On the Hardness of Being Truthful

Christos Papadimitriou
Computer Science Division
University of California at Berkeley
CA, 94720 USA
christos@cs.berkeley.edu

Michael Schapira
School of Computer Science and Engineering
The Hebrew University
Jerusalem, Israel
mikesch@cs.huji.ac.il

Yaron Singer
Computer Science Division
University of California at Berkeley
CA, 94720 USA
yaron@cs.berkeley.edu

Abstract

The central problem in computational mechanism design is the tension between incentive compatibility and computational efficiency. We establish the first significant approximability gap between algorithms that are both truthful and computationally-efficient, and algorithms that only achieve one of these two desiderata. This is shown in the context of a novel mechanism design problem which we call the COMBINATORIAL PUBLIC PROJECT PROBLEM *(*CPPP*). CPPP is an abstraction of many common mechanism design situations, ranging from elections of kibbutz committees to network design.*

Our result is actually made up of two complementary results – one in the communication-complexity model and one in the computational-complexity model. Both these hardness results heavily rely on a combinatorial characterization of truthful algorithms for our problem. Our computational-complexity result is one of the first impossibility results connecting mechanism design to complexity theory; its novel proof technique involves an application of the Sauer-Shelah Lemma and may be of wider applicability, both within and without mechanism design.

1 Introduction

In real networks, routing is done by routers that choose paths other than shortest, and this results in distance matrices that do not satisfy the triangle inequality. In such a situation, it is often desirable to construct an *overlay:* A set of k nodes spread throughout the network and with high-quality routing between them, so that other nodes can improve the distance between them by routing through the closest overlay node. An overlay is beneficial to different nodes in different degrees, and we wish to design the overlay that maximizes total welfare.

This is a typical instance of a general problem we define, called COMBINATORIAL PUBLIC PROJECT PROBLEM, or CPPP. There are n *agents* and m *resources*, and, for each agent i, a *private valuation function* v_i which specifies i's value for every subset of the resources. We assume that all valuations are *nondecreasing and submodular* (a case that encompasses among many others the overlay network example above, see definitions in Subsec 1.1). The objective is to find a subset of the resources S of size k, where k is a parameter, which maximizes the *social welfare*, i.e., the sum of agents' values for the chosen subset $\Sigma_i\, v_i(S)$.

Computationally speaking, this problem is relatively benign; while many of its special cases are NP-hard [10] it is well known [22] that the greedy algorithm achieves a constant approximation ratio (specifically, $1 - \frac{1}{e}$).

But the fact that valuations are private presents us with a formidable problem: unless otherwise incentivized, agents are likely to *lie,* exaggerating the degree to which they prefer one alternative over another[1], and this misrepresentation makes optimization impossible. There is a very general method for providing incentives for the agents to reveal their true valuation, namely the *Vickrey-Clarke-Groves (VCG) mechanism* [34, 4, 14]. However, VCG requires that we solve *exactly* (typically

[1] According to a March 26 2008 Gallup poll, 28% of Clinton's supporters declare that they will not vote for Obama in a presidential election.

many instances of) the CPPP — an NP-hard problem. We could of course turn to approximation, as we always do when faced with intractability. The tragedy of this area is that *approximation and truthfulness do not mix:* Running VCG with approximate solutions is, in general, *not* incentive compatible [24].

In other words, efficient approximability and incentive compatibility seem to be at loggerheads. This tension underlies much of the work in algorithmic mechanism design [24, 25, 16, 5]. In this paper, we establish a huge gap between the quality of the solutions that can be obtained by algorithms for CPPP that are *both* polynomial *and* truthful, and by algorithms that satisfy only *one* of these two desiderata. We show this by proving that *no* truthful *and* computationally-efficient algorithm for CPPP can obtain any reasonable approximation ratio, specifically, a ratio better than $\sqrt{m}$ (this holds even for the case of two agents)[2]. Our inapproximability result settles a long-standing open question in algorithmic mechanism design [13, 29]: we exhibit a problem (CPPP) that is easy from a computational perspective (a constant approximation algorithm exists), and from an economic perspective (an optimal truthful algorithm exists), but is hard (does not allow for constant approximations) if we care about both[3].

Our result is actually made up of two complementary results – one in the communication-complexity model and one in the computational-complexity model. Technically speaking, each of these results must overcome two main challenges [29]: First, we must provide a combinatorial characterization of truthful algorithms. Then, once we have such a characterization, we can exploit it to prove an inapproximability result.

An interesting way to view our results is the following: Over the past four decades, complexity theory has been successful in classifying optimization problems into various classes, such as P, NP, NP-hard, and APX (those problems that can be approximated within some constant factor in polynomial time). Mechanism design is about *incentive-compatible optimization*, in which the inputs are provided by agents who have their own objectives. In this new regime, for social-welfare maximization problems,[4] classical VCG theory implies that P, NP, and NP-hardness are preserved under truthfulness. *Our computational-complexity result essentially states that APX is* not *preserved* (our communication-complexity result proves an analogous statement in the communication model).

