEFINITION 12. *A valuation v is said to be based on the underlying measure μ on a base set Γ if there are sets $I_1 \ldots I_m \subseteq \Gamma$ such that for each set S of goods, $v(S) = \mu(\cup_{i \in S} I_i)$.*

LEMMA 4. *Every valuation that is based on an underlying measure is submodular.*

PROOF. Consider $S \subseteq T$, and an item x. Denote $\tilde{S} = \cup_{i \in S} I_i$, and similarly $\tilde{T}$. Clearly $\tilde{S} \subseteq \tilde{T}$. We now have $v(x|S) = \mu(I_x - \tilde{S}) \geq \mu(I_x - \tilde{T}) = v(x|T)$. □

4.2 Separation Examples

4.2.1 $OXS \neq GS$

EXAMPLE 1. *Consider a set X of four items: $X = \{a, b, c, d\}$ and the valuation v defined by: the value of any singleton is 10 and the value of any other set is 19, except for the two sets $\{a, c\}$ and $\{b, d\}$ the value of which is 15.*

CLAIM 1. *The valuation v is not in OXS.*

PROOF. Suppose $v = v_1 \vee \ldots \vee v_n$. Since $v(\{a\}) = 10$, there is some i with $v_i(\{a\}) = 10$. Without loss of generality, assume $i = 1$. Since $v(\{b\}) = 10$, there is some i with $v_i(\{b\}) = 10$. Since $v(\{a,b\}) < 20$, for any $i \neq 1$, $v_i < 10$. We conclude that $v_1(\{b\}) = 10$. Similarly, $v_1(\{c\}) = v_1(\{d\}) = 10$. Consider now that $v(\{a,c\}) = 15$, therefore, since $v_1(\{a\}) = 10$, for any $i > 1$, $v_i(\{c\}) \leq 5$. Similarly, for any item x and any $i > 1$, $v_i(\{x\}) \leq 5$. But $v(\{a,b\}) = 19$ and therefore there must be $i \neq j$ such that $v_i(\{a\}) + v_j(\{b\}) = 19$, a contradiction. □

CLAIM 2. *The valuation v is in GS.*

PROOF. (sketch) We will use the single improvement property (lemma 2). Let $\vec{p}$ be any price vector and consider any non-preferred set A. Let $D \in D(v|\vec{p})$ be a preferred set. We need to show that we can improve the surplus of A by deleting at most one element and inserting at most one element. This requires a tedious case by case analysis that we omit, except for the only nontrivial case where $A = D^c$ and both are of size 2, e.g. $A = \{a,b\}$ and $D = \{c,d\}$. In this case either $\{a,d\}$ or $\{b,c\}$ will have a surplus greater than A's. □

4.2.2 GS $\neq$ SM

EXAMPLE 2. *Consider the additive valuation with budget limit 4 on three elements: $v(1) = v(2) = 2$, $v(3) = 4$. Since the budget limit is 4, each set that contains at least two elements has a valuation of 4.*

This valuation is submodular by Proposition 2, but it does not satisfy the gross substitutes property. Consider the price vector $p_1 = 0, p_2 = 1, p_3 = 2$. At these prices the preferred subset is $\{1,2\}$ giving a surplus of 3. However, if we increase the price of item 1 to $p_1 = 2$, then the preferred subset is $\{3\}$ giving a surplus of 2, where any set that contains item 2 will have a surplus of at most 1. Thus we have reduced the demand for item 2.

4.2.3 SM $\neq$ XOS

EXAMPLE 3. *Consider the symmetric valuation on three elements defined by $p_1 = 2$, $p_2 = 0$, $p_3 = 1$: a set of one or two elements is valued at 2, while the set of all three elements is valued at 3.*

By definition, it is not downward sloping, therefore not submodular, by Proposition 1, but it is obtained by the following XOR-of-OR-of-singletons:

$$(\{1\}:2) \oplus (\{2\}:2) \oplus (\{3\}:2) \oplus w$$

where $w = (\{1\}:1) \vee (\{2\}:1) \vee (\{3\}:1)$.

