Greedy Mechanism Design for Truthful Combinatorial Auctions

Allan Borodin[1] Brendan Lucier[1]
[1]Dept of Computer Science, University of Toronto
bor@cs.toronto.edu blucier@cs.toronto.edu

Abstract: We study the combinatorial auction problem, in which a set of objects are to be distributed amongst selfish bidders with the goal of maximizing social welfare. We consider mechanisms that apply greedy algorithms, and ask to what extent truthful greedy mechanisms can find approximate solutions to the combinatorial auction problem. We associate the notion of greediness with a broad class of algorithms known as priority algorithms. We show that no truthful priority mechanism obtains a sub-linear approximation ratio. We also consider a particular type of priority algorithm for use with submodular bidders, and show that no such algorithm can be made truthful.

Keywords: Combinatorial Auction; Truthfulness; Greediness; and Priority Algorithms

1 Introduction

The goal of mechanism design is the construction of algorithms for use in situations where inputs are controlled by selfish agents. The difficulty in such a setting is that agents may lie about their inputs in order to obtain a more desirable outcome. It is often possible to circumvent this obstacle by using payments to elicit truthful responses. Indeed, if the goal of the algorithm is to maximize the total welfare of all agents, then the well-known VCG mechanism does precisely that: each agent will maximize their utility by reporting truthfully. However, the VCG mechanism requires that the underlying optimization problem be solved exactly, and is therefore ill-suited for problems that cannot be solved efficiently. As initiated by Nisan and Ronen [34], the field of algorithmic mechanism design attempts to bridge the competing demands of selfish agents with that of algorithmic constraints. We consider this fundamental issue in the context of conceptually simple algorithmic methods (independent of time bounds) rather than in the context of time constrained algorithms. Specifically, we consider how well “greedy or greedy-like” approximation algorithms can be converted into mechanisms for combinatorial auctions.

The combinatorial auction (CA) problem is a well-studied problem. From an algorithmic standpoint, the state of general CAs is largely resolved. One can obtain an $O(\min\{n, \sqrt{m}\})$ approximation for CAs with n bidders and m objects with a relatively simple greedy algorithm [28]. This is essentially the best result possible, since (by a reduction from the max clique problem) approximating to within a factor of $O(\min\{n^{1-\epsilon}, m^{\frac{1}{2}-\epsilon}\})$ for any $\epsilon > 0$ is known to be hard [18] (subject to the complexity assumption that $NP \neq ZPP$) even for single minded bidders. The situation changes, however, when one adds strategic considerations to the analysis. The Lehmann, O’Callahan and Shoham [28] greedy algorithm can be made truthful for single-minded bidders, but it is not incentive compatible given even a single bidder with more complex demands [28]. We note that no deterministic truthful mechanism is known to obtain an approximation ratio better than $O(\frac{m}{\sqrt{\log m}})$ [4]. Indeed, even for certain restricted auctions such a s-CAs (in which desired sets contain at most s objects), the same $O(\frac{m}{\sqrt{\log m}})$ approximation [4] is the best deterministic truthful mechanism currently known. A lower bound for the combinatorial public project problem [35] shows that there is a gap separating approximation by deterministic algorithms and by deterministic truthful mechanisms in general; whether such a large gap exists specifically for the CA problem is an important open question.

Although the most straightforward greedy algorithms for CAs are not incentive compatible, the notion of greediness still appears often in the mechanism design literature (see, for example, [7, 22, 27, 28, 31]). This is not surprising given that simplicity and determinism are important criteria in mechanism design, since straightforward and transparent algorithms are easy for bidders to understand and trust. We therefore ask whether any deterministic greedy-like *truthful* mechanism for the CA problem can obtain an approximation ratio close to the best-possible $\min\{n, \sqrt{m}\}$. We pursue this question by defining the notion of a "greedy-like" algorithm for the combinatorial auction problem, which we associate with the class of priority algorithms [6]. We define priority algorithms in Sections 2.3 and 3. We show that if an incentive compatible mechanism makes allocations in a "greedy-like" way, then it cannot perform much better than the trivial algorithm that allocates all objects to a single bidder.

Theorem: Any deterministic incentive compatible priority algorithm (of a certain type described in the text) for the combinatorial auction problem cannot obtain a $o(\min\{m, n\})$-approximation to the optimal social welfare.

The class of algorithms we consider is independent of the manner in which valuation functions are accessed. In particular, our results apply to algorithms in the demand query model and the general query model, as well as to auctions in which bids are explicitly represented. Also, in addition to greedy algorithms, our class captures some types of online algorithms. We note, however, that the priority algorithm framework is considerably more general than many online mechanism design settings that have been studied previously [25, 32], as discussed in Section 1.1.

We also consider the combinatorial auction problem for submodular bidders (SMCA), which has been the focus of much study [11, 12, 20, 27]. We study a class of greedy algorithms that is especially well-suited to the SMCA problem. Such algorithms consider the objects of the auction one at a time and greedily assign them to bidders to maximize marginal utilities. It was shown in [27] that any such algorithm attains a 2-approximation to the SMCA problem, but that not all are incentive compatible. The question of whether any such algorithm is incentive compatible was left open. We resolve this question negatively.

Theorem: Any deterministic algorithm for submodular combinatorial auctions that considers objects and assigns them in order to maximize marginal utility cannot obtain a bounded approximation to the optimal social welfare.

1.1 Related Work

Following the Lehmann et al [28] truthful greedy mechanism for single minded CAs, Mu'alem and Nisan [31] showed that any *monotone* greedy algorithm for single minded bidders is incentive compatible using critical prices. However, as previously noted, these results are provably limited to single minded bidders. For arbitrary CAs, there exists a randomized mechanism (based on randomized rounding of LP solutions) that obtains an approximation ratio of $O(\sqrt{m})$ and is truthful in expectation [26], and also a $O(\sqrt{m})$ approximate randomized mechanism (based on sampling the values of some agents in order to fix prices for other agents) that satisfies the stronger notion of universal truthfulness [10]. These papers motivated our interest in trying to understand when greedy algorithms can be made into deterministic truthful mechanisms for auctions which are not single minded.

Various alternatives to truthfulness have been considered previously for auction problems (see [9]). These include the concepts of ϵ-truthfulness [21] and truthfulness in undominated strategies [1].

In contrast to the negative results of this paper, greedy algorithms can provide good approximations in the context of Nash equilibria. In particular, the Nash equilibrium solution concept has been applied to submodular combinatorial auctions [8]. Recently, we have shown [30] that in a wide variety of contexts, c-approximate monotone greedy allocations can be made into mechanisms whose Nash equilibria all enjoy (nearly) c approximations; that is, monotone greedy allocations with appropriate payment rules yield price of anarchy results close to or equal to the greedy allocation approximation bound. Performance at pure Nash equilibria has also been previously considered for other types of auctions, most notably related to Internet advertising slots [14, 36]. In that line of work the goal is revenue maximization, rather than social welfare maximization, and the auctions considered are special cases of the more general combinatorial auctions. The weaker notion of "Set-Nash" equilibrium has been used to analyze online mechanisms [25], though this is a very different setting than that of a general CA mechanism.

We are not the first to prove lower bounds for allocation problems relating to CA mechanisms. Gonen and Lehmann [17] showed that no algorithm that considers bids for sets one at a time and accepts them greedily can guarantee an approximation better than $\sqrt{m}$. A similar bound was derived by Krysta [22], who showed that no oblivious greedy algorithm (in our terminology: fixed order greedy priority algorithm) can obtain an approximation ratio better than $\sqrt{m}$. (In fact, Krysta derives this bound for a more general class of problems that includes multi-unit CAs.) In contrast to these results, we consider the more general class of priority algorithms but restrict them to be incentive-compatible, and obtain a stronger bound.

Theorem 5 of [23] shows that any truthful CA mechanism that uses a suitable bidding language, is unanimity-respecting, and satisfies the independence of irrelevant alternatives property (IIA) cannot attain a polynomial approximation ratio. Our result is incomparable, as priority algorithms need not be unanimity-respecting or satisfy IIA[1]. It has also been shown that, roughly speaking, no truthful polytime subadditive combinatorial auction mechanism that is *stable*[2] can obtain an approximation ratio better than 2 [13]. This is incomparable to our results, as priority algorithms need not be stable.