Communication-complexity lower bound. Our first main contribution is an exponential lower bound on the amount of communication required by *any* truthful algorithm that approximates the CPPP within any reasonable ratio:

Theorem: *Any truthful algorithm for the* CPPP *that obtains an approximation ratio of* $O(m^{\frac{1}{2}-\epsilon})$ *requires communication that is exponential in* m *(for every* $\epsilon > 0$ *and even for* $n = 2$*).*

The first step of the proof is showing that any truthful algorithm for CPPP must be an affine maximizer. An affine maximizer is an algorithm defined by a fixed set of solutions, and which, given a set of valuations, picks the solution in the set that maximizes social welfare. A celebrated result by Roberts [28] essentially states that if an algorithm is truthful for all valuations, then it has to be an affine maximizer. However, this result does not necessarily hold when restricted to special settings (submodular valuations, etc.). In fact, for many restricted settings, truthful algorithms that are *not* affine maximizers are known (e.g. [3, 2]). In the present case, by carefully applying machinery from Roberts's (long and delicate) proof — actually, its recent interpretation by Lavi, Mu'alem and Nisan [16, 17] — we are able to *prove that any truthful algorithm for the nondecreasing submodular case of the* CPPP *is an affine maximizer.* The proof explores the geometric and topological properties that must hold for any truthful algorithm for CPPP to prove affine maximization.

After providing this combinatorial characterization of truthfulness the second step is proving a lower bound for affine maximizers. We do this by proving a lower bound on the number of solutions in the range of the mechanism, and then exploiting affine maximization to to establish our communication-complexity lower bound. The proof of the theorem draws ideas from works in many different fields of research (e.g., [28, 26, 23, 16, 17, 5, 8, 11, 19]), and so we defer a discussion of the related work to Section 2.

Computational-complexity lower bound. Our previous result is a *communication-complexity* lower bound. Informally, the agents are assumed to compute their valuations based on exponentially large data, and then it is shown that too much of this data must be exchanged for the algorithm to be truthful. However, many interesting valuations depend on very succinct data (recall the introductory overlay problem, in which the input is a distance matrix), and *computational-complexity* techniques would be needed to establish lower bounds for these; *no such results had been known* (with the possible exception of Theorem 5 in [16]). Our next major result does

[2] We note that a simple truthful polynomial-time algorithm that obtains a $\sqrt{m}$ approximation ratio for CPPP shows that our result is tight in both models. [32]

[3] It is known that algorithms that are both truthful *and* computationally-efficient might not perform (in terms of approximation ratio) as well as algorithms that only meet *one* of these two requirements. However, it was unclear by how much their performance is harmed. In fact, so far researchers have only been able to establish a gap of 2 between these two types of algorithms [16, 8].

[4] See discussion about other optimization goals in [21].

exactly this: We consider a class of nonnegative submodular (though not nondecreasing) valuations, that are succinctly described. We show that, despite the fact that a $1-\frac{1}{e}$ approximation exists for this class, no truthful and polynomial-time algorithm can obtain an approximation ratio (asymptotically) better than $\sqrt{m}$.

As in our communication-complexity result, the proof of this result consists of two parts: Proving that any truthful algorithm must be an affine-maximizer, and proving a hardness result for affine-maximizers. The main challenge now lies in the second part (the first part is basically derived from the characterization of truthfulness in our communication complexity result). The heart of our proof is therefore showing the following inapproximability result for a class of succinct submodular functions we call "*coverage-penalty valuations*":

Theorem: *For* CPPP *with coverage-penalty valuations there is no polynomial-time affine maximizer with* $O(m^{\frac{1}{2}-\epsilon})$ *approximation ratio unless NP* $\subseteq$ *BPP (for every* $\epsilon > 0$*).*

To establish this, we first show, very much as in the proof of our communication-complexity result, an exponential lower bound on the number of solutions, and then we use a probabilistic version of the Sauer-Shelah Lemma [30, 33][5] to obtain a hardness result. This result has interesting projections on complexity theory as discussed in section 3.

Organization of the paper. In the balance of this section we introduce the CPPP and the communication- and computational-complexity notions. In the next section we prove our exponential communication-complexity lower bound, while in Section 3 we prove our computational-complexity lower bound. In Section 4 we discuss our results and present several open questions.

