On the Limitations of Greedy Mechanism Design for Truthful Combinatorial Auctions

Allan Borodin[1] and Brendan Lucier[1]

[1] Dept of Computer Science,University of Toronto, bor@cs.toronto.edu.
[2] Dept of Computer Science,University of Toronto, blucier@cs.toronto.edu.

Abstract. We study the combinatorial auction (CA) problem, in which m objects are sold to rational agents and the goal is to maximize social welfare. Of particular interest is the special case in which agents are interested in sets of size at most s (s-CAs), where a simple greedy algorithm obtains an $s+1$ approximation but no truthful algorithm is known to perform better than $O(m/\sqrt{\log m})$. As partial work towards resolving this gap, we ask: what is the power of truthful greedy algorithms for CA problems? The notion of greediness is associated with a broad class of algorithms, known as priority algorithms, which encapsulates many natural auction methods. We show that no truthful greedy priority algorithm can obtain an approximation to the CA problem that is sublinear in m, even for s-CAs with $s \geq 2$.

1 Introduction

The field of algorithmic mechanism design attempts to bridge the competing demands of agent selfishness and computational constraints. 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, the well-known VCG mechanism does precisely that: each agent maximizes his utility by reporting truthfully. However, the VCG mechanism requires that the underlying optimization problem be solved exactly, and is therefore ill-suited for computationally intractable problems. Determining the power of truthful *approximation* mechanisms is a fundamental problem in algorithmic mechanism design.

The combinatorial auction (CA) problem holds a position at the center of this conflict between truthfulness and approximability. Without strategic considerations, one can obtain an $O(\min\{n, \sqrt{m}\})$ approximation for CAs with n bidders and m objects with a conceptually simple (albeit not obvious) greedy algorithm [25], and this is the best possible under standard complexity assumptions [16, 31]. However, no deterministic truthful mechanism for multi-minded auctions is known to obtain an approximation ratio better than $O(\frac{m}{\sqrt{\log m}})$ [17]. This is true even for the special case where each bidder is interested only in sets of size at most some constant s (the s-CA problem), where the obvious greedy algorithm obtains an $s+1$ approximation. Whether these gaps are essential for the CA problem, or whether there is some universal process by which approximation algorithms can be made truthful without heavy loss in performance, is a central open question that has received significant attention over the past decade [14, 21, 25, 28, 30].

A lower bound for the related combinatorial public project problem [30] shows that there is a large asymptotic gap separating approximation by deterministic algorithms and by deterministic truthful mechanisms in general allocation problems. Currently, the only such lower bounds known for the CA problem are limited to max-in-range (MIR) algorithms [9]. While many known truthful CA algorithms are MIR, the possibility yet remains that non-MIR algorithms could be used to bridge the gap between truthful and non-truthful CA design. We consider lower bounds for truthful CAs by focusing on an alternative class of algorithms. We ask: can any truthful *greedy* algorithm obtain an approximation ratio better than $O(\frac{m}{\sqrt{\log(m)}})$? Our interest in greedy algorithms is motivated threefold. First, most known examples of truthful, non-MIR algorithms for combinatorial auction problems apply greedy methods [1, 4, 8, 10, 20, 24, 25, 28]; indeed, greedy algorithms embody the conceptual monotonicity properties generally associated with truthfulness, and are thus natural candidates for truthful mechanism construction. Second, simple greedy auctions are often used in practice, despite the fact that they are not incentive compatible; this leads us to suspect that they are good candidates for auctions due to other considerations, such as ease of public understanding. Finally, greedy algorithms are known to obtain asymptotically tight approximation bounds for many CA problems despite their simplicity.

We use the term "greedy algorithm" to refer to any of a large class of algorithms known as *priority algorithms* [7]. The class of priority algorithms captures a general notion of greedy algorithm behaviour. Priority algorithms include, for example, many well-known primal-dual algorithms, as well as other greedy algorithms with adaptive and non-trivial selection rules. Moreover, this class is *independent of computational constraints* and also 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. Roughly speaking, a priority algorithm has some notion of what consistutes the "best" bid in any given auction instance; the auction finds this bid, satisfies it, then iteratively resolves the reduced auction problem with fewer objects (possibly with an adaptive notion of the "best" bid). For example, the truthful algorithm for multi-unit auctions due to Bartal et al. [4] that updates a price vector while iteratively satisfying agent demands falls into this framework. Our main result demonstrates that if a truthful auction for an s-CA proceeds in this way, then it cannot perform much better than the trivial algorithm that allocates all objects to a single bidder.