The class of priority algorithms is loosely related to the notion of online algorithms. Mechanism design has been studied in a number of online settings, and lower bounds are known for the performance of truthful algorithms in these settings [25, 32]. The critical difference between these results and the lower bounds we present for priority algorithms is that a priority algorithm has a great deal of control over the order in which input items are considered, whereas in an online setting this order can be chosen adversarially. Thus our lower bounds demonstrate that truthful mechanisms with non-trivial approximation ratios cannot proceed by considering bidders or bids for sets one at a time, even if they have adaptive control over the order in which the bidders (or sets, or objects, depending on the model) are considered, as long as each ordering can be expressed as a total order over the set of all possible input items.

2 Defintions and Preliminary Results

2.1 Combinatorial Auctions

A *combinatorial auction* consists of a set N of bidders and a set M of objects. Let $m = |M|$ and $n = |N|$. Each bidder i has a value for each subset of objects $S \subseteq M$, say $v_i(S)$. These values have the following properties: $v_i(\emptyset) = 0$, $v_i(S) \geq 0$ for all $S \subseteq M$, and $v_i(S) \leq v_i(T)$ for all $S \subseteq T \subseteq M$. We will think of v_i as a valuation function $v_i : 2^M \to \mathbb{R}$. We denote by V_i the space of all possible valuation functions for agent i. The entire valuation space is denoted $V = V_1 \times V_2 \times \cdots \times V_n$. Given $i \in N$ and $v \in V$, we will often write $v_{-i} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$, so that $v = (v_i, v_{-i})$.

A valuation function v is *single-minded* if there exists a set $S \subseteq M$ and a value $x \geq 0$ such that, for all $T \subseteq M$,

$$v(T) = \begin{cases} x & \text{if } S \subseteq T \\ 0 & \text{otherwise.} \end{cases}$$

A valuation function v is *k-minded* if it is the maximum of k single-minded functions. That is, there exist k sets $S_1, \ldots, S_k$ such that $v(S_i)$ is given for each S_i, and for all subsets $T \subseteq M$ we have $v(T) = \max\{v(S_i) | S_i \subseteq T\}$. An *additive* valuation function v is specified by m values $x_1, \ldots, x_m \in \mathbb{R}_{\geq 0}$ so that $v(T) = \sum_{a_i \in T} x_i$. A valuation function v is *submodular* if it satisfies $v(T) + v(S) \geq v(S \cup T) + v(S \cap T)$ for all $S, T \subseteq M$. The submodular valuation functions are also characterized by the property of decreasing marginal utilities: for all $S \subseteq T$ and $x \in M$, $v(S \cup \{x\}) - v(S) \geq v(T \cup \{x\}) - v(T)$.

---

[1] The notion of IIA has been associated with priority algorithms, but in a different context than in [23]. In mechanism design IIA is a property of the mapping between input valuations and output allocations, whereas for priority algorithms the term IIA describes restrictions on the order in which input items can be considered.

[2] In a stable mechanism, no player can alter the outcome (ie. by changing his declaration) without causing his own allocated set to change.

A *direct revelation mechanism* $\mathcal{M}$ for a combinatorial auction is composed of two parts: an *allocation algorithm* G and a *payment algorithm* P. Given $v \in V$, $G(v)$ returns an allocation of objects to bidders, and $P(v)$ returns the payment extracted from each agent for the set of objects they receive. For each $i \in N$ we write $G_i(v)$ and $P_i(v)$ for the set given to and payment taken from bidder i. We require that the allocation given by $\mathcal{M}$ is *valid* for all $v \in V$, meaning that $G_i(v) \cap G_j(v) = \emptyset$ for all $i \neq j$. We will refer to a direct revelation mechanism as a *mechanism* (due to the well-known revelation principle).

Fix a mechanism $\mathcal{M}$ and valuations $t \in V$ representing the true types of the agents. Then for any $S \subseteq M$, $t_i(S)$ represents the *value* of set S to bidder i. Given any $v \in V$, the *utility* of bidder i given bids v is given by $u_i(v) = t_i(G_i(v)) - P_i(v)$. That is, the utility of bidder i is the value she places on the set she receives, minus the amount she had to pay for it. The *social welfare* obtained by M given bids v is $\sum_{i \in N} t_i(G_i(v))$. The *optimal social welfare* is the maximum of $\sum_{i \in N} t_i(S_i)$ over all valid allocations $(S_1, \ldots, S_n)$. The goal of a mechanism for the combinatorial auction problem is to maximize the social welfare.

A mechanism $\mathcal{M}$ is *truthful* if for every $i \in N$, $v_{-i} \in V_{-i}$, and $t_i, v'_i \in V_i$, $t_i(G_i(t_i, v_{-i})) - P_i(t_i, v_{-i}) \geq t_i(G_i(v'_i, v_{-i})) - P_i(v'_i, v_{-i})$. That is, if agent i's true valuation is t_i, she would not obtain more utility by instead reporting a different valuation v'_i. An allocation function G is *incentive compatible* if there exists a payment function P such that the mechanism (G, P) is truthful.

2.2 Critical Prices

An allocation algorithm G defines *critical prices* $p_j(S, v_{-j})$ for any agent j and set S where $p_j(S, v_{-j})$ is the minimum amount that agent j could bid on set S and still win S assuming the other agents bid according to v_{-j}. From Bartal, Gonen and Nisan [3], we have the following characterization of truthful mechanisms for combinatorial auctions, which we will use throughout our proofs.

Theorem 2.1. *A mechanism is truthful if and only if for every bidder j and every vector of bids of the other bidders v_{-j} it:*

1. *fixes the critical price $p_j(S, v_{-j})$ for every possible allocation S to bidder j, and whenever bidder j is allocated S his payment is $p_j(S, v_{-j})$ (note that $p_j(S, v_{-j})$ does not depend on v_j), and*
2. *allocates to j a set S that maximizes the value of $v_j(S) - p_j(S, v_{-j})$ over all S that can be allocated to j (for any choice of v_j).*

Note that $p_j(S, v_{-j})$ from the statement of Theorem 2.1 need not be finite: if $p_j(S, v_{-j}) = \infty$ then the mechanism will not allocate S to bidder j for any reported valuation v_j. However, it is easy to show that one can assume critical prices are monotonic.

Claim 2.2. *Suppose that a mechanism satisfies the conditions of Theorem 2.1. Then one can assume without loss of generality that for all $j \in N$, all $v_{-j} \in V_{-j}$, and all $S \subseteq T \subseteq M$, $p_j(S, v_{-j}) \leq p_j(T, v_{-j})$.*

2.3 Priority Algorithms

We now provide the definition of a priority algorithm [6]. We view an input instance to an algorithm as a set of *input items* from a known input space $\mathcal{I}$. It is important to note that $\mathcal{I}$ is the set of all possible input items; an input instance is a finite subset of $\mathcal{I}$. The definition of the set $\mathcal{I}$ will depend on the context of the problem being considered. The problem definition may also place restrictions on the nature of an input instance; we will say that an input instance $I \subseteq \mathcal{I}$ is *valid* if it satisfies all such restrictions.

Once the input to an algorithm has been specified in the above way, one can view the output of the algorithm as a decision made for each input item. For example, these decisions may be of the form "accept/reject", schedule job i on machine j, allocate set S to agent i, etc. For an optimization problem the goal of the algorithm is to optimize some function of the output space. As with input instances, the problem definition may place restrictions on the nature of the decisions made by the algorithm; we say that the output of the algorithm is *valid* if it satisfies all such restrictions. We then define the class of priority algorithms as follows:

```
ADAPTIVE PRIORITY
**Input:** A set $I$ of items, $I \subseteq \mathcal{I}$
while not empty($I$)
    **Ordering:** Choose, without looking at $I$,
```

```
        a total ordering $\mathcal{T}$ over $\mathcal{I}$
    next := first item in $I$ according to ordering $\mathcal{T}$
    Decision: make a decision for item next;
    remove next from $I$;
    remove from $\mathcal{I}$ any items
        preceding next in $\mathcal{T}$
end while
```

We emphasize the importance of the ordering step in this framework: an adaptive priority algorithm is free to choose *any* total ordering over the space of possible input items, and moreover can change this ordering adaptively after each input item is considered. When an input item is considered, the priority algorithm must then make an irrovable decision for this item. Once an item has been processed, the algorithm is not permitted to revisit that item and modify its decision. Note also that on each iteration a priority algorithm learns what (higher-priority) items are *not* in the input. A special case of (adaptive) priority algorithms are fixed order priority algorithms in which one fixed ordering is chosen before the while loop (i.e. the "ordering" and "while" statements are interchanged). Our inapproximation results for truthful CAs will hold for the more general class of adaptive priority algorithms but we note that many greedy CA algorithms are fixed order.