1.1 Definitions

In CPPP there is a set of n *agents* $\{1,...,n\}$, and a set of m *resources* $\{1,...,m\}$. Each agent has a *private valuation function* $v_i : 2^{[m]} \rightarrow \Re_{\geq 0}$. We denote by V_i the space of possible valuations of agent i, and by V the *domain* of valuations $V_1 \times \ldots \times V_n$. We shall assume that $v_i(\emptyset) = 0$. We assume that every v_i is *submodular*, i.e., for every $S, T \subseteq [m]$ $v_i(S \cup T) + v_i(S \cap T) \leq v_i(S) + v_i(T)$. Submodularity is known to be equivalent to the following easily verifiable property, called "*decreasing marginal utilities*": For every $S \subset T \subseteq [m]$, and for every $j \in [m]$ such that $j \notin T$ $v_i(S \cup \{j\}) - v_i(S) \geq v_i(T \cup \{j\}) - v_i(T)$. Submodularity arises in many contexts – both economic and computational (see [18, 6, 7, 9, 12, 11, 35] and references therein). This paper focuses on submodular functions that are *nondecreasing*[6] in that $S \subseteq T$ implies $v(S) \leq v(T)$. Nondecreasing submodular functions are known to have particularly good properties; for example, they can be approximated within a ratio of $1-\frac{1}{e}$ (for general nonnegative submodular functions a constant approximation ratio exists [11]).

The objective in the CPPP is to find a subset of size k, where k is a parameter of the problem, of resources which maximizes the *social-welfare*. That is, we wish to find $T \in argmax_{S \subseteq [m],\ |S|=k} \Sigma_i v_i(S)$.

We are interested in algorithms (*mechanisms*) for CPPP satisfying three desiderata:

Quality of solution. We want our mechanisms to return a solution (set of resources) whose social welfare is as close, in terms of ratio, to the optimum as possible.

Computational efficiency. Our algorithms should run in time that is polynomial in the natural parameters of the problem – m and n. However, as the "input" (the data enabling each agent to compute the valuation) can be exponential in m we must specify how it can be accessed. In mechanism design one often takes a "black box" approach (see [6]): We assume that valuations are computed by an oracle that can answer a certain type of queries, and we restrict algorithms to ask a polynomial number (in n and m) of such queries. There are two common types of queries:

- In the *value query model* a query is a subset of resources $S \subseteq [m]$, and the answer is simply $v_i(S)$; This weaker model is mainly used for designing algorithms.
- The *general query* model is equivalent to *Yao's communication model* [36, 15], in which the agents take turns announcing messages; a message by agent i is any function (even a computationally intractable one) of the values of v_i and of the previous messages. We use this stronger model for impossibility results.

A different approach would be to consider cases in which the "input" (valuations) can be concisely represented, i.e., can be encoded in a natural way that is polynomial in m and n. We follow this approach in Section 3.

Truthfulness. We want an algorithm (*mechanism*) A to be such that the agents are rationally motivated to truthfully answer the algorithm's queries. This is achieved by a *payment function* p which, for every n-tuple of valuation functions $v = (v_1, ...v_n) \in V$, demands a payment

[5] This lemma is closely related to the notion of the Vapnik-Chervonenkis (VC) dimension.

[6] We slightly relax this assumption in Section 3.

from each agent. p is such that no agent can increase his utility (the value of the set chosen by the algorithm minus the payment assigned to him) by misreporting his valuation[7]. Formally, for every $i \in [n]$ we have that:

$$\forall v_i, v_i' \in V_i, \forall v_{-i} \in V_{-i},$$

$$v_i(A(v_i, v_{-i})) - p(v_i, v_{-i}) \geq v_i(A(v_i', v_{-i})) - p(v_i', v_{-i}),$$

where V_{-i} is the cartesian product of all V_j's such that $j \neq i$, (v_i, v_{-i}) is the valuations profile in which i has v_i and the other agents have v_{-i}, (v_i', v_{-i}) is defined similarly, and $A(v)$ is the set A outputs for the valuation profile v.

2 Communication-Complexity of Mechanism Design

In this section we prove the following:

Theorem 2.1 *Any truthful algorithm for* CPPP *that obtains an approximation ratio of $O(m^{\frac{1}{2}-\epsilon})$ requires exponential communication (for every $\epsilon > 0$ and even for $n = 2$).*

The proof proceeds in two steps: We first establish (Lemma 2.3) that any truthful algorithm for the CPPP must be of a very restricted kind called an *affine maximizer*. We then show that any affine maximizer for the CPPP requires exponential communication (Lemmas 2.16 and 2.17). This second part uses techniques introduced by Dobzinski and Nisan [5] for proving communication complexity lower bounds for affine maximizers.