4.2.4 XOS $\neq$ CF

EXAMPLE 4. *Assume a set X of three items and $v(S) = 2$ if $|S| = 1$, $v(S) = 3$ if $|S| = 2$ and $v(S) = 5$ if $|S| = 3$.*

The valuation v is in CF since $2 + 2 \geq 3$ and $2 + 3 \geq 5$. We shall show that it cannot be expressed as a XOR-of-ORs-of-singletons. Assume that it is, and consider the OR clause that provides the valuation of 5 to the set of all three elements. This OR clause contains (at most) three singleton bids for the three items; if we take the two highest bids of these three we must get a valuation of at least $5 \cdot 2/3 > 3$, in contradiction to the valuation of any two elements being 3.

This is also an example for the fact that the class CF is not closed under marginal valuations. The marginal valuation v_W, where W contains any single item, gives $v_W(S) = 1$ if $|S| = 1$ and $v_W(S) = 3$ if $|S| = 2$ and is not in CF.

4.2.5 SM is not closed under OR

The following example will show that SM is not closed under OR.

EXAMPLE 5. *Let $u(1) = 3$, $u(2) = 5$ and $u(3) = 3$, where u has a budget limit of 6. Let w be the additive valuation: $w(1) = 1$, $w(2) = 2$, $w(3) = 0$. Let $v = u \vee w$.*

The valuation u is submodular by Proposition 2. One can see that: $v(\{1,2\}) = 6$ (u gets both), $v(\{2,3\}) = 6$ (u gets both). Therefore $v(\{1,2\}) + v(\{2,3\}) = 12$. But $v(\{1,2,3\}) = 8$ (u gets 1 and 3 and w gets 2), $v(2) = 5$ (u gets it), and $8 + 5 > 12$ and v is not submodular.

5. ALLOCATION

We now turn to the computational problem of allocating the items in a combinatorial auction in which all bidders are submodular. As a computational problem, we must first consider the format of the input, i.e. how are the valuation functions $v_1 \ldots v_n$ presented to the algorithm. In the most general case representing a valuation may require exponential size (in m, the number of items). We, on the other hand, are looking for algorithms that are polynomial in the relevant parameters, n and m. There are two possible approaches for obtaining efficient algorithms despite the exponential size of the input.

The first approach considers the general case, but where the valuations are presented to the algorithm as "valuation oracles" – black boxes that can be queried for the valuation of a set S, returning its valuation $v(S)$. In a mechanism, this corresponds to allowing bidders to send an arbitrary representation of the valuation function, as long as the valuations of sets can be efficiently computed from it[4]. The second approach fixes a representation, and provides allocation algorithms that require polynomial time in the size of the representation of the valuations in this format. Such algorithms will generally be as interesting as the strength of the representation format.

We present our main positive result, in section 5.3, in the general terms of valuation oracles. Our main negative result, the NP-completeness of allocation for submodular valuations, presented in subsection 5.2, holds for a special case (additive valuations with a budget limit) that has a simple short representation. Concrete lower bounds for the valuation oracle representation are derived in [16].

5.1 The Case of Gross Substitutes

The case of a combinatorial auction where all valuations satisfy the gross substitutes property is considered easy. Since Walrasian equilibria exist in this case [9] (see definition 7 and theorem 2 above), they can be found and an optimal allocation results. let us look more closely at the

[4]Surprisingly, this is not totally trivial – see section 3.6 of [15].

computational process of finding this Walrasian equilibrium. The following procedure of [9] yields prices and an allocation that is arbitrarily close to optimal:

1. Initialize all item prices to 0.

2. Repeat the following procedure: compute a preferred set for each bidder at these prices and increase by a small amount ϵ the price of all items that are demanded by more than one bidder.

3. Stop when each item is demanded by at most one bidder, and let the allocation be the preferred sets at these prices.

This procedure produces an allocation whose distance from optimal depends only on ϵ and whose running time is polynomial in n, m, ϵ^{-1}. This algorithm is naturally implemented as a mechanism, in some cases is actually incentive compatible, and in the general case can serve as a basis for an incentive compatible mechanism [1]. We wish to consider two computational issues here.