The term "greedy" implies a more "opportunistic" aspect (i.e. restrcition) than is mandated in the definition of priority algorithms and indeed we view priority algorithms more generally as "greedy-like". A *greedy* priority algorithm satisfies an additional property: the choice made for each input item must optimize the objective of the algorithm as though that item were the last item in the input. In Section 3.2 we will consider greedy priority algorithms when we consider "sets" (being bid upon) as input items.

3 Truthful Priority Algorithms

As we already noted, Lehmann, O'Callahan and Shoham show that a $O(\sqrt{m})$-approximation for combinatorial auctions can be made truthful (using critical pricing) for single-minded bidders but is not incentive compatible (i.e. cannot be made truthful for any payment rule) for more general bidders. We wish to show that no greedy-like mechanism (i.e. within the priority framework) can be both incentive compatible and provide a "good" approximation to the social welfare. In order to apply the concept of priority algorithms to combinatorial auctions we must define the set $\mathcal{I}$ of possible input items and the nature of the decisions. We will show that for two natural input formulations, no mechanism using a priority algorithm allocation can approximate the optimal social welfare significantly better than the trivial mechanism that allocates all objects to a single bidder. We can assume that n, the number of bidders, and m, the number of objects are known to the mechanism and let $k = \min\{m, n\}$.

3.1 Bidders as Items

Under this model, $\mathcal{I}$ consists of all pairs (i, v_i), where $i \in N$ and $v_i \in V_i$. A valid input instance contains one item for each bidder. In this case, processing an item (i, v_i) involves choosing a set $S \subseteq M$ to assign to bidder i. We note that the truthful CA mechanism for single-minded bidders falls within this model, as does its (non-truthful) generalization to complex bidders [28]. The primal-dual schema algorithm of [7] and the generic algorithm of [3] for multi unit CAs also fall within this model.

Let us recall the greedy mechanism in [28] and how it becomes a priority algorithm with bidders as items. In this algorithm, sets are allocated greedily according to their rank, where the rank $r_i(S)$ of set S for bidder i is defined as $\frac{v_i(S)}{\sqrt{|S|}}$. The algorithm begins by ordering the bidders by maximum rank over all sets. The algorithm is then presented with the first bidder in this ordering; it will allocate to that bidder the set with the largest rank. The algorithm then adaptively reorders the remaining bidders by maximum rank for any subset *of the remaining items*. The first bidder in this ordering is considered and given the subset of the remaining items with the largest rank. The algorithm continues in this way until either all objects are allocated or each bidder has been considered.

We now establish an inapproximation bound for truthful priority allocations using bidders as items. The approach to proving such a result is to consider two types of bidders: the first type (a "poor" bidder) has a low value for any set, and the second type (a "wealthy" bidder) has a very high value for some singleton and a slightly higher value for the set of all objects . We tailor the wealthy valuations so that, if all bidders are poor except one, then the single wealthy bidder must be given all objects due to incentive compatibility. On the other hand, if a linear number of bidders are wealthy, then the algorithm must give singletons to many wealthy bidders to obtain a good approximation to the social welfare. The algorithm is forced to make an allocation to some bidder before being able to distinguish between these two cases.

Theorem 3.1. *Any incentive compatible priority algorithm that uses bidders as items cannot obtain a* $\frac{(1-\delta)k}{2}$*-approximation to the optimal social welfare for any* $\delta > 0$*. This inapproximation applies even if all bidders have 2-minded valuation functions.*

Proof. Choose $\delta > 0$ and suppose for contradiction that A is an incentive compatible adaptive priority algorithm that achieves an approximation ratio of $k(1-\delta)/2$. We will denote an item as a tuple (i, v_i), where $1 \leq i \leq n$ is a bidder and $v_i : 2^M \to \mathbb{R}$ is a valuation function.

We will construct a set of input instances for which A is forced to make a particular allocation, due to incentive compatibility. We define two sets of valuation functions, $\{g_1, \ldots, g_k\}$ and $\{f_1, \ldots, f_k\}$, that will be used in these input instances. The functions $g_1, \ldots, g_k$ are straightforward: for each $1 \leq i \leq k$, define valuation function g_i by

$$g_i(S) = \begin{cases} 1 & \text{if } a_i \in S \\ 0 & \text{otherwise.} \end{cases}$$

Then g_i is a single-minded valuation function, where the desired set is $\{a_i\}$ with value 1.

The definition of valuation functions $f_1, \ldots, f_k$ is more involved. Fix $i \in N$ and define $V'_{-i} := \{g_1, \ldots, g_k\}^{n-1}$. Consider an instance (v_i, v_{-i}) of the combinatorial auction problem in which $v_{-i} \in V'_{-i}$. That is, each bidder $j \neq i$ is single-minded, and desires a singleton with value 1. By the critical price property, there is a critical price $p_i(M, v_{-i})$ for set M given this v_{-i}.

Claim 3.2. *Under the above definitions,* $p_i(M, v_{-i}) \leq kn$*.*

We are now ready to define the valuations $f_1, \ldots, f_k$. They are based on values $x, y \in \mathbb{R}$. Define $x \in \mathbb{R}$ as follows:

$$x := 1 + \max_{i \in N} \max_{v_{-i} \in V'_{-i}} \{p_i(M, v_{-i})\}.$$

That is, x is a value greater than the maximum of the critical price for M for bidder i, over all choices of i and desires of singletons with value 1 by other bidders. Set $y := x\delta^{-1}$.

For each $1 \leq i \leq k$, define valuation function f_i as

$$f_i(S) = \begin{cases} y & \text{if } \{a_i\} \subseteq S \subset M \\ y + x & \text{if } S = M \\ 0 & \text{otherwise.} \end{cases}$$

Then $f_i(S)$ is a 2-minded valuation function. We now consider the following subset $\mathcal{I}' \subseteq \mathcal{I}$ of possible input items: $\mathcal{I}'$ contains all bidder-valuation pairs of the form (i, v_i) where $1 \leq i \leq n$ and $v_i = f_j$ or $v_i = g_j$ for some $1 \leq j \leq k$. Note that $\mathcal{I}'$ is not a valid input instance; we think of $\mathcal{I}'$ simply as a subset of $\mathcal{I}$.

The following claim exploits the incentive compatibility of A.

Claim 3.3. *Suppose* $I = \{(1, v_1), \ldots, (n, v_n)\}$ *is a valid input instance, in which there exists* $i \in N$ *such that* $v_i \in \{f_1, \ldots, f_k\}$*, and for all* $j \neq i$, $v_j \in \{g_1, \ldots, g_k\}$*. Then on input* I*,* A *must allocate* M *to bidder* i *and* $\emptyset$ *to all other bidders.*

Our next step is to construct an input instance $I \subseteq \mathcal{I}'$ on which A obtains a poor approximation ratio. To do this we will rely on the following claim which will be proven by induction on i.

Claim 3.4. *There exists a labelling of bidders and objects such that the following is true for all* $0 \leq i < k/2$*. Define* $I_i := \{(j, g_j) | j \leq i\}$*. Then for any valid input instance* I *such that* $I_i \subseteq I \subseteq \mathcal{I}'$*,* A *will consider all the items in* I_i *before all other items in* I*, and will choose to assign* $\emptyset$ *for each of the items in* I_i*.*

Now suppose $I_{k/2-1}$ is the set from Claim 3.4 with $i = k/2 - 1$. Define input instance I by

$$I := I_{k/2-1} \cup \{(j, g_{k/2}) | k/2 \leq j \leq k\}.$$

Note that I is a valid input instance and $I_{k/2-1} \subseteq I \subseteq \mathcal{I}'$. Then by Claim 3.4, algorithm A must assign $\emptyset$ to each of bidders $1, \ldots, k/2 - 1$. Therefore A can obtain a social welfare of at most 1, by assigning $\{a_{k/2}\}$ to some bidder $j \geq k/2$. However, the optimal social welfare is $k/2$, by assigning $\{a_i\}$ to bidder i for all $1 \leq i \leq k/2$. Hence A obtains an approximation no better than $k/2$, which is a contradiction. This completes the proof of Theorem 3.1.