The Characterization Lemma is the heart of our proof. It establishes that the CPPP has a rich enough structure, and appropriately strong interaction between the agents, so that a subtle variant of Roberts's proof [28] is enabled. The line of argument used in our proof follows that of Roberts', as presented by Lavi, Mu'alem, and Nisan [17]. However, applying this machinery to our setting is not straightforward as we are dealing with a *restricted* domain of valuation functions (submodular, nondecreasing). Therefore, we must handle various technical difficulties: We must ensure that the valuation functions we construct belong to this restricted domain. We must also take measures to deal with the fact that our valuation domain is *not open* (in the standard topological sense). Unlike Roberts' proof, our proof makes repetitive use of the *strong monotonicity* constraint introduced in [16] (also see [8]).

[7] The notion of truthfulness we consider is the standard notion of truthfulness in dominant strategies. All our results apply to the weaker notion of truthfulness in ex-post Nash.

2.1 The Characterization Lemma

Let A be an algorithm for the nondecreasing submodular CPPP with m resources and parameter k. We define A's *range* R_A to be the resource subsets of size k that are output by A for *some* input, i.e., $R_A = \{S | \exists v = (v_1, ..., v_n) \; s.t. \; A(v) = S\}$. Informally, A is an affine-maximizer if it *always* optimizes over its entire range R_A.

Definition 2.2 ([28]) *An algorithm A is said to be an affine maximizer if there exist nonnegative* agent weights $w_1, ..., w_n$ *(not all equal to 0), and* outcome weights $\{C_S\}_{S \in R_A}$ *such that*

$$\forall v = (v_1, ..., v_n)$$

$$A(v) \in argmax_{S \in R_A}[(\Sigma_i w_i v_i(S)) + C_S].$$

Lemma 2.3 *Any truthful algorithm for the nondecreasing submodular* CPPP *with $n = 2$ is an affine-maximizer.*

Remark 2.4 *We note that, as we are aiming for an impossibility result, we prove the characterization lemma only for the 2-agent case. We shall later show that our inapproximability result holds even for this special case.*

Proof: As is common in such proofs (see for example [28, 17]), we study the topological structure of vectors of *valuation differences*. For any pair of valuation functions $v = (v_1, v_2)$, we shall denote by $v(S) - v(T)$ the vector in $\Re^2$ $(v_1(S) - v_1(T), v_2(S) - v_2(T))$. We also define

$$P(S,T) = \{\alpha \in \Re^2 \; | \exists \; v \; s.t. \; A(v) = S \; and$$

$$v(S) - v(T) = \alpha\}.$$

If A is an affine maximizer that outputs a set S for the valuation functions $v = (v_1, v_2)$ then it must hold that for every $T \neq S$ in R_A (for some *fixed* agent-weights w_1, w_2 and outcome-weights $\{C_R\}_{R \in R_A}$)

$$(\Sigma_i w_i v_i(S)) + C_S \geq (\Sigma_i w_i v_i(T)) + C_T,$$

which implies that

$$\Sigma_i w_i (v_i(S) - v_i(T)) + (C_S - C_T) \geq 0.$$

Informally, observe that the inequalities above suggest that if A is an affine maximizer, then there is a line l in $\Re^2$ of the form $w_1 x + w_2 y = 0$ such that *every* $P(S,T)$ has l as its lower boundary (possibly shifted by some constant $\gamma(S,T) = C_S - C_T$ from the centre of

the axes). So, to prove that a truthful algorithm A is an affine maximizer, we need to show that there are weights w_1, w_2 (and $\{C_R\}_R$) that induce such a line l (the same l for all choices of $S \neq T \in R_A$). This is precisely what we mean to show. The reader is referred to [17] for an explanation of the geometric intuition behind the proof of Roberts' Theorem (which also underlies our proof).

Strong monotonicity. We shall require the strong monotonicity property.

Definition 2.5 *An algorithm satisfies strong-monotonicity if for every $i \in [n], v_i, v_i' \in V_i$, and $v_{-i} \in V_{-i}$, if $A(v_i, v_{-i}) = S$ and $A(v_i', v_{-i}) = T \neq S$ then it must hold that $v_i(S) - v_i(T) > v_i'(S) - v_i'(T)$.*

Not any truthful algorithm is necessarily strongly-monotone (and vice-versa), yet we shall prove the theorem for strongly-monotone algorithms. At the end of the proof we shall revisit this assumption and explain why it can be removed. The following proposition shall play a crucial role in our proofs.