The first issue is the fact that this procedure only provides an approximation (it is actually a "FPAS" – fully polynomial approximation scheme) and not an exact solution. We observe[5] that an exact solution can also be computed efficiently using linear programming. A slight difficulty exists since the natural linear programming formulation of the allocation problem has an exponential number of variables, but one can solve it using a separation-based algorithm on the dual. See [16] for more details.

The second issue to consider is how the valuations are presented to the algorithm. This procedure requires the computation of preferred sets. This may be termed as access to a *demand oracle* for v – accepting as input a price vector $\vec{p}$ and outputting a preferred set $S \in D(v|\vec{p})$ at these prices. It turns out that this demand oracle is also exactly what is needed for finding the exact solution using a separation based linear programming algorithm. It is not clear how to obtain such a demand oracle given only a valuation oracle for the valuations. The proof of theorem 7 actually implies that there is no general transformation, as computing a preferred set for an additive valuation with a budget limit is NP-hard.

Open problem: Can an optimal or near-optimal allocation be found in polynomial time among GS valuations that are given by valuation oracles?

A special case of interest is when all valuations in the auction are OXS valuations. If these valuations are given using the OR of XOR of singletons representation, then the allocation problem reduces to a matching problem in a bipartite graph and can thus be computed in polynomial time [21, 3]. We do not know whether an optimal allocation can be computed efficiently given only valuation oracles even if all valuations are known to be in OXS.

5.2 NP-hardness of exact solution

The problem of finding an optimal allocation in combinatorial auctions is known [17] to be NP-hard even if bidders are single-minded, i.e., place only one bid each. Is the problem any easier if the bidders are assumed to have submodular valuations? The answer is negative. Finding an optimal allocation is still an NP-hard problem, even if all valuations are additive with budget limit. Note that, in such a case, all players valuations may be succinctly expressed.

THEOREM 7. *Finding an optimal allocation in a combinatorial auction with two valuations that are additive with a budget limit is NP-hard.*

PROOF. We will reduce from the well-known NP-complete problem "Knapsack" [5]: Given a sequence of integers $a_1 ... a_m$, and a desired total t, determine whether there exists some subset S of the integers whose sum is t, $\sum_{i \in S} a_i = t$. Given an input of this form, construct the following two valuations on m items:

- The first valuation is additive giving the price a_i to each item i: $v_1(S) = \sum_{i \in S} a_i$.
- The second valuation is additive with a budget limit of $2t$, and gives the price $2a_i$ to each item i: $v_2(S) = 2 \cdot \min(t, \sum_{i \in S} a_i)$.

Fix an allocation of S to valuation 2 and S^c to valuation 1 and consider the 3 cases: $\sum_{i \in S} a_i < t$, $\sum_{i \in S} a_i = t$, and $\sum_{i \in S} a_i > t$. Denote $f = \sum_i a_i$. If $\sum_{i \in S} a_i = t$ then $t + \sum_{i \in S^c} a_i = f$ and $v_1(S^c) + v_2(S) = \sum_{i \in S^c} a_i + 2t = f + t$. If $\sum_{i \in S} a_i < t$ then $v_1(S^c) + v_2(S) = \sum_{i \in S^c} a_i + 2 \sum_{i \in S} a_i = f + \sum_{i \in S} a_i < f + t$. If $\sum_{i \in S} a_i > t$ then $\sum_{i \in S^c} a_i + t < f$, and $v_1(S^c) + v_2(S) = \sum_{i \in S^c} a_i + 2t < f + t$. Thus we see that the auction has an allocation $(S^c : S)$ with value $f + t$ if the knapsack problem has a positive answer S, and otherwise the allocation has a lower value. $\square$

5.3 A 2-approximation

Our main algorithmic result is a 2-approximation algorithm for combinatorial auctions in which all valuations are submodular. The result does not rely on any specific representation of submodular functions: it only assumes that one can effectively compute the values of singletons and the marginal valuations of a given valuation.