□

3.2 Sets and Bids as Items

Although the use of bidders as items gives the mechanism access to the entire bid vector of a bidder, this input model can be restrictive, as the decisions made by the mechanism must be made about players without being able to gain any knowledge of other players (beyond knowing they have lower priority) not yet considered. An (incomparable) alternative is to associate items with more "elementary bids". Namely, an item would consist of a bidder i, a set S, and $v_i(S)$, the value that bidder i has for set S. In this case $\mathcal{I}$ consists of all triples (i, S, t) such that $i \in N$, $S \subseteq M$, and $t \in \mathbb{R}_{\geq 0}$. A valid input instance $I \subset \mathcal{I}$ contains at most one tuple $(i, S, v_i(S))$ for each $i \in N$ and $S \subseteq M$ and for every pair of tuples (i, S, v) and (i', S', v') in I such that $i = i'$ and $S \subseteq S'$, it must be that $v \leq v'$. We note that as described a valid input instance may contain an exponential number of items, and hence this model applies most directly to algorithms that use oracles to query input valuations, such as demand oracles[3]. However, the model can also apply to succinctly represented auctions. For succinctly (and explicitly) represented auctions, there is a choice as to whether or not the priority algorithm has access to non specified implied bids; that is, when a value $v(S')$ is implied by $\max\{v(S) : S \subseteq S'\}$. We say that the priority model uses *sets as items* when any $(i, S, v(S))$, whether or not explicitly bid upon, will be considered if it is given highest priority by the algorithm. In contrast, we say that the priority algorithm uses *bids as items* when only explicit bids $(i, S.v(S)$ can be considered by the algorithm. In such a model, we will assume that the mechanism is not "wasteful" and will not allocate a larger set when a smaller set will achieve the same value. We consider inapproximations in both of these models.

The decision to be made for item (i, S, t) is whether or not the objects in S should be added to the set of objects to be allocated to bidder i. For example, an allocation algorithm may consider an item (i, S_1, t_1) and decide to allocate S_1 to bidder i, then later consider another item (i, S_2, t_2) (where S_2 and S_1 are not necessarily disjoint) and if feasible decide to allocate S_2 to bidder i as well, and so on. If ℓ items are assigned to bidder i, then the final allocation to bidder i would be $S := S_1 \cup S_2 \cup \ldots \cup S_\ell$. Note that bidder i's value for set S is unambiguous, since a valid input instance would also contain an item of the form (i, S, t).

A *greedy algorithm* in the sets or bids as items model must accept any feasible item (i, S, t) it considers; that is, as long as no objects in S have already been allocated to another bidder and $t > 0$. A priority algorithm in the sets or bids as items model *does not allow incremental updates* if it can accept at most one set for any given bidder. The Lehmann et al [28] greedy allocation can be viewed as a fixed order greedy priority algorithm (using sets or bids as items) that does not allow incremental updates.

Theorem 3.5. *Suppose A is an incentive compatible* greedy *priority algorithm that uses sets as items. Then A cannot approximate the optimal social welfare by a factor of $\frac{(1-\delta)k}{2}$ for any $\delta > 0$. This inapproximation applies to 3-minded auctions where there are at most three sets of value for each bidder.*

Consider the intuition as to why such an algorithm A cannot exist. Suppose some bidder has a relatively very large value for each of two singletons. A good approximation algorithm A would surely allocate one of these singletons to this bidder. Since A is greedy, it must do so without first considering the values held by other bidders for those singletons. However, a truthful algorithm A must also maximize utility, so it must allocate the singleton which has the smaller price. This implies that the relationship between the prices for these

---

[3] It is tempting to assume that this model is equivalent to a value query model, where the mechanism queries bidders for their values for given sets. The priority algorithm model is actually more general, as the mechanism is free to choose an arbitrary ordering over the space of possible set/value combinations. In particular, the mechanism could order the set/value pairs by the utility they would generate under a given set of additive prices, simulating a demand query oracle.

singletons must be independent of their value to other bidders! This is inconsistent, since a singleton desired at a high value by many players must have a higher price than a singleton not desired by any other players, in order to guarantee a good approximation ratio.

Proof. Choose $\delta > 0$ and suppose for contradiction that A does obtain a $(1-\delta)k/2$ approximation to the social welfare. For each $i \in N$, let $V_{-i}^{+} \subseteq V_{-i}$ be the set of valuations with the property that $v_j(S) > 0$ for all $j \neq i$ and all non-empty $S \subseteq M$. The heart of our proof is the following claim, which shows that the relationship between critical prices for singletons for one bidder is independent of the valuations of other bidders. Recall that $p_i(S, v_{-i})$ is the critical value for set S for bidder i, given that other bidders declare valuations v_{-i}.

Claim 3.6. *For all $i \in N$, and for all $a, b \in M$, either $p_i(\{a\}, v_{-i}) \geq p_i(\{b\}, v_{-i})$ for all $v_{-i} \in V_{-i}^{+}$, or $p_i(\{a\}, v_{-i}) \leq p_i(\{b\}, v_{-i})$ for all $v_{-i} \in V_{-i}^{+}$.*

We can think of Claim 3.6 as defining, for each $i \in N$, an ordering over the elements of M. Specifically, for each $i \in N$ we define ordering $\preceq_i$ over M by $a \preceq_i b$ if and only if $p_i(a, v_{-i}) \leq p_i(b, v_{-i})$ for all $v_{-i} \in V_{-1}^{+}$. Note that this ordering may be different for each $i \in N$.

For all $i \in N$ and $a \in M$, define $T_i(a) = \{a_j : a \preceq_i a_j\}$. That is, $T_i(a)$ is the set of objects that are greater or equal to a in ordering $\preceq_i$. Our next claim shows the strong relationship between whether a is allocated to bidder i and whether any object in $T_i(a)$ is allocated to bidder i.

Claim 3.7. *Choose $a \in M$, $i \in N$, and $S \subseteq M$, and suppose $S \cap T_i(a) \neq \emptyset$. Choose some $v_i \in V_i$ and suppose that $v_i(a) > v_i(S)$. Then if $v_{-i} \in V_{-i}^{+}$, bidder i cannot be allocated set S by algorithm A given input v.*

Proof. We know that $p_i(S, v_{-i}) \geq p_i(a_j, v_{-i})$ for any $a_j \in S$ since by Claim 2.2 we can assume critical payments are monotone. Thus, regardless of the choice of v_{-i},

$$p_i(S) \geq \max_{a_j \in S \cap T_i(a)} (p_i(a_j)) \geq p_i(a)$$

from the definition of $T_i(a)$. Since $v_i(a) > v_i(S)$, this implies that $v_i(a) - p_i(a) > v_i(S) - p_i(S)$, so by Theorem 2.1 bidder i cannot be allocated set S, as required. □

Claim 3.7 is strongest when $T_i(a)$ is large; that is, when a is "small" in the ordering $\preceq_i$. We therefore wish to find an object of M that is small according to many of these orderings, simultaneously. Let $S(a) = \{i \in N : |T_i(a)| \geq k/2\}$. Then $S(a)$ is the set of players for which there are at least $k/2$ objects greater than a. The following claim follows by a straightforward counting argument.

Claim 3.8. *There exists $a^* \in M$ such that $|S(a^*)| \geq k/2$.*

We are now ready to proceed with the proof of Theorem 3.5. Let $a^* \in M$ be the object from Claim 3.8. Let $\epsilon > 0$ be a sufficiently small value to be defined later. We now define a particular input instance to algorithm A. For each $i \in S(a^*)$, bidder i will declare the following valuation function, v_i:

$$v_i(S) = \begin{cases} 1 & \text{if } a^* \in S \\ 1 - \delta/2 & \text{if } a^* \notin S \text{ and } S \cap (T_i(a^*)) \neq \emptyset \\ \epsilon & \text{otherwise.} \end{cases}$$

Each bidder $i \notin S(a^*)$ will declare a value of ϵ for every set.

For each $i \in S(a^*)$, bidder i obtains a value of at least $1 - \delta/2$ on set $\{a_j\}$ for every $a_j \in T_i(a^*)$ (including a^*). Since $S(a^*) \geq k/2$ and $T_i(a^*) \geq k/2$, we have at least $k/2$ bidders obtaining a value of at least $1-\delta/2$ on each of at least $k/2$ singletons. It is therefore possible to obtain a social welfare of at least $(1-\delta/2)k/2$ by allocating singletons to bidders in $S(a^*)$.