Proposition 2.6 *Let A be a strongly monotone algorithm, and let $v \in V$ be such that $A(v) = S$. If $v' \in V$ and $v(S) - v(T) \leq v'(S) - v'(T)$ (i.e., $\leq$ in each coordinate) then $A(v') \neq T$.*

Proof: Assume, for point of contradiction, that the conditions stated in the theorem hold and $A(v') = T$. Let $v = (v_1, v_2)$ and $v' = (v_1', v_2')$. Let $\alpha_1 = v_1(S) - v_1(T)$. We shall prove the proposition for the case that $\alpha_1 \geq 0$ (the other case requires a very similar construction). Let $j \in S \setminus T$ (since all sets are of equal size such a j is guaranteed to exist). We define a valuation function $v_1'' \in V_1$ as follows (for some arbitrarily large $\beta > 0$):

$$\forall R \;\; v_1''(R) = \alpha_1 |R \cap \{j\}| + \beta |S||T| - \beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

It is easy to verify that this is indeed a nondecreasing submodular function (this construction is inspired by [11, 19]). We shall now show that $A(v_1'', v_2) = S$. This is because if $A(v_1'', v_2) = Q \neq S$ then, by strong monotonicity, $v_1(S) - v_1(Q) > v_1''(S) - v_1''(Q)$. It is easy to show that if $Q \neq S, T$ then this results in a contradiction (as we can set the value of β to be as high as we like). Observe that if we set $Q = T$ then this too results in a contradiction. Similarly, we can show that $A(v_1'', v_2') = T$. However, in this case by strong monotonicity we have that $v_2(S) - v_2(T) > v_2'(S) - v_2'(T)$. A contradiction. □

Main part of the proof.

Claim 2.7 *Let $\alpha \in P(S,T)$ for some $S, T \in R_A$. Let $\epsilon = (\epsilon_1, \epsilon_2) \geq 0$. Then, $\alpha + \epsilon \in P(S,T)$.*

Proof: Since $\alpha = (\alpha_1, \alpha_2) \in P(S,T)$ then, by definition, there are valuation functions $v = (v_1, v_2)$ such that $A(v) = S$ and $v(S) - v(T) = \alpha$. We prove the claim for the case $\alpha \geq 0$ (other cases are handled similarly). Let $j \in S \setminus T$. We define valuation functions $v' = (v_1', v_2')$ as follows:

$$\forall R \;\; v_1'(R) = (\alpha_1 + \epsilon_1)|R \cap \{j\}| + \beta |S||T| - \beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

$$\forall R \;\; v_2'(R) = (\alpha_2 + \epsilon_2)|R \cap \{j\}| + \beta |S||T| - \beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

The use of strong monotonicity (as in the proof of Proposition 2.6) and of Proposition 2.6 itself shows that $A(v_1', v_2) = S$. Similarly, we can then show that $A(v_1', v_2') = S$. Observe that $v'(S) - v'(T) = \alpha + \epsilon$. Therefore, by definition, $\alpha + \epsilon \in P(S,T)$. □

Claim 2.7 tells us something important about the structure of the different $P(S,T)$ sets in $\Re^2$: If a point is in $P(S,T)$ then so are all points "above" it and "to the right" of it. Hence, each $P(S,T)$ is defined by a lower boundary with a nonincreasing slope. However, we do not yet know that this nonincreasing slope is a straight line (and certainly not that it is the same straight line for all choices of S, T). We shall require the two following technical propositions:

Proposition 2.8 *For any $S \neq T \in R_A$ $\alpha \in P(S,T)$ iff $-\alpha \notin P(T,S)$.*

Proof: Let $\alpha = (\alpha_1, \alpha_2) \in P(S,T)$. Then, by definition, there exists $v = (v_1, v_2)$ such that $A(v) = S$ and $v(S) - v(T) = \alpha$. Suppose, for point of contradiction, that $-\alpha \in P(T,S)$. Then, by definition, there exists $v' = (v_1', v_2')$ such that $A(v) = T$ and $v'(T) - v'(S) = -\alpha$ (or, $v'(S) - v'(T) = \alpha$). However, by Proposition 2.6 this is impossible ($A(v')$ cannot equal T).

We now prove the other direction, which is equivalent to showing that if $\alpha \notin P(S,T)$ then $-\alpha \in P(T,S)$. Since S is in R_A there must be valuation functions $v = (v_1, v_2)$ such that $A(v) = S$. Let $\alpha^W = v(S) - v(W)$. We prove the proposition for the case $\alpha \geq 0$ (other cases are handled similarly). Let $j \in S \setminus T$. We define valuation functions $v' = (v_1', v_2')$ as follows (we choose β to be huge, and in particular higher than the values of all coordinates of all the different α^W's):

$$\forall R \;\; v_1'(R) = \alpha_1 |R \cap \{j\}| + \beta |S||T| - \beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

$$\forall R \;\; v_2'(R) = \alpha_2 |R \cap \{j\}| + \beta |S||T|$$

$$-\beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

The repeated use of Proposition 2.6 for every $W \in R_A$ such that $W \neq S, T$ shows that $A(v'_1, v'_2)$ must be in $\{S, T\}$. However, since $v'(S) - v'(T) = \alpha$ and $\alpha \notin P(S, T)$, it must be that $A(v') = T$. Observe that $v'(T) - v'(S) = -\alpha$. So, by definition of $P(T, S)$, $-\alpha \in P(T, S)$. □