Theorem: No deterministic truthful priority algorithm (defined formally in the text) for the CA problem obtains an $o(\min\{m, n\})$ approximation to the optimal social welfare (even for s-CAs with $s \geq 2$).

The gap described in our result is extreme: for $s = 2$, the standard (but non-truthful) greedy algorithm is a 3-approximation for the s-CA problem, but no truthful greedy algorithm can obtain a sublinear approximation bound.

We also consider the combinatorial auction problem for submodular bidders (SMCA), which has been the focus of much study [13, 12, 19, 24]. 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 [24] that any such algorithm[3] attains a 2-approximation to the SMCA problem, but that not all are incentive compatible. We show that, in fact, no such algorithm can be incentive compatible.

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

Many truthful approximation mechanisms are known for CAs with single-minded bidders. Following the Lehmann et al. [25] truthful greedy mechanism for single-minded CAs, Mu'alem and Nisan [28] showed that any *monotone* greedy algorithm for single-minded bidders is truthful, and outlined various techniques for combining approximation algorithms while retaining truthfulness. This led to the development of many other truthful algorithms in single-minded settings [2, 8] and additional construction techniques, such as the iterative greedy packing of loser-independent algorithms due to Chekuri and Gamzu [10].

Less is known in the setting of general bidder valuations. Bartal et al. [4] give a greedy algorithm for multi-unit CAs that obtains an $O(Bm^{\frac{1}{B-2}})$ approximation when there are B copies of each object. Lavi and Swamy [23] give a general method for constructing randomized mechanisms that are truthful in expectation, meaning that agents maximize their expected utility by declaring truthfully. Their construction generates a k-approximate mechanism from an LP for which there is an algorithm that verifies a k-integrality gap. In the applications they discuss, these verifiers take the form of greedy algorithms, which play a prominant role in the final mechanisms.

A significant line of research aims to give lower bounds on the approximating power of deterministic truthful algorithms for CAs. Lehmann, Mu'alem, and Nisan [21] show 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. It has also been shown that, roughly speaking, no truthful polytime subadditive combinatorial auction mechanism that is *stable*[4] can obtain an approximation

[3] The degree of freedom in this class of algorithms is the order in which the objects are considered.

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

ratio better than 2 [14]. Also, no max-in-range algorithm can obtain an approximation ratio better than $\Omega(\sqrt{m})$ when agents have budget-constrained additive valuations [9]. These lower bounds are incomparable to our own, as priority algorithms need not be MIR, stable, unanimity-respecting, or satisfy IIA[5].

Another line of work gives lower bounds for greedy algorithms without truthfulness restrictions. Gonen and Lehmann [15] showed that no algorithm that greedily accepts bids for sets can guarantee an approximation better than $\sqrt{m}$. Similarly, Krysta [20] showed that no oblivious greedy algorithm (in our terminology: fixed order greedy priority algorithm) obtains 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, we consider the more general class of priority algorithms but restrict them to be incentive-compatible.

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 [22, 27]. The critical difference between these results and our lower bounds is that a priority algorithm has control over the order in which input items are considered, whereas in an online setting this order is chosen adversarily.

In contrast to the negative results of this paper, greedy algorithms can provide good approximations when rational agents are assumed to bid at Bayes-Nash equilibria. In particular, there is a greedy combinatorial auction for submodular agents that obtains a 2-approximation at equilibrium [11], and the greedy GSP auction for internet advertising can be shown to obtain a 1.6-approximation at equilibrium [26]. Recently, we have shown [6] that, in a wide variety of contexts, c-approximate monotone greedy allocations can be made into mechanisms whose Bayes-Nash equilibria yield $c(1 + o(1))$ approximations.

2 Defintions and Preliminary Results

Combinatorial Auctions. A *combinatorial auction* consists of n bidders and a set M of m objects. Each bidder i has a value for each subset of objects $S \subseteq M$, described by a valuation function $v_i : 2^M \to \mathbb{R}$ which we call the *type* of agent i. We assume each v_i is monotone and normalized so that $v_i(\emptyset) = 0$. We denote by V_i the space of all possible valuation functions for agent i, and $V = V_1 \times V_2 \times \cdots \times V_n$. We write $\mathbf{v}$ for a profile of n valuation functions, one per agent, and $\mathbf{v}_{-i} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$, so that $\mathbf{v} = (v_i, v_{-i})$.