Now consider the social welfare obtained by algorithm A. The algorithm can allocate object a^* to at most one bidder, say bidder j, who will obtain a social welfare of at most 1. For any bidder $i \in S(a^*)$, $i \neq j$, $v_i(S) = 1 - \delta/2 < 1$ for any S containing elements of $T_i(a^*)$ but not a^*. Thus, by Claim 3.7, no bidder in $S(a^*)$ can be allocated any set S that contains an element of $T_i(a^*)$ but not a^*. Therefore every bidder other than bidder j can obtain a value of at most ϵ, for a total social welfare of at most $1 + k\epsilon$.

We conclude that algorithm A has an approximation factor of at least $\frac{k(1-\delta/2)}{2(1+k\epsilon)}$. Choosing $\epsilon < \frac{\delta}{2(1-\delta)k}$ yields an approximation greater than $\frac{k(1-\delta/2}{2}$, giving the desired contradiction to complete the proof of Theorem 3.5. $\square$

We conjecture that the greediness assumption of Theorem 3.5 can be removed. What we will show is that the greediness assumption can be removed for priority algorithms in the bids as items model that do not allow incremental allocations. We believe that this restriction can be removed.

Theorem 3.9. *Suppose A is an incentive compatible priority algorithm that uses bids as items and does not allow incremental updates. This inapproximation applies in the case of 2-minded auctions.*

Proof. Suppose A is a truthful adaptive priority algorithm, where the items to be considered are associated with bids. That is, an item is a tuple (i, S, t) where $v_i(S) = t$. On processing each item, the algorithm must decide whether S will be the set allocated to bidder i and if S is allocated to i then that will be the final allocation to bidder i. Suppose for contradiction that A obtains an approximation ratio of $(1-\delta)k$ for some $\delta > 0$.

We first observe that if only bidder i places bids, then $p_i(M) = 0$. Now let I_1 be an input instance containing items $(i, M, 1+\delta)$ and $(i, S, 1)$ for all $S \neq M$, for each $1 \leq i \leq N$. That is, each bidder is $(m+1)$-minded, with a value of 1 for each singleton and $1+\delta$ for the set of all objects. Then A must consider some input item first given input I_1; suppose the first item has corresponding bidder j. Now consider cases based on the nature of the first item.

Case 1: $(j, M, 1+\delta)$**.** Consider the decision made by A for this item. If A allocates M to j, then for input instance I_1 A obtains a social welfare of $1+\delta$, whereas the optimal welfare is k. Thus A has an approximation ratio no better than $(1+\delta)^{-1}k > (1-\delta)k$, a contradiction. Next suppose A does not allocate M to j. Consider input instance $I_2 \subset I_1$ that contains only item $(j, M, 1+\delta)$. Then A cannot distinguish between I_1 and I_2 when considering item $(j, M, 1+\delta)$. Thus A will not allocate M to bidder j on input I_2, which contradicts Theorem 2.1.

Case 2: $(j, S, 1)$**,** $S \neq M$**.** Consider the decision made by A for this item. Suppose A does not allocate S to bidder j. Let $I_3 \subseteq I_1$ be the input instance consisting only of items $(j, T, 1)$ for all $T \supseteq S$; that is, player j has a single-minded valuation for set S. Since A cannot distinguish between I_1 and I_3 when considering item $(j, S, 1)$, it must be that A does not allocate S to bidder j on input I_3. Since A does not allocate any set T to player j other than set S (by assumption), it must not allocate anything to player j. Thus A obtains a social welfare of 0 when 1 was possible, contradicting the supposed approximation ratio of A.

Thus A must allocate S to bidder j on input I_3. Let $I_4 \subseteq I_1$ be the input instance consisting of items $(j, S, 1)$ and $(j, M, 1+\epsilon)$. Then A will allocate S to bidder j in instance I_4, but this contradicts Theorem 2.1 (which requires that A allocate M to bidder j). $\square$

4 Truthful Submodular Priority Auctions

In [27] a class of greedy algorithms was proposed that is well-suited to submodular bidders. In such an algorithm, objects are considered one by one, and each object is assigned to the bidder with the highest marginal utility for it, with respect to the partial allocations already assigned. It was shown in that paper that any ordering of the items leads to a 2-approximation algorithm, but not every ordering of items leads to an algorithm that can be made truthful. However, this did not preclude the possibility of some particularly clever and adaptive method for ordering the objects, which would lead to truthfulness.

We introduce a model of priority algorithms that is motivated by the above greedy algorithm, which we call the *objects as items* model. In this model, items are associated with the m objects for the auction. An item will contain an object index j, plus the value $v_i(j|S)$ for all $i \in N$ and $S \subseteq M$ (where $v_i(x|S) := v_i(S \cup \{x\}) - v_i(S)$ is the marginal utility of bidder i for item x, given set S). We note that the online greedy algorithm described above falls into this model. We show that no greedy priority algorithm in this model can be truthful. By contrast, the same online greedy algorithm [27] that allocates each object to the bidder with highest marginal value is an optimal allocation (and hence incentive compatible by VCG pricing) for additive bidders.

Theorem 4.1. *Any greedy priority algorithm for the combinatorial auction problem that uses objects as items is not incentive compatible. This holds even if the bidders are assumed to be submodular.* [4]

5 Future Work

As indicated, the fundamental issue of algorithmic mechanism design attempts to bridge the competing demands of selfish agents with that of algorithmic constraints. We have considered this issue in the context of truthful greedy algorithms for CAs. Because of the importance of greedy algorithms in many current mechanisms, our results concerning priority algorithms (as a general model for greedy mechanisms) is a natural beginning to a more general study of the power and limitations of conceptually simple mechanisms. While the priority framework represents a restricted algorithmic approach, it models many existing algorithms and there are still many unresolved questions for the most basic mechanism design questions. In particular, we believe that the results of Section 3.2 can be unified removing the greediness and no incremental updates assumptions. Furthermore, we conjecture that truthful priority mechanisms cannot achieve an approximation ratio that (substantially) improves upon the $\min\{m, n\}$ approximation achieved by the "trivial greedy algorithm" (that only allocates objects to the bidder with the highest bid) for the s-CA problem where each bidder only desires sets of at most s objects. This is in contrast to the non truthful $(s + 1)$-approximation achieved by the "standard greedy algorithm" that considers and greedily accepts items (bidders or sets) as prioritized by non-increasing bid values. We also ask whether or not the greediness assumption can be eliminated in Theorem 4.1.

The given definition of priority algorithms can be extended in many ways. The simplest extension would allow the priority mechanism algorithm to have (simply computed) prior global knowledge (for example, the average over all bidders of the minimum and maximum values that bidders claim for sets). A more substantial extension would be to consider randomized priority algorithms. Going beyond priority algorithms, there are other conceptually simple algorithmic models, some of which can also be considered "greedy-like". For packing problems (such as the underlying allocation problem in combinatorial auctions), priority algorithms can be generalized to allow "revocable acceptances" (see [19]). That is, an algorithm may "de-allocate" sets or objects that have been previously allocated. That is, in terms of the priority model, the "acceptance" of an input item (e.g. granting a set to a bidder) can later be rejected to make a subsequent allocation feasible but all rejections of input items are irrevocable and feasibility must always be maintained. A somewhat related model is motivated by the local-ratio/ primal-dual framework (see [2]). More specifically, we can consider priority stack algorithms ([5]) where items (e.g. bidders or bids) initially accepted are placed in a stack and then the stack is popped to ensure feasibility. We note that both of these extensions permit non-monotone allocations even when the priority rule is monotone (as defined in [31]). We remark that our priority inapproximation results do not assume that the priority ordering is monotone and we leave open the question as to whether or not such algorithms (and extensions) can be made monotone when say the algorithms reject high priority bids. One extension that reflects some current algorithms is to consider algorithms that chose an allocation amongst a small set of (say) priority algorithms. In particular, we believe that our results can be extended to allow the mechanism to optimize between a given priority allocation and giving all objects to one bidder.

The results in this paper have been restricted to greedy CAs but the basic question applies to all mechanism design problems. Namely, when can a conceptually simple approximation to the underlying optimization problem be converted into a truthful mechanism that achieves (nearly) the same approximation? For example, what is the best truthful priority approximation for the celebrated machine scheduling problem introduced in Nisan and Ronen [34].

---

[4] By contrast, the greedy algorithm in [27] will find the optimal solution for *additive* bidders and can therefore be made truthful with VCG payments. On the other hand, it is not hard to show that truthful priority algorithms that use bidders as items cannot obtain even a sublinear approximation for additive bidders.