Input: $v_1, \ldots, v_n$ - submodular valuations, given as black boxes.

Output: An allocation (partition of the items) $S_1, \ldots, S_n$ which is a 2-approximation to the optimal allocation

Algorithm

1. Set $S_1 = S_2 = ... = S_n \leftarrow \emptyset$.

2. For $x = 1...m$ do:

   (a) Let j be the bidder with highest value of $v_j(x|S_j)$.

   (b) Allocate x to bidder j, i.e. $S_j \leftarrow S_j \cup \{x\}$.

The algorithm obviously requires only a polynomial number of operations and calls to valuation oracles for v_j since $v_j(x|S_j) = v_j(S_j \cup \{x\}) - v_j(S_j)$.

THEOREM 8. *The greedy algorithm above provides a 2-approximation to the optimal one.*

PROOF. Let Q be the original problem and define Q' to be the problem on the $m-1$ remaining items after item 1 is removed: i.e., item 1 is unavailable and v_j is replaced by v'_j with $v'_j(S) = v(S|\{1\}) = v(S \cup \{1\}) - v(\{1\})$, where j is the

[5] This observation was independently made by Rakesh Vohra.

player to which item 1 was allocated. All other valuations v_i, $i \neq j$ are unchanged. Notice that the algorithm above may be viewed as first allocating item 1 to j and then allocating the other elements using a recursive call on Q'.

Let us denote by $ALG(Q)$ the value of the allocation produced by this algorithm, and by $OPT(Q)$ the value of the optimal allocation. Let $p = v_j(\{1\})$. By the definition of Q', it is clear that $ALG(Q) = ALG(Q') + p$. We will now show that $OPT(Q) \leq OPT(Q') + 2p$. Let $S_1, \ldots, S_n$ be the allocation optimal for Q, and assume that $i \in S_k$, i.e., item 1 is allocated to bidder k by the algorithm above. Let S' be the allocation of items $2, \ldots, m$ that is similar to S. This is a possible solution to Q'. Let us compute its value, by comparing it to $OPT(Q)$. All players except k get the same allocation and all players except j have the same valuation. Without loss of generality, assume $k \neq j$. Player k looses at most $v_k(\{1\})$ since v_k is submodular. But $v_k(\{1\}) \leq v_j(\{1\}) = p$ and player k looses at most p. Player j looses at most p since, by monotonicity of v_j, $v'_j(S_j) = v_j(S_j \cup \{1\}) - v_j(\{1\}) \geq v_j(S_j) - p$. Therefore $OPT(Q') \geq OPT(Q) - 2\,p$. The proof is concluded by induction on Q' since, by lemma 1, Q' also consists of submodular valuations:

$$OPT(Q) \leq OPT(Q') + 2\,p \leq$$

$$2\,ALG(Q') + 2\,p = 2\,ALG(Q).$$

□

Looking at the proof one may see that many variants of the above algorithm also produce a 2-approximation. In particular the items x may be enumerated in any order in the outer loop. One may think of several heuristics for choosing the next x. For example: take the item x that maximizes the difference between the first and second values of $v_j(x|S_j)$.

It is easy to see that the algorithm above may provide only a 2-approximation even for submodular valuations: take $v_1(\{1\}) = v_1(\{2\}) = v_1(\{1,2\}) = v_2(\{1\}) = v_2(\{1,2\}) = 1$ and $v_2(\{2\}) = 0$.

Open Problem: Does any polynomial algorithm provide a better approximation ratio?

The results of [16] show that achieving an approximation ratio of better than $1+1/m$ requires an exponential number of queries to the valuation oracles.