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 $T \subseteq M$, $v(T) = x$ if $S \subseteq T$ and 0 otherwise. 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 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

[5] The notion of IIA has been associated with priority algorithms, but in a different context than in [21]. 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.

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 $S, T \subseteq M$.

A *direct revelation mechanism* (or just *mechanism*) $\mathcal{M} = (G, P)$ consists of an *allocation algorithm* G and a *payment algorithm* P. Given valuation profile $\mathbf{d}$, $G(\mathbf{d})$ returns an allocation of objects to bidders, and $P(\mathbf{d})$ returns the payment extracted from each agent. For each agent i we write $G_i(\mathbf{d})$ and $P_i(\mathbf{d})$ for the set given to and payment extracted from i. We think of $\mathbf{d}$ as a profile of declared valuations made by the agents to the mechanism. The *social welfare* obtained by G on declaration $\mathbf{d}$ is $SW(\mathbf{d}) = \sum_{i \in N} d_i(G_i(\mathbf{d}))$. The *optimal social welfare*, SW_{opt}, is the maximum of $\sum_{i \in N} t_i(S_i)$ over all valid allocations $(S_1, \ldots, S_n)$. Algorithm G is a c-approximation if $SW(\mathbf{t}) \geq \frac{1}{c} SW_{opt}$ for all type profiles $\mathbf{t}$.

Fixing mechanism $\mathcal{M}$ and type profile $\mathbf{t}$, the *utility* of bidder i given declaration $\mathbf{d}$ is $u_i(\mathbf{d}) = t_i(G_i(\mathbf{d})) - P_i(\mathbf{d})$. Mechanism $\mathcal{M}$ is *truthful* (or *incentive compatible*) if for every type profile $\mathbf{t}$, agent i, and declaration profile $\mathbf{d}$, $u_i(t_i, \mathbf{d}_{-i}) \geq u_i(\mathbf{d})$. That is, agent i maximizes his utility by declaring his type, regardless of the declarations of the other agents. We say that G is truthful if there exists a payment function P such that the mechanism (G, P) is truthful.

Critical Prices. From Bartal, Gonen and Nisan [4], we have the following characterization of truthful CA mechanisms.

Theorem 1. *A mechanism is truthful if and only if, for every i, S, and $\mathbf{d}_{-i}$, there is a price $p_i(S, \mathbf{d}_{-i})$ such that whenever bidder i is allocated S his payment is $p_i(S, \mathbf{d}_{-i})$, and agent i is allocated a set S_i that maximizes $d_i(S_i) - p_i(S_i, \mathbf{d}_{-i})$.*

We refer to $p_i(S, \mathbf{d}_{-i})$ as the *critical price* of S for agent i. Note that $p_i(S, \mathbf{d}_{-i})$ need not be finite: if $p_i(S, \mathbf{d}_{-i}) = \infty$ then the mechanism will not allocate S to bidder i for any reported valuation d_i. In addition, one can assume without loss of generality that critical prices are monotone.

Priority Algorithms. We view an input instance to an algorithm as a selection of *input items* from a known input space $\mathcal{I}$. Note that $\mathcal{I}$ depends on the problem being considered, and is the set of *all possible* input items: an input instance is a finite subset of $\mathcal{I}$. The problem definition may place restrictions on the input: an input instance $I \subseteq \mathcal{I}$ is *valid* if it satisfies all such restrictions. The output of the algorithm is a decision made for each input item. For example, these decisions may be of the form "accept/reject", allocate set S to agent i, etc. The problem 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. A *priority algorithm* is then any algorithm of the following form:

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 $\leftarrow$ first item in I according to ordering $\mathcal{T}$
  Decision: make an irrevocable 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* ordering over the space of possible input items, and can change this ordering adaptively after each input item is considered. Once an item is processed, the algorithm is not permitted to modify its decision. 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.

The term "greedy" implies a more opportunistic aspect than is apparent in the definition of priority algorithms and indeed we view priority algorithms 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.