Acknowledgements

We are thankful to Shahar Dobzinski, Rica Gonen and Silvio Micali for helpful discussions.

Appendix: Missing Proofs

Claim 2.2: Suppose that a mechanism satisfies the conditions of Theorem 2.1. Then one can assume without loss of generality that for all $j \in N$, all $v_{-j} \in V_{-j}$, and all $S \subseteq T \subseteq M$, $p_j(S, v_{-j}) \leq p_j(T, v_{-j})$.

Proof. The if direction follows immediately from Theorem 2.1. For the other direction, suppose that $\mathcal{M} = (G, P)$ is an incentive compatible mechanism; then $\mathcal{M}$ satisfies the critical pricing property, say with prices $p_j(S)$. We will construct a new set of critical prices that satisfies the conditions of Theorem 2.1 while also satisfying the conditions of our claim.

For all $j \in N$, $v_{-j} \in V_{-j}$, and $S \subseteq M$, define $p'_j(S)$ by

$$p'_j(S) = \min\{p_j(T) | T \supseteq S\}.$$

Notice that $p'_j(S) \leq p'_j(T)$ whenever $S \subseteq T$. Furthermore, $p'_j(S) \leq p_j(S)$ for all $S \subseteq M$.

We claim that p'_j satisfies the conditions of Theorem 2.1 for the mechanism $\mathcal{M}$. Choose any $j \in N$ and $(v_j, v_{-j}) \in V$ and suppose that $G_j(v_j, v_{-j}) = S$. Then, by the critical pricing property (for prices p_j), $v_j(S) - p_j(S) \geq v_j(T) - p_j(T)$ for all $T \subseteq M$, and furthermore $P_j(v_j, v_{-j}) = p_j(S)$.

Suppose that $p_j(S) > p_j(T)$ for some $S \subseteq T$. This would imply $v_j(S) - p_j(S) < v_j(T) - p_j(T)$, which is a contradiction since we assumed that the prices p_j were critical prices. We can therefore assume that $p_j(S) \leq p_j(T)$ for all $S \subseteq T$. Then, from the definition of p'_j, $p'_j(S) = p_j(S)$. This implies that for any $T \subseteq M$, $v_j(S) - p'_j(S) = v_j(S) - p_j(S) \geq v_j(T) - p_j(T) \geq v_j(T) - p'_j(T)$. We therefore have that the prices p'_j are critical prices for the mechanism, as required. ☐

Claim 3.2: Under the above definitions, $p_i(M, v_{-i}) \leq kn$.

Proof. Suppose otherwise that $p_i(M, v_{-i}) > kn$. Suppose further that bidder i is single-minded with desired set M, and with $v_i(M) = kn$. Then $v_i(M) - p_i(M, v_{-i}) < 0 = v_i(\emptyset) - p_i(\emptyset, v_{-i})$. Therefore, by the critical pricing property, A cannot allocate M to bidder i, and hence bidder i obtains a value of 0. Now consider the social welfare obtained by A: it can be at most $n - 1$, since bidder i obtains a welfare of 0 and each other bidder has value at most 1 for any set. The optimal social welfare is kn, obtained by allocating M to bidder i. Hence A obtains an approximation ratio of $\frac{kn}{n-1} > k$ for this input instance, which is a contradiction. ☐

Claim 3.3: Suppose $I = \{(1, v_1), \ldots, (n, v_n)\}$ is a valid input instance, in which there exists $i \in N$ such that $v_i \in \{f_1, \ldots, f_k\}$, and for all $j \neq i$, $v_j \in \{g_1, \ldots, g_k\}$. Then on input I, A must allocate M to bidder i and $\emptyset$ to all other bidders.

Proof. For this input instance we have that $v_{-i} \in V'_{-i}$. Then $x > p_i(M, v_{-i})$ from the definition of x. But now, from the definition of f_i, $v_i(M) - p_i(M, v_{-i}) > (y + x) - x = y \geq v_i(S) \geq v_i(S) - p_i(S, v_{-i})$ for all $S \neq M$. Therefore, by the critical pricing property (Theorem 2.1), A must allocate M to bidder i as required. ☐

Claim 3.4: There exists a labelling of bidders and objects such that the following is true for all $0 \leq i < k/2$. Define $I_i := \{(j, g_j) | j \leq i\}$. Then for any valid input instance I such that $I_i \subseteq I \subseteq \mathcal{I}'$, A will consider all the items in I_i before all other items in I, and will choose to assign $\emptyset$ for each of the items in I_i.

Proof. We proceed by induction on i. The base case holds by taking $I_0 = \emptyset$. For general $i \geq 1$, suppose the claim is true for $i - 1$. Then $I_{i-1} = \{(1, g_1), \ldots, (i - 1, g_{i-1})\}$. Define $\mathcal{I}_i \subseteq \mathcal{I}'$ as follows:

$$\mathcal{I}_i := \{(j, v_j) : (j, v_j) \in \mathcal{I}', j \geq i\}.$$

That is, $\mathcal{I}_i$ contains items of $\mathcal{I}'$ corresponding to bidders that are not present in I_{i-1}. Then note that if I is a valid input instance such that $I_{i-1} \subseteq I$, then $I \subseteq I_{i-1} \cup \mathcal{I}_i$.

Consider the execution of A on any valid input instance $I \subseteq I_{i-1} \cup \mathcal{I}_{i-1}$. The algorithm will first consider the items of I_{i-1} and allocate $\emptyset$ to each bidder $1, \ldots, i - 1$ (by assumption). Once this is done, the algorithm will choose an ordering $\mathcal{T}$ over $\mathcal{I}_i$ and examine the next item in I according to $\mathcal{T}$.

Some item $(j, v_j) \in \mathcal{I}_i$ must come first under this ordering $\mathcal{T}$. Without loss of generality (by relabeling indices) this item is (i, f_i) or (i, g_i). We consider these two cases separately.

Case 1: The first item is (i, f_i). In this case we will choose I so that $(i, f_i) \in I$. Then A must consider this item next when processing input instance I, and A must assign some set S to bidder i. If $S = M$, then we will choose I to contain (j, f_j) for all $j > i$; note that $I \subseteq I_{i-1} \cup \mathcal{I}_{i-1}$ as required. Since A allocated M to bidder i, it obtains a social welfare of $x + y$ on input I. However, the optimum welfare is at least $(k - i + 1)y$, since this can be attained by allocating $\{a_j\}$ to bidder j for all $i \leq j \leq k$. Thus the approximation ratio obtained by A is at least

$$\frac{(k-i+1)y}{x+y} > \frac{(k/2)y}{y(1+\delta)} > \frac{(1-\delta)k}{2},$$

a contradiction.

If, on the other hand, $S \neq M$, we choose I to contain (j, g_j) for all $j > i$. Then I satisfies the requirements of Claim 3.3, so A must allocate M to bidder i. This is a contradiction. We conclude that this first case cannot occur.

Case 2: The item is (i, g_i). In this case we will choose I so that $(i, g_i) \in I$. As in the previous case, A must consider this item next in I, and assign some set S to bidder i. Suppose $S \neq \emptyset$. Then we will choose I to contain $(i + 1, f_{i+1})$, and also (j, g_j) for all $j > i + 1$. Note that then $I \in I_{i-1} \cup \mathcal{I}_{i-1}$ as required. Also, in this instance of a combinatorial auction, $v_{-(i+1)}$ contains only single-minded valuations for singletons with value 1. Thus, by the same argument used in Case 1, it must be that bidder $i + 1$ is allocated M. However, this is not possible, since bidder i is assigned $S \neq \emptyset$. This is a contradiction. We conclude that in this case, bidder i must be assigned $\emptyset$.

This ends our case analysis. We conclude that item (i, g_i) must occur first in $\mathcal{I}_{i-1}$ in ordering $\mathcal{T}$, and furthermore if $(i, g_i) \in I$ then A will consider (i, g_i) next after processing the items in I_{i-1} and will assign $\emptyset$ to bidder i. We can therefore set $I_i = I_{i-1} \cup \{(i, g_i)\}$ to satisfy the requirements of the claim. □

Claim 3.6: For all $i \in N$, and for all $a, b \in M$, either $p_i(\{a\}, v_{-i}) \geq p_i(\{b\}, v_{-i})$ for all $v_{-i} \in V_{-i}^{+}$, or $p_i(\{a\}, v_{-i}) \leq p_i(\{b\}, v_{-i})$ for all $v_{-i} \in V_{-i}^{+}$.