5.4 Strategic considerations

The approximation algorithm presented above may be viewed as a simple sequence of auctions of the single items, one by one. Simplistic bidders whose strategies do not take the future into account should indeed value an item i at its current marginal value for the bidder. Thus if each auction in this sequence is designed to be incentive compatible (e.g. a second price auction) and if all bidders follow this simplistic bidding strategy (myopic bidders) then the desired 2-approximate allocation would be obtained. In this subsection we ask whether there is a payment scheme to be used with this allocation algorithm that guarantees incentive-compatibility of the complete auction, i.e. that will reach this allocation when the bidders are rational and not myopic. The answer is negative.

An example will show that no payment scheme can make the greedy allocation scheme for sub-modular combinatorial auctions a truthful mechanism. In this example, the greedy scheme allocates the most expensive items first. We do not know whether the result may be generalized to any greedy scheme. The spirit of the proof is similar to that of Section 12 in [11].

EXAMPLE 6. *Two goods: a and b. Two sub-modular players: Red and Green. Red declares 10 for a, 6 for b and 11 for the set $\{a, b\}$. Notice this is a sub-modular declaration, but this is not crucial. Green has two personalities:*

- *Green1 declares 11 for a, 10 for b and 18 for the set $\{a, b\}$.*
- *Green2 declares 9 for a, 10 for b and 19 for the set $\{a, b\}$.*

Notice both are sub-modular declarations.

The allocation between Red and Green1 goes in the following way: a is allocated first to Green1 (Green's 11 vs. Red's 10) and then b is allocated to Green1 (Green's 18-11 vs. Red's 6). Notice that if b had been allocated first Green would have obtained b but not a. Green1 is therefore allocated the set $\{a, b\}$ and pays a sum p. Notice that, since all declarations are fixed, p is a number; it does not depend on anything. The allocation between Red and Green2 goes a different way: any ordering gives a to Red and b to Green. Green2 pays q (just a number).

If the mechanism is truthful and Green is Green1, it must be the case that Green cannot gain by disguising himself as Green2: $18 - p \geq 10 - q$. If the mechanism is truthful and Green is Green2, it must be the case that Green cannot gain by disguising himself as Green1: $10 - q \geq 19 - p$. A contradiction.

6. FALSE-NAME BIDS

We now turn to consider a Generalized Vickrey Auction among submodular players. In [18], a serious problem with Generalized Vickrey Auctions has been put in evidence: a player may benefit by sending two straw players to bid on his behalf. The reader should consult [18, 25, 26, 27] on this problem. We shall show that no such problem exists if the combined valuation of all other players is submodular. We then discuss this condition and show that the result is sharp.

THEOREM 9. *A player cannot benefit from placing false name bids in a combinatorial auction using the VCG rules, whenever the combined valuation of all other players is submodular.*

PROOF. We shall show that two players, one of them bidding a valuation v_1 and the other a valuation v_2 will pay *no less* than a single player bidding $v_1 \vee v_2$, as long as the combined valuation function of the other players is submodular.

Consider the set of bids $v_1, v_2, u_1, \ldots, u_m$ and let S_1 be the set allocated to player 1, S_2 be the set allocated to player 2, and T be the set allocated to all the u_i bidders together: thus the total set of goods is $T \cup S_1 \cup S_2$. Let $u = u_1 \vee \ldots \vee u_m$.

The VCG rules (GVA auction) specify that player 1 will pay:

$$(u \vee v_2)(T \cup S_1 \cup S_2) - (u \vee v_2)(T \cup S_2).$$

Since the optimal allocation of $T \cup S_2$ among the valuations u and v_2 allocates T to u and S_2 to v_2, we have that

$(u \vee v_2)(T \cup S_2) = u(T) + v_2(S_2)$. By considering the allocation of S_2 to v_2 and $T \cup S_1$ to u, we can bound

$$(u \vee v_2)(T \cup S_1 \cup S_2) \geq u(T \cup S_1) + v_2(T_2).$$

We thus get that player 1 pays at least $u(T \cup S_1) - u(T)$. Similarly player 2 will pay at least $u(T \cup S_2) - u(T)$.