Proposition 2.9 *For any $S, T, W \in R_A$, such that no two are equal, if $\alpha \in P(S, T)$ and $\alpha' \in P(T, W)$ then $\alpha + \alpha' \in P(S, W)$.*

Proof:

Let $\alpha = (\alpha_1, \alpha_2)$. Let $\alpha' = (\alpha'_1, \alpha'_2)$. Let Y be an additive valuation function such that $\forall j \notin S \cup T \cup W$ $Y(\{j\}) = 0$ and the two following properties hold:

- $\Sigma_{j \in S} Y(\{j\}) - \Sigma_{j \in T} Y(\{j\}) = \alpha_1$
- $\Sigma_{j \in T} Y(\{j\}) - \Sigma_{j \in W} Y(\{j\}) = \alpha'_1$

Observe that because the number of variables is greater than the number of equations such a function Y exists. Since S, T, W are of equal size we can also assume that Y only assigns nonnegative values (otherwise increase the value of each $j \in S \cup T \cup W$ by some identical large enough constant).

Similarly, let Z be an additive valuation function such that $\forall j \notin S \cup T \cup W$ $Z(\{j\}) = 0$ and the two following properties hold:

- $\Sigma_{j \in S} Z(\{j\}) - \Sigma_{j \in T} Z(\{j\}) = \alpha_2$
- $\Sigma_{j \in T} Z(\{j\}) - \Sigma_{j \in W} Z(\{j\}) = \alpha'_2$

We define (for some huge β to be determined later) the following valuations $v' = (v'_1, v'_2)$:

$$\forall R \;\; v'_1(R) = Y(R) + \beta|S||T| -$$
$$\beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$
$$\forall R \;\; v'_2(R) = Z(R) + \beta|S||T| -$$
$$\beta(|S| - |S \cap R|)(|T| - |T \cap R|)$$

Since $\alpha \in P(S, T)$ there must be some valuations $v = (v_1, v_2)$ such that $A(v) = S$. The repeated use of Proposition 2.6 (as in the proof of Proposition 2.8) shows that $A(v') \in \{S, W\}$. Similarly, by taking advantage of the fact that $\alpha' \in P(T, W)$ one can show (using similar arguments) that $A(v') \in \{S, T\}$. We conclude that $A(v') = S$. Observe that $v'(S) - v'(W) = (v'(S) - v'(T)) + (v'(T) - v'(W)) = \alpha + \alpha'$. Hence, by definition of $P(S, W)$ $\alpha + \alpha' \in P(S, W)$. □

We shall prove our result for the case that $\overrightarrow{0} = (0, 0) \in P(S, T)$ for every $S \neq T \in R_A$. This greatly simplifies the exposition and enables us to convey the main idea of the proof. The proof for the more general case is achievable via the exact same logic as presented in [17] (claim 4 and the following claims in the first proof in that paper follow from what we have proven thus far).

Corollary 2.10 *For every $S \neq T, U \neq W \in R_A$ it holds that $P(S, T) = P(U, W)$.*

Proof: Let $\alpha \in P(S, T)$. As $\overrightarrow{0} \in P(T, W)$, we have that $\alpha = \alpha + \overrightarrow{0} \in P(S, W)$. We also know that $\overrightarrow{0} \in P(U, S)$, and so $\alpha = \overrightarrow{0} + \alpha \in P(U, W)$. □

That is, all the $P(S, T)$ sets (for every choice of S, T) are, in fact, the very same set, that we shall refer to as X. We shall now prove that X is convex, thus showing that the (lower) boundary of X (which we know is non-increasing) must be a straight line[8]. Our proof for the convexity of X relies on the assumption that the domain of valuations V is open. Unfortunately, it is *not* true that the domain of nondecreasing submodular valuations is open. At the end of the Characterization Lemma's proof we revisit this assumption and show how it too can be removed.

Definition 2.11 *A domain of valuations V is open if for each $v = (v_1, ..., v_n) \in V$, there is some $\epsilon > 0$ such that for all $v' = (v'_1, ..., v'_n)$, if $\forall S \subseteq M$ and $\forall i \in [n]$, $|v'_i(S) - v_i(S)| \leq \epsilon$ then $v' \in V$.*

Proposition 2.12 *X is convex.*

Proof: We first show that if $\alpha, \alpha' \in X$ then $\frac{\alpha + \alpha'}{2} \in X$. Suppose, by contradiction, that $\alpha, \alpha' \in X$ but $\frac{\alpha + \alpha'}{2} \notin X$. By Proposition 2.9 $\alpha + \alpha' \in X$, and by Proposition 2.8 $-\frac{\alpha + \alpha'}{2} \in X$. However, Proposition 2.9 now implies that $\frac{\alpha + \alpha'}{2} = (\alpha + \alpha') + (-\frac{\alpha + \alpha'}{2}) \in X$. A contradiction.