Proof. Choose $i \in N$, $a, b \in M$, and $v_i, v_i' \in V_{-i}^{+}$. Suppose for contradiction that $p_i(\{a\}, v_{-i}) > p_i(\{b\}, v_{-i})$ but $p_i(\{b\}, v'_{-i}) > p_i(\{a\}, v'_{-i})$. We will consider a number of possible valuations to be declared by our bidders.

Let v^* be the maximum value assigned to any set by any player $j \neq i$ with valuations in v_{-i} or v'_{-i}. Then note that the maximum social welfare that can be obtained is $(k-1)v^*$ if bidder i does not participate and other bidders declare values v_{-i} or v'_{-i}. Let $x = k^2 v^*$. We will define three different possible valuation functions for bidder i: f, g, and h. $f(S) = \begin{cases} x & \text{if } a \in S \\ x & \text{if } b \in S \\ 0 & \text{otherwise.} \end{cases}$

$$g(S) = \begin{cases} \epsilon & \text{if } a \in S \\ \epsilon & \text{if } b \in S \\ x & \text{if } S = M \\ 0 & \text{otherwise.} \end{cases}$$

$$h(S) = \begin{cases} \epsilon & \text{if } a \in S \\ \epsilon & \text{if } b \in S \\ 0 & \text{otherwise.} \end{cases}$$

We are now ready to discuss the behaviour of algorithm A. Consider the subset $\mathcal{I}_1 \subset \mathcal{I}$ that contains the following input items: $(i, S, f(S))$, $(i, S, g(S)$, and $(i, S, h(S)$ for every $S \subseteq M$; and $(j, S, v_j(S))$, $(j, S, v_j'(S))$, (j, S, ϵ), and (j, S, v^*) for all $j \neq i$ and $S \subseteq M$. In other words, $\mathcal{I}_1$ contains all of the input items consistent with the valuation functions we defined above, plus values of ϵ and v^* for each set and each bidder $j \neq i$.

We know that A must have some initial ordering over $\mathcal{I}$, and hence over $\mathcal{I}_1$. Consider the first item in $\mathcal{I}_1$ under this ordering. We consider different cases for the nature of this item.

Case 1: (j, S, t), $j \neq i$. Then $t \in \{v_j(S), v_j'(S), \epsilon, v^*\}$. Suppose first that $t = v_j(S)$, and consider the input instance I_1 corresponding to a declaration of valuation g by bidder i and valuation v_j for each bidder $j \neq i$. Then $I_1 \subseteq \mathcal{I}_1$ and $(j, S, v_j(S)) \in I_1$, so item $(j, S, v_j(S))$ will be considered first by algorithm A on input I_1.

Since $v_{-i} \in V_{-i}^{+}$, it must be that $v_j(S) > 0$ and hence, since A is greedy, A will allocate set S to bidder j. Then it must be that, in the final allocation, bidder i is not allocated M. Thus bidder i obtains a value of at most ϵ, and all other bidders can obtain a total welfare of at most $(k-1)v^*$, for a total social welfare of at most $(k-1)v^* + \epsilon$. On the other hand, a total of $x = k^2 v^*$ is possible by allocating M to bidder i. Then as long as $\epsilon < v^*$ the approximation ratio obtained by A is at least k, a contradiction.

The other cases for t are handled similarly.

Case 2: $(i, S, f(S))$**,** $a \in S$ **or** $b \in S$. By symmetry we can assume $a \in S$. Consider the input instance I_2 in which bidder i declares valuation f, and every other bidder $j \neq i$ declares valuation v_j. Then $(i, S, f(S)) \in I_2 \subseteq \mathcal{I}_1$, so A will consider item $(i, S, f(S))$ first on input I_2. Since $f(S) = x > 0$ and A is greedy, the algorithm will assign set S to bidder i.

Suppose that in the final allocation, bidder i is allocated some set $T \supseteq S$. Then since $a \in T$, we know that $p_i(T, v_{-i}) \geq p_i(\{a\}, v_{-i}) > p_i(\{b\}, v_{-i})$. But note $f(T) = f(\{a_k\}) = x$, so that $f(T) - p_i(T) < f(\{b\}) - p_i(\{b\})$. In other words, A does not maximize the utility of player i, so by Theorem 2.1 A is not incentive compatible, a contradiction.

Case 3: $(i, S, g(S))$**,** $a \in S$ **or** $b \in S$. This case leads to a contradiction in the same way as Case 2.

Case 4: $(i, S, h(S))$**,** $a \in S$ **or** $b \in S$. This case leads to a contradiction in the same way as Case 2.

Case 5: (i, S, t)**,** $a \notin S$ **and** $b \notin S$. Then from the definitions of f, g, and h, we must have $t = 0$. Thus when processing this item, A is free to allocate S to bidder i or not. If A does not allocate S to i, then we will consider the *next* item considered by the algorithm A, and repeat our case analysis. The case analysis proceeds in the same way, since no objects would have been allocated. This process must terminate, as algorithm A must eventually consider some set S that contains either a or b.

Suppose, on the other hand, that A does allocate S to i. Then consider the input instance I_3 in which bidder i declares valuation h and all other bidders declare the following valuation f_S:

$$f_S(T) = \begin{cases} v^* & \text{if } S \subseteq T \\ \epsilon & \text{otherwise.} \end{cases}$$

We note that valuation f_S defines the value of any set to be either ϵ or v^*, so in particular $I_3 \subseteq \mathcal{I}_1$. Since $(i, S, 0) \in I_3$, this item will be considered first by A on input I_3, and S will be allocated to player i. But then in the final allocation each bidder can obtain a welfare of at most ϵ, for a total welfare of at most $k\epsilon$. On the other hand, a welfare of v^* was possible by allocating S to any bidder other than bidder i. Thus, if we choose $\epsilon < v^*/k^2$ we conclude that A has an approximation ratio of at least k, a contradiction.

We have shown that every case leads to a contradiction, completing the proof of Claim 3.6. ☐

Claim 3.8: There exists $a^* \in M$ such that $|S(a^*)| \geq k/2$.

Proof. We note that

$$\sum_{i \in N} \sum_{\substack{a \in M \\ |T_i(a)| \geq k/2}} 1 = \sum_{i \in N} (m - k/2) = n(m - k/2).$$

Rearranging order of summation, we also have

$$\sum_{i \in N} \sum_{\substack{a \in M \\ |T_i(a)| \geq k/2}} 1 = \sum_{a \in M} \sum_{\substack{i \in N \\ |T_i(a)| \geq k/2}} 1 = \sum_{a \in M} |S(a)|.$$

We conclude that $\sum_{a \in M} |S(a)| = n(m - k/2)$, so there must exist some $a^* \in M$ such that $|S(a^*)| \geq \frac{n(m-k/2)}{m}$. We know that either $n \geq m = k$ or $m \geq n = k$; in either case we obtain $|S(a^*)| \geq \frac{n(m-k/2)}{m} \geq k/2$ as required. ☐

Theorem 4.1: Any greedy priority algorithm for the combinatorial auction problem that uses objects as items is not incentive compatible. This holds even if the bidders are assumed to be submodular.

Proof. Suppose for contradiction that A is an incentive compatible truthful greedy priority algorithm. Consider an instance of the combinatorial auction with $M = \{a_1, a_2, a_3\}$. Suppose that bidder 1 declares the following valuation function: $v_1(S) = 9 + |S|$ for all $S \neq \emptyset$. It is easy to verify that this is indeed submodular. Then by Theorem 2.1 this valuation defines a critical price $p_2(S)$ for each subset $S \subseteq M$. Consider the critical prices for all subsets of size 2 and suppose without loss of generality that $\{a_1, a_2\}$ has the smallest. That is, $p_2(\{a_1, a_2\}) \leq p_2(\{a_2, a_3\})$ and $p_2(\{a_1, a_2\}) \leq p_2(\{a_1, a_3\})$.

We now define a valuation function v_2 to be declared by bidder 2. The motivation for v_2 is that items a_1 and a_2 will have lower values than a_3 when considered individually, but will have a large value when taken together.

$$\begin{aligned} v_2(\{a_1\}) = v_2(\{a_2\}) &= 9 \\ v_2(\{a_3\}) &= 11 \\ v_2(\{a_1, a_2\}) &= 18 \\ v_2(\{a_1, a_3\}) = v_2(\{a_2, a_3\}) &= 17 \\ v_2(\{a_1, a_2, a_3\}) &= 18. \end{aligned}$$

The reader is encouraged to verify that this valuation is submodular.