Consider, on the other hand what happens when instead of v_1 and v_2 a single $v_1 \vee v_2$ bid is submitted. The allocation to this bidder is exactly $S_1 + S_2$. The VCG payment of this player will be $u(T \cup S_1 \cup S_2) - u(T)$. The submodularity of u directly implies that this is less than or equal to the sum of the payments of players 1 and 2 in the previous case. □

COROLLARY 3. *A player cannot benefit from placing false name bids in a combinatorial auction using the VCG rules, where all valuations satisfy the gross substitutes property.*

PROOF. According to theorem 5, the combined valuation (OR) of GS valuations is GS and thus is also submodular. □

This in particular implies the result of [18] showing that in a Vickrey multi-unit auction of identical items in which all players have a downward sloping valuations, no player could benefit from placing false-name bids.

Could Theorem 9 assume, instead, that all players have a submodular valuation? The following example shows that this is not the case. The following example builds on Example 5 and presents an auction in which each player is submodular, but the combined valuation of the opponents is not and one can benefit from false name bids.

EXAMPLE 7. *Red has budget limit 6 and values: $red(a) = red(c) = 3$, $red(b) = 5$. Blue has unbounded budget: $blue(a) = 1$, $blue(b) = 2$ $blue(c) = 0$. Let v be the combined valuation, $v = red \vee blue$. We have that $v(abc) = 8$, $v(ab) = v(ac) = v(bc) = 6$. Green has unbounded budget: $green(a) = 2$, $green(b) = 0$, $green(c) = 5$.*

If Green acts as himself, he gets $\{a, c\}$ (utility 7) and Red gets $\{b\}$ (utility 5) for a total of 12, unbeatable. Green pays: 8-5 = 3.

But, assume that Green acts under two different identities $G1(a) = 2$ and $G2(c) = 5$. Obviously $G1$ gets $\{a\}$ and $G2$ gets $\{c\}$. $G1$ pays: 11-10=1. $G2$ pays: 8-7=1. On the whole $G1$ and $G2$ together pay only 2 which is strictly less than 3.

7. BOUNDED COMPLEMENTARITY

This section generalizes some of our results to classes of valuations outside CF. The definitions and results of Section 2.4 can be generalized to deal with valuations that exhibit a limited amount of complementarity. We assume $a \geq 1$ is a real number.

DEFINITION 13. *A valuation v is said to exhibit a-bounded complementarities if, for any set A and item x:*

$$v(A \cup \{x\}) \leq v(A) + a\, v(\{x\}).$$

DEFINITION 14. *A valuation v is said to be a-modular if and only if for every subset W of items, the valuation v_W exhibits a-bounded complementarities.*

The proofs of the following results are similar to those of Section 2.4.

PROPOSITION 3. *A valuation v is a-modular if and only if one of the following equivalent propositions holds.*

- *For any $x \in X$ and $S, T \subseteq X$, such that $S \subseteq T$ and $x \notin T$: $v_S(x) \geq a\, v_T(x)$.*
- *For any $S, T, V \subseteq X$, such that $S \subseteq T$: $v_S(V) \geq a\, v_T(V)$.*
- *For any $A, B \subseteq X$: $v(A) + a\, v(B) \geq v(A \cup B) + a\, v(A \cap B)$.*

The 2-approximation result of Section 5 generalizes to a $1 + a$-approximation.

THEOREM 10. *The greedy algorithm provides a $(1 + a)$ approximation to the optimal one if all valuations are a-submodular.*

8. ACKNOWLEDGEMENTS

We thank Sushil Bikhchandani, Moti Perry, Rakesh Vohra, and Makoto Yokoo for helpful discussions, and Liad Blumrosen for comments on an earlier draft of this manuscript.

9. REFERENCES

[1] Lawrence M. Ausubel. An efficient dynamic auction for heterogeneous commodities. first draft: 5 July 2000, 8 September 2000.

[2] Sven de Vries and Rakesh Vohra. Combinatorial auctions: A survey. http://www.kellogg.nwu.edu/faculty/vohra/htm/res.htm, January 2001.

[3] Michael L. Fredman and Robert Endre Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 34(3):596–615, July 1987.