By repeatedly using this fact, we can, for any $\alpha, \alpha' \in X$ and $\lambda \in (0, 1)$ build a series of points that approach $\lambda\alpha + (1 - \lambda)\alpha'$, such that any point in the series has a ball of small radius that is fully contained in X. This (and the openness of V) suffices to prove that $\lambda\alpha + (1 - \lambda)\alpha' \in X$. □

Now we know that X is convex and therefore has a lower boundary in the form of a straight line l. Moreover, l goes through the origin of the axes $\overrightarrow{0}$, as by

[8] Observe that if we define $-X = \{-\alpha | \; \alpha \in X\}$ then the convexity of X implies the convexity of $-X$. If X and $-X$ are convex, their union is $\Re^2$, and their interiors are disjoint (by Proposition 2.8), then they must be separated by a straight line.

Proposition 2.8 $\alpha \in X$ iff $-\alpha \notin X$. So, l can be described by $w_1 x + w_2 y = 0$ for some positive constants w_1, w_2 (recall that l's slope is nonincreasing). Therefore, we get that A is an affine maximizer (for agent-weights w_1, w_2 and by setting all outcome-weights to be 0). This concludes the proof of the lemma.

□

Removing the Strong Monotonicity and Open Domain Assumptions. So far, the proof of the Characterization Lemma relied on the assumptions that the algorithms in question are strongly monotone and that the domain of valuations is open. Here we exploit the following machinery to do away with these:

Theorem 2.13 *[16] If a domain of valuations V is open, then for every truthful algorithm A there exists an algorithm A' that satisfies strong monotonicity such that if A' is an affine maximizer then so is A.*

Theorem 2.13 implies the following modus operandi: We shall focus on a "rich" open subdomain of the domain of all nondecreasing submodular valuation functions. We shall show that all our arguments actually apply to that domain. Since in that domain truthfulness is equivalent to strong monotonicity (in the sense stated by Theorem 2.13) this will conclude the proof. We now provide a way to slightly tweak all constructions of valuation functions in Subsections 2.1 and 2.2 to ensure that they all belong to this open domain.

Specifically, we define the *strict domain* V^{strict} to be the domain of all valuations that are *strictly* nondecreasing (i.e., $\forall S \subset T \subseteq [m]$ and $\forall i \in [n]$ $v_i(S) < v_i(T)$) and have *strictly* decreasing marginal utilities. (i.e., $\forall S \subset T \subseteq [m]$, $\forall j \notin T$ and $\forall i \in [n]$ $v_i(S \cup \{j\}) - v_i(S) > v_i(T \cup \{j\}) - v_i(T)$).

Claim 2.14 *V^{strict} is an open domain.*

Proof: Consider some $v = (v_1, ..., v_n) \in V^{strict}$. Let δ be the minimal gap caused by the above strict inequalities (taken over all possible sets of resources, agents, etc.). It is easy to see that $\epsilon = \frac{\delta}{3}$ meets the requirement in the definition of an open domain. □

We now show that any nondecreasing and submodular valuation function can be slightly tweaked to fit in V^{strict}. Hence, we can prove our theorem for the case that all valuations ar slightly tweaked in this manner. The reader may verify that (for sufficiently small choices ϵ) all the arguments in this section also apply to these slightly different constructions of valuation functions.

Claim 2.15 *For any nondecreasing and submodular valuation function $v_i \in V_i$, and any $\epsilon > 0$, there exists a valuation function $v_i' \in V_i$ that is strictly nondecreasing and has strictly decreasing marginal utilities such that $\forall S \subseteq [m]$ $|v_i(S) - v_i'(S)| \leq \epsilon$.*

Proof: Fix some $\epsilon > 0$ and valuation function v_i that is nondecreasing and submodular. Let $\epsilon' << \epsilon$. Let $g_{\epsilon'}$ be a valuation function such that $\forall R \subseteq [m]$ $g_{\epsilon'}(R) = |R|\epsilon' - \frac{2^{|R|}}{2^{m+1}}\epsilon'$. The reader can verify that this function is strictly nondecreasing and has strictly decreasing marginal utilities. Moreover, for any nondecreasing submodular valuation function v and for any ϵ' it holds that $v_i' = v_i + g_{\epsilon'}$ is *strictly* nondecreasing and has *strictly* decreasing marginal utilities. □

2.2 Lower Bound for Affine Maximizers

Let k, the number of chosen resources in the CPPP, be fixed to $\sqrt{m}$ throughout this subsection. We shall now show that any algorithm A that obtains a reasonable approximation ratio must have a range R_A of exponential size. We will then exploit affine maximization to show that the size of the range is a lower bound on the communication complexity.