Given as input the valuations v_1 and v_2, algorithm A must consider each object in turn, and assign that object to the player who obtains the greatest marginal utility from it. The algorithm is free to choose the order in which the items are considered. However, regardless of the order, the only possible outcomes are that bidder 2 is allocated $\{a_1, a_3\}$ or bidder 2 is allocated $\{a_2, a_3\}$. This can be seen by examining each of the 6 possible orderings of items, or by noticing that the first item considered will go to bidder 2 if and only if it is a_3, that bidder 1 will never be allocated a second object, and that bidder 2 will never be allocated a third object.

We will assume without loss of generality that bidder 2 is allocated $\{a_1, a_3\}$. Then, by Theorem 2.1,

$$v_2(\{a_1, a_3\}) - p_2(\{a_1, a_3\}) \geq v_2(\{a_1, a_2\}) - p_2(\{a_1, a_2\}).$$

Since $v_2(\{a_1, a_3\}) = 17$ and $v_2(\{a_1, a_2\}) = 18$, this implies that

$$p_2(\{a_1, a_2\}) > p_2(\{a_1, a_3\})$$

which contradicts the minimality of $p_2(\{a_1, a_2\})$.

We have now proved the result for the case of exactly two bidders and three objects. The result follows more generally by noticing that we may add additional players who value all sets at 0, and additional items for which no players have value, without affecting the above construction. □

References

[1] M. Babaioff, R. Lavi, E. Pavlov, *Single-Value Combinatorial Auctions and Implementation in Undominated Strategies*. In 17th ACM-SIAM Symposium on Discrete Algorithms (SODA 2006), 1054–1063, 2006.

[2] A. Bar-Noy, R. Bar-Yehuda, A. Freund, J. Naor, B. Schieber, *A unified approach to approximating resource allocation and scheduling*, JACM, volume 48(5), 1069-1090, 2001.

[3] Y. Bartal, R. Gonen, N. Nisan, *Incentive compatible multi unit combinatorial auctions*. In 9th Conf. Theor. Aspects of Rationality and Knowledge (TARK), 72–87, 2003.

[4] L. Blumrosen, N. Nisan, *On the Computational Power of Iterative Auctions*. In 6th ACM Conf. on Electronic Commerce (EC 2005). ACM Press, 29–43, 2005.

[5] A. Borodin, D. Cashman, A. Magen, *How well can Primal-Dual and Local-Ratio algorithms perform?*, Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP), 943-955, 2005, Journal version submitted to Transactions on Algorithms and accepted subject to revisions.

[6] A. Borodin, M. N. Nielsen, C. Rackoff, *(Incremental) priority algorithms*. In 13th ACM-SIAM Symposium on Discrete Algorithms (SODA 2002), 752–761, 2002.

[7] P. Briest, P. Krysta, B. Vöcking, *Approximation techniques for utilitarian mechanism design*. In 37th ACM Symposium on Theory of Computing (STOC 2005), 39–48, 2005.

[8] G. Christodoulou, A. Kovács, M. Schapira, *Bayesian Combinatorial Auctions*. In 35th International Colloquium on Automata, Languages and Programming (ICALP 2008), 820–832, 2008.

[9] V. Conitzer, T. Sandholm, *Computational Criticisms of the Revelation Principle*. In 5th ACM Conf. on Electronic Commerce (EC 2004), 262–263, 2004.

[10] S. Dobzinski, N.Nisan, M. Schapira, *Truthful randomized mechanisms for combinatorial auctions*, In Proceedings of the thirty-eighth annual ACM symposium on Theory of computing (STOC 2006), 644–652, 2006.

[11] S. Dobzinski, N. Nisan, M. Schapira, *Approxmation algorithms for combinatorial auctions with complement-free bidders*. In 37th ACM Symposium on Theory of Computing (STOC 2005), 610–618, 2005.

[12] S. Dobzinski, M. Schapira, *An Improved Approximation Algorithm for Combinatorial Auctions with Submodular Bidders*. In 17th ACM-SIAM Sympo-

sium on Discrete Algorithms (SODA 2006), 1064–1073, 2006.

[13] S. Dobzinski, M. Sundararajan, *On Characterizations of Truthful Mechanisms for Combinatorial Auctions and Scheduling*. In 9th ACM Conf. on Electronic Commerce (EC 2008), 38–47, 2008.

[14] B. Edelman, M. Ostrovsky, M. Schwarz, *Internet Advertising and the Generalized Second Price Auctions: Selling Billions of Dollars Worth of Keywords.* American Economic Review, 97(1): 242–259, 2007.

[15] J. Feigenbaum, S. Shenker, *Distributed Algorithmic Mechanism Design: Recent Results and Future Directions*. In 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (Dial-M 2002), 1–13, 2002.

[16] E. Friedman, D. Parkes, *Pricing WiFi at Starbucks - Issues in online mechanism design*. In 4th ACM Conf. on Electronic Commerce (EC 2003), 240–241, 2003.

[17] R. Gonen and D. Lehmann, *Optimal Solutions for Multi-Unit Combintorial Auctions: Branch and Bound Heuristics*. In 2nd ACM Conference on Electronic Commerce (EC 2000), 13–20, 2000.

[18] J. Hastad, *Clique is hard to approximate within* $n^{1-\epsilon}$, ECCC Report TR97-038, 1997.

[19] S. Horn, *One-Pass algorithms with revocable acceptances for job interval selection*, University of Toronto MSC thesis, 2004.

[20] S. Khot, R. Lipton, E. Markakis, A. Mehta, *Inapproximability Results for Combinatorial Auctions with Submodular Utility Functions*. 1st Workshop on Internet and Network Economics (WINE), 92–101, 2005.

[21] A. Kothari, D. Parkes, S. Suri, *Approximately-Strategyproof and Tractable Multi-Unit Auctions*. 4th ACM conference on Electronic commerce (EC 2003), 166–175, 2003.

[22] P. Krysta, *Greedy Approximation via Duality for Packing, Combinatorial Auctions and Routing*. In 30th International Symposium on Mathematical Foundations of Computer Science, 615–627, 2005.

[23] R. Lavi, A. Mu'alem, N. Nisan, *Towards a characterization of truthful combinatorial auctions*. In 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2003), 574, 2003.

[24] R. Lavi, N. Nisan, *Competitive analysis of incentive compatible on-line auctions*. In 2nd ACM Conference on Electronic Commerce (EC 2000), 233–241, 2000.

[25] R. Lavi, N. Nisan, *Online Ascending Auctions for Gradually Expiring Items*. In 16th ACM-SIAM Symposium on Discrete Algorithms (SODA 2005), 1146–1155, 2005.

[26] R. Lavi, C. Swamy, *Truthful and Near-Optimal Mechanism Design via Linear Programming*. In 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005), 595–604, 2005.

[27] B. Lehmann, D. Lehmann, and N. Nisan, *Combinatorial Auctions with Decreasing Marginal Utilities*. In 3rd ACM Conference on Electronic Commerce (EC 2001), 18–28, 2001.

[28] D. Lehmann, L. I. O'Callaghan, Y. Shoham, *Truth revelation in approximately efficient combinatorial auctions*. Journal of the ACM 49(5), 577–602, 2002.

[29] B. Lucier *Beyond Equilibra: Mechanisms for Repeated Combinatorial Auctions,* submitted to ICS.

[30] B. Lucier A.Borodin, *Price of Anarchy for Greedy Auctions*. To appear in SODA 2010.

[31] A. Mu'alem, N. Nisan, *Truthful Approximation Mechanisms for Restricted Combinatorial Auctions.* Eighteenth national conference on Artificial intelligence (AAAI 2002), 379–384, 2002.

[32] M. Mahdian, A. Saberi, *Multi-unit Auctions with Unknown Supply.* In 7th ACM Conf. on Electronic Commerc (EC 2006), 243–249, 2006.

[33] N. Nisan, *Bidding and allocation in combinatorial auctions.* 2nd ACM conference of Electronic Commerce (EC 2000), 1–12, 2000.

[34] N. Nisan and A. Ronen, *Algorithmic Mechanism Design.* Games and Economic Behavior (35): 166-196, 2001.

[35] C. Papadimitriou, M. Schapira and Y. Singer, *On the Hardness of Being Truthful.* In 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2008), 2008.

[36] H. Varian, *Position Auctions.* International Journal of Industrial Organization (25): 1163–1178, 2006.