3 Truthful Priority Algorithms

We wish to show that no truthful priority algorithm can provide a non-trivial approximation to social welfare. In order to apply the concept of priority algorithms we must define the set $\mathcal{I}$ of possible input items and the nature of decisions to be made. We consider two natural input formulations: sets as items, and bidders as items. We 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 Sets as Items

In our primary model, we view an input instance to the combinatorial auction problem as a list of set-value pairs for each bidder. An item is a tuple (i, S, t), $i \in N$, $S \subseteq M$, 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 since a valid input instance may contain an exponential number of items, this model applies most directly to algorithms that use oracles to query input valuations, such as demand oracles[6], but it can also apply to succinctly represented valuation functions.[7]

The decision to be made for item (i, S, t) is whether or not the objects in S should be added to any objects already allocated to bidder i. For example,

[6] 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.

[7] That is, by assigning priority only to those tuples appearing in a given representation.

an algorithm may consider 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 change bidder i's allocation to $S_1 \cup S_2$.

A *greedy algorithm* in the sets as items model must accept any feasible, profitable item (i, S, t) it considers[8]. Our main result is a lower bound on the approximation ratio achievable by a truthful greedy algorithm in the sets as items model.

Theorem 2. *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 result also applies to the special case of (3-minded bidders for) the 2-CA problem, in which each desired set has size at most 2.*

Theorem 2 implies a severe separation between the power of greedy algorithms and the power of truthful greedy algorithms. A simple greedy algorithm obtains a 3-approximation for the 2-CA problem, yet no truthful greedy priority algorithm (indeed, any algorithm that irrevocably satisfies bids based on a notion of priority) can obtain even a sublinear approximation.

Proof. Choose $\delta > 0$ and suppose A obtains a bounded approximation ratio. For each $i \in N$, let V_{-i}^{+} be the set of valuations with the property that $v_\ell(S) > 0$ for all $\ell \neq i$ 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 $p_i(S, \mathbf{d}_{-i})$ is the critical price for set S for bidder i, given $\mathbf{d}_{-i}$.

Lemma 1. *For all $i \in N$, and for all $a, b \in M$, either $p_i(\{a\}, \mathbf{d}_{-i}) \geq p_i(\{b\}, \mathbf{d}_{-i})$ for all $\mathbf{d}_{-i} \in V_{-i}^{+}$, or $p_i(\{a\}, \mathbf{d}_{-i}) \leq p_i(\{b\}, \mathbf{d}_{-i})$ for all $\mathbf{d}_{-i} \in V_{-i}^{+}$. This is true even when agents desire sets of size at most 2.*

We can think of Lemma 1 as defining, for each $i \in N$, an ordering over the elements of M. For each $i \in N$ and $a, b \in M$, write $a \preceq_i b$ to mean $p_i(a, \mathbf{d}_{-i}) \leq p_i(b, \mathbf{d}_{-i})$ for all $\mathbf{d}_{-i} \in V_{-1}^{+}$. 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 have higher price than a for agent i. Our next claim shows a strong relationship between whether a is allocated to bidder i and whether any object in $T_i(a)$ is allocated to bidder i.

Lemma 2. *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 $\mathbf{v}_{-i} \in V_{-i}^{+}$, bidder i cannot be allocated set S by algorithm A given input $\mathbf{v}$.*

Lemma 2 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 $R(a) = \{i \in N : |T_i(a)| \geq k/2\}$, so $R(a)$ is the set of players for which there are at least $k/2$ objects greater than a. The next claim follows by a straightforward counting argument.

[8] That is, any item (i, S, t) such that no objects in S have already been allocated to another bidder and $t > 0$.

Lemma 3. *There exists $a^* \in M$ such that $|R(a^*)| \geq k/2$.*

We are now ready to proceed with the proof of Theorem 2. Let $a^* \in M$ be the object from Lemma 3. Let $\epsilon > 0$ be a sufficiently small value to be defined later. We now define a particular input instance to algorithm A. For each $i \in R(a^*)$, 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 $i \notin R(a^*)$ will declare a value of ϵ for every set.

For each $i \in R(a^*)$, $v_i(a_j) \geq 1-\delta/2$ for every $a_j \in T_i(a^*)$. Since $|R(a^*)| \geq k/2$ and $|T_i(a^*)| \geq k/2$, it is possible to obtain a social welfare of at least $\frac{(1-\delta/2)k}{2}$ by allocating singletons to bidders in $R(a^*)$.