[4] Yuzo Fujishima, Kevin Leyton-Brown, and Yoav Shoham. Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In *Proceedings of IJCAI'99*, Stockholm, Sweden, July 1999. Morgan Kaufmann.

[5] M. R. Garey and D. S. Johnson. *Computers and Intractability: a Guide to the Theory of NP-completeness*. Freeman, San Francisco, 1979.

[6] Rica Gonen and Daniel Lehmann. Optimal solutions for multi-unit combinatorial auctions: Branch and bound heuristics. In *Second ACM Conference on Electronic Commerce (EC-00)*, pages 13–20, Minneapolis, Minnesota, October 2000.

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

[8] F. Gul and E. Stacchetti. The english auction with differentiated commodities. *Journal of Economic Theory*, 92:66–95, 2000.

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

[10] Daniel Lehmann, Liadan Ita O'Callaghan, and Yoav Shoham. Truth revelation in rapid, approximately efficient combinatorial auctions. In *Proceedings of the First ACM Conference on Electronic Commerce. EC'99*, pages 96–102, Denver, Colorado, November 1999. SIGecom, ACM Press.

[11] Daniel Lehmann, Liadan Ita O'Callaghan, and Yoav Shoham. Truth revelation in rapid, approximately efficient combinatorial auctions. Technical note STAN-CS-TN-99-88, Stanford University, Computer Science, Robotics Lab., June 1999.

[12] Jeffrey K. MacKie-Mason and Hal R. Varian. Generalized Vickrey auctions. Working paper, Un. of Michigan, July 1994.

[13] H. Moulin. *Axioms of Cooperative Decision Making.* Cambridge University Press, Cambridge, U.K., 1988.

[14] H. Narayanan. *Submodular Functions and Electrical Networks*, volume 54 of *Annals of Discrete Mathematics.* North-Holland, 1997.

[15] Noam Nisan. Bidding and allocation in combinatorial auctions. In *Proceedings of the 2nd ACM Conference on Electronic Commerce EC'00*, pages 1–12, Minneapolis, Minnesota, October 2000. ACM Press.

[16] Noam Nisan. The Communication Complexity of Combinatorial Auctions. Preprint, 2001. Available from http://www.cs.huji.ac.il/˜noam/mkts.html.

[17] Michael H. Rothkopf, Aleksandar Pekeč, and Ronald M. Harstad. Computationally manageable combinatorial auctions. *Management Science*, 44(8):1131–1147, 1998.

[18] Yuko Sakurai, Makoto Yokoo, and Sigeo Matsubara. A limitation of the generalized vickrey auction in electronic commerce : Robustness against false-name bids. In *Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, pages 86–92, 1999.

[19] T. Sandholm and S. Suri. Improved algorithms for optimal winner determination in combinatorial auctions and generalizations. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Austin, TX, July 1999.

[20] Tuomas Sandholm. An algorithm for optimal winner determination in combinatorial auctions. In *IJCAI-99*, pages 542–547, Stockholm, Sweden, July 1999.

[21] Robert Endre Tarjan. *Data Structures and Network Algorithms.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1983.

[22] Moshe Tennenholtz. Some tractable combinatorial auctions. Unpublished draft: January 2000.

[23] Hal R. Varian. Economic mechanism design for computerized agents. In *Proceedings of the First Usenix Conference on Electronic Commerce*, New York, July 1995.

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

[25] Makoto Yokoo, Yuko Sakurai, and Sigeo Matsubara. The effect of false-name declarations in mechanism design: Towards collective decision making on the internet. In *20th International Conference on Distributed Computing Systems (ICDCS-2000)*, pages 146–153, 2000.

[26] Makoto Yokoo, Yuko Sakurai, and Sigeo Matsubara. Robust combinatorial auction protocol against false-name bids. In *Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 110–116, 2000.

[27] Makoto Yokoo, Yuko Sakurai, and Sigeo Matsubara. Robust double auction protocol against false-name bids. In *21st International Conference on Distributed Computing Systems (ICDCS-2001)*, 2001.