Consider the social welfare obtained by algorithm A. The algorithm can allocate object a^* to at most one bidder, say bidder i, who will obtain a social welfare of at most 1. For any bidder $\ell \in R(a^*)$, $\ell \neq i$, $v_\ell(S) = 1 - \delta/2 < 1$ for any S containing elements of $T_\ell(a^*)$ but not a^*. Thus, by Lemma 2, no bidder in $R(a^*)$ can be allocated any set S that contains an element of $T_i(a^*)$ but not a^*. Therefore every bidder other than bidder i 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 no better than $\frac{k(1-\delta/2)}{2(1+k\epsilon)}$. Choosing $\epsilon < \frac{\delta}{2(1-\delta)k}$ yields an approximation ratio greater than $\frac{k(1-\delta)}{2}$, completing the proof of Theorem 2.

We believe that the greediness assumption of Theorem 2 can be removed, but we leave this as an open problem. As partial progress we show that this is true for the following (more restricted) model of priority algorithms, in which an algorithm can only consider and allocate sets whose values are not implied by the values of other sets.

Elementary bids as items. Consider an auction setting in which agents do not provide entire valuation functions, but rather each agent specifies a list of *desired sets* $S_1, \ldots, S_k$ and a value for each one. Moreover, each agent receives either a desired set or the empty set. This can be thought of as an auction with a succinct representation for valuation functions, in the spirit of the XOR bidding language [29]. We model such an auction as a priority algorithm by considering items to be the bids for desired sets. In such a setting, the specified set-value pairs are called *elementary bids*. We say that the priority model uses *elementary bids as items* when only elementary bids $(i, S, v(S))$ can be considered by the algorithm. For each item $(i, S, v(S))$, the decision to be made is whether or not S will be the final set allocated to agent i; that is, whether or not the elementary bid for S will be "satisfied." In particular, unlike in the sets as items model, we do not permit the algorithm to build up an allocation incrementally by accepting many elementary bids from a single agent.

We now show that the greediness assumption from Theorem 2 can be removed when we consider priority algorithms in the elementary bids as items model.

Theorem 3. *Suppose A is an incentive compatible priority algorithm for the CA problem that uses elementary bids as items. Then A cannot approximate the optimal social welfare by a factor of $(1-\delta)k$ for any $\delta > 0$.*

3.2 Bidders as Items

Roughly speaking, the lower bounds in Theorems 2 and 3 follow from a priority algorithm's inability to determine which of many different mutually-exclusive desires of an agent to consider first when constructing an allocation. One might guess that such difficulties can be overcome by presenting an algorithm with more information about an agent's valuation function at each step. To this end, we consider an alternative model of priority algorithms in which the agents themselves are the items, and the algorithm is given complete access to an agent's declared valuation function each round.

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. The decision to be made for item (i, v_i) is a set $S \subseteq M$ to assign to bidder i. The truthful greedy CA mechanism for single-minded bidders falls within this model, as does its (non-truthful) generalization to complex bidders [25], the primal-dual algorithm of [8], and the (first) algorithm of [4] for multi-unit CAs. We now establish an inapproximation bound for truthful priority allocations that use bidders as items.

Theorem 4. *Suppose A is an incentive compatible priority algorithm for the (2-minded) CA problem that uses bidders as items. Than A cannot approximate the optimal social welfare by a factor of $\frac{(1-\delta)k}{2}$ for any $\delta > 0$.*

4 Truthful Submodular Priority Auctions

Lehmann, Lehmann, and Nisan [24] proposed a class of greedy algorithms that is well-suited to auctions with submodular bidders; namely, objects are considered in any order and incrementally assigned to greedily maximize marginal utility. They showed that any ordering of the objects leads to a 2-approximation of social welfare, but not every ordering of objects leads to an incentive compatible algorithm. However, this does not preclude the possibility of obtaining truthfulness using some adaptive method of ordering the objects.

We consider a model of priority algorithms which uses the m objects as input items. In this model, an item will be represented by an object x, plus the value $v_i(x|S)$ 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 is incentive compatible.

Theorem 5. *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.*

5 Future Work

The goal of algorithmic mechanism design is the construction of algorithms in situations where inputs are controlled by selfish agents. We considered this fundamental issue in the context of conceptually simple methods (independent of time bounds) rather than in the context of time constrained algorithms. Our results concerning priority algorithms (as a model for greedy mechanisms) is a natural beginning to a more general study of the power and limitations of conceptually simple mechanisms. Even though the priority framework represents a restricted (albeit natural) algorithmic approach, there are still many unresolved questions even for the most basic mechanism design questions. In particular, we believe that the results of Section 3 can be unified to show that the linear inapproximation bound holds for all priority algorithms (without restrictions). The power of greedy algorithms for unit-demand auctions (s-CAs with $s = 1$) is also not understood; it is not difficult to show that optimality cannot be achieved by priority algorithms, but is it possible to obtain a sublinear approximation bound with greedy methods? Even though an optimal polytime algorithm exists for this case, greedy algorithms for the problem are still of interest, evidenced by the use of greedy algorithms in practice to resolve unit-demand AdWord auctions.

An obvious direction of future work is to widen the scope of a systematic search for truthful approximation algorithms; priority algorithms can be extended in many ways. One might consider priority algorithms with a more esoteric input model, such as a hybrid of the sets as items and bidders as items models. Priority algorithms can be extended to allow revocable acceptances [18] whereby a priority algorithm may "de-allocate" sets or objects that had been previously allocated to make a subsequent allocation feasible. Somewhat related is the priority stack model [5] (as a formalization of local ratio/primal dual algorithms [3]) where items (e.g. bidders or bids) initially accepted are placed in a stack and then the stack is popped to ensure feasibility. This is similar to algorithms that allow a priority allocation algorithm to be followed by some simple "cleanup" stage [20]. Another possibility is to consider allocations that are comprised of taking the best of two (or more) priority algorithms. A special case that has been used in the design of efficient truthful combinatorial auction mechanisms [4, 8, 28] is to optimize between a priority allocation and the naïve allocation that gives all objects to one bidder. Another obvious extension is to consider randomized priority algorithms, potentially in a Bayesian setting. Finally, one could study more general models for algorithms that implement integrality gaps in LP formulations of packing problems; it would be of particular interest if a deterministic truthful k-approximate mechanism could be constructed from an arbitrary packing LP with integrality gap k, essentially derandomizing the construction of Lavi and Swamy [23].

The results in this paper have thus far been restricted to combinatorial auctions but *the basic question* being asked applies to all mechanism design problems. Namely, when can a conceptually simple approximation to the underlying combinatorial optimization problem be converted into an incentive compatible mechanism that achieves (nearly) the same approximation? For example, one might consider the power of truthful priority mechanisms for approximating unrelated machines scheduling, or for more general integer packing problems.

Acknowledgements

We thank S. Dobzinski, R. Gonen, and S. Micali for helpful discussions.

References

1. Y. Azar, I. Gamzu, and S. Gutner. Truthful unsplittable flow for large capacity networks. In *Proc. 19th ACM Symp. on Parallel Algorithms and Architectures*, 2007.
2. M. Babaioff and L. Blumrosen. Computationally-feasible truthful auctions for convex bundles. In *Proc. 7th Intl. Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, 2004.
3. A. Bar-Noy, R. Bar-Yehuda, A. Freund, J. Naor, and B. Schieber. A unified approach to approximating resource allocation and scheduling. *Journal of the ACM*, 48:1069–1090, 2001.
4. Y. Bartal, R. Gonen, and N. Nisan. Incentive compatible multi unit combinatorial auctions. In *Proc. 9th Conf. on Theoretical Aspects of Rationality and Knowledge*, 2003.
5. A. Borodin, D. Cashman, and A. Magen. How well can primal-dual and local-ratio algorithms perform? In *Proc. 32nd Intl. Colloq. on Automata, Languages and Programming*, 2005.
6. A. Borodin and B. Lucier. Price of anarchy for greedy auctions. In *Proc. 21th ACM Symp. on Discrete Algorithms*, 2010.
7. A. Borodin, M. N. Nielsen, and C. Rackoff. (incremental) priority algorithms. In *Proc. 13th ACM Symp. on Discrete Algorithms*, 2002.
8. P. Briest, P. Krysta, and B. Vöcking. Approximation techniques for utilitarian mechanism design. In *Proc. 36th ACM Symp. on Theory of Computing*, 2005.
9. D. Buchfuhrer, S. Dughmi, H. Fu, R. Kleinberg, E. Mossel, C. Papadimitriou, M. Schapira, Y. Singer, and C. Umans. Inapproximability for vcg-based combinatorial auctions. In *Proc. 21th ACM Symp. on Discrete Algorithms*, 2010.
10. C. Chekuri and I. Gamzu. Truthful mechanisms via greedy iterative packing. In *Proc. 12th Intl. Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, 2009.
11. G. Christodoulou, A. Kovács, and Michael Schapira. Bayesian combinatorial auctions. In *Proc. 35th Intl. Colloq. on Automata, Languages and Programming*, pages 820–832, 2008.
12. S. Dobzinski, N. Nisan, and M. Schapira. Approximation algorithms for combinatorial auctions with complement-free bidders. In *Proc. 36th ACM Symp. on Theory of Computing*, 2005.
13. S. Dobzinski and M. Schapira. An improved approximation algorithm for combinatorial auctions with submodular bidders. In *Proc. 17th ACM Symp. on Discrete Algorithms*, 2006.
14. S. Dobzinski and M. Sundararajan. On characterizations of truthful mechanisms for combinatorial auctions and scheduling. In *Proc. 10th ACM Conf. on Electronic Commerce*, 2008.

15. R. Gonen and D. Lehmann. Optimal solutions for multi-unit combinatorial auctions: Branch and bound heuristics. In *Proc. 2nd ACM Conf. on Electronic Commerce*, 2000.
16. J. Hastad. Clique is hard to approximate within $n^{1-\epsilon}$. *Electronic Colloquium on Computational Complexity*, 38, 1997.
17. R. Holzman, N. Kfir-Dahav, D. Monderer, and M. Tennenholtz. Bundling equilibrium in combinatiorial auctions. *Games and Economic Behavior*, 47:104–123, 2004.
18. S. Horn. One-pass algorithms with revocable acceptances for job interval selection. University of Toronto MSC thesis, 2004.
19. S. Khot, R. Lipton, E. Markakis, and A. Mehta. Inapproximability results for combinatorial auctions with submodular utility functions. In *Workshop on Internet and Network Economics (WINE)*, 2005.
20. P. Krysta. Greedy approximation via duality for packing, combinatorial auctions and routing. In *Proc. 30th Intl. Symp. on Mathematical Foundations of Computer Science*, 2005.
21. R. Lavi, A. Mu'alem, and N. Nisan. Towards a characterization of truthful combinatorial auctions. In *Proc. 44th IEEE Symp. on Foundations of Computer Science*, 2003.
22. R. Lavi and N. Nisan. Online ascending auctions for gradually expiring goods. In *Proc. 16th ACM Symp. on Discrete Algorithms*, 2005.
23. R. Lavi and C. Swamy. Truthful and near-optimal mechanism design via linear programming. In *Proc. 46th IEEE Symp. on Foundations of Computer Science*, 2005.
24. B. Lehmann, D. Lehmann, and N. Nisan. Combinatorial auctions with decreasing marginal utilities. In *Proc. 3rd ACM Conf. on Electronic Commerce*, 2001.
25. D. Lehmann, L. I. O'Callaghan, and Y. Shoham. Truth revelation in approximately efficient combinatorial auctions. In *Proc. 1st ACM Conf. on Electronic Commerce*, pages 96–102. ACM Press, 1999.
26. R. Paes Leme and E. Tardos. Sponsored search equilibria for conservative bidders. In *Fifth Workshop on Ad Auctions*, 2009.
27. M. Mahdian and A. Saberi. Multi-unit auctions with unknown supply. In *Proc. 8th ACM Conf. on Electronic Commerce*, 2006.
28. A. Mu'alem and N. Nisan. Truthful approximation mechanisms for restricted combinatorial auctions. *Games and Economic Behavior*, 64:612–631, 2008.
29. N. Nisan. Bidding and allocation in combinatorial auctions. In *Proc. 2nd ACM Conf. on Electronic Commerce*, 2000.
30. C. Papadimitriou, M. Schapira, and Y. Singer. On the hardness of being truthful. In *Proc. 49th IEEE Symp. on Foundations of Computer Science*, 2008.
31. D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proc. 37th ACM Symp. on Theory of Computing*, 2006.