Lemma 2.16 *For any algorithm A that obtains an approximation ratio of $m^{\frac{1}{2}-\epsilon}$ to the optimal social welfare it must hold that $|R_A| = \Omega(e^{m^\epsilon})$, even for $n = 1$.*

Proof: (Sketch) Consider an algorithm A with range R_A. Choose a set of resources $V \subseteq [m]$ by sampling $[m]$ independently with probability $p = m^{-\frac{1}{2}+\epsilon}$, where $\epsilon > 0$ is small and fixed, and consider the valuation function $v(S) = |S \cap V|$. We claim that, unless R_A contains at least $\Omega(e^{m^\epsilon})$ sets, with high probability it approximates v very poorly.

Indeed, it follows from the Chernoff bound that, for any small $\delta > 0$, with probability at least $1 - |R_A| \cdot \exp(\delta^2 m^{\frac{1}{2}+\epsilon})$, in this instance of CPPP the optimum $|V|$ is at least $(1-\delta)m^{\frac{1}{2}+\epsilon}$, while the solution returned by the affine maximizer will be at most $(1+\delta)m^\epsilon)$. The lemma follows. □

The following now concludes the proof of our main result:

Lemma 2.17 *Any affine maximizer A with range R_A requires $\Omega(|R_A|)$ communication, even for $n = 2$.*

Proof: (Sketch) Consider an affine maximizer A with range R_A. We shall construct two submodular valuation functions which require $\Omega(|R_A|)$ communication in order to achieve optimality in the range R_A. Recall that A maximizes $w_1 v_1(S) + w_2 v_2(S) + C_S$ over all $S \in R_A$; it is easy to see that one can assume $w_1 = w_2 = 1$

for all S, w.l.o.g. Now, suppose that each agent i has a private set $R_A^i \subseteq R_A$, which induces the following valuation function: $v_i(T) = \beta \cdot (\Pi_{S \in R_A^i} |S| - \Pi_{S \in R_A^i}(|S| - |S \cap T|))$ $\forall T \subseteq [m]$. Note that since β can be arbitrarily large, w.l.o.g. we can consider the case in which $C_S = 0$ for all $S \in R_A$. Observe that if A maximizes over the range R_A, it implicitly distinguishes between the case for which R_A^1 and R_A^2 intersect, and the case in which $R_A^1 \cap R_A^2 = \emptyset$. Therefore this is a reduction from the communication set-disjointness problem, establishing that $\Omega(|R_A|)$ communication is required. □

3 Computational-Complexity of Mechanism Design

We now describe a simple new technique for deriving *computational-complexity* (*not* communication-complexity) lower bounds for mechanism design problems, which we believe has much broader applicability. In particular, we show that, even if agents have succinctly described valuations, CPPP is inapproximable *in polynomial time* by truthful algorithms, unless *NP* $\subseteq$ *BPP*. Specifically, we prove the following theorem:

Theorem 3.1 *There is a class of succinctly-described, nonnegative and submodular valuation functions C such that:*

- *There is a polynomial-time algorithm for the* CPPP *with valuations in C that approximates the optimal social-welfare within a ratio of $1 - \frac{1}{e}$.*
- *Any truthful and polynomial-time algorithm cannot achieve an approximation ratio better than $m^{\frac{1}{2}-\epsilon}$ unless NP $\subseteq$ BPP (for every $\epsilon > 0$).*

We shall start by proving our lower bound for affine maximizers. We shall then show how this result can be extended to any truthful algorithm.

3.1 Lower Bound For Affine Maximizers.

We define a class of succinctly described valuation functions called *"coverage-penalty valuations"*. A coverage-penalty valuation $v_{F,T}$ is defined by a family $F = R_1, ...R_m$ of m subsets of a universe U, and a subset $T \subseteq [m]$: $\forall S \subseteq [m]$, $v_{F,T}(S) = |\bigcup_{j \in S} R_j| - \frac{\epsilon}{|T|}(\max\{0, |S \cap T| - \frac{|T|}{2}\})$ (for some extremely small value of ϵ). Intuitively, for every $S \subseteq [m]$, $v_{F,T}$ assigns a value that equals the number of elements in U covered by the union of the sets in F with indices in S, minus an insignificant "penalty" if S has "too many" elements in common with T. It is easy to verify that this function is indeed nonnegative[9] and submodular (but not nondecreasing).

Observe that for any value of k, it is easy to obtain a $1 - \frac{1}{e}$ approximation ratio for CPPP with coverage-penalty valuations (simply ignore the "penalty terms" in the definitions of the v_i's and apply the $1 - \frac{1}{e}$ greedy approximation algorithm). We shall now show that despite this fact, any affine-maximizer fails to obtain a reasonable approximation ratio:

Theorem 3.2 *No polynomial-time affine maximizer for* CPPP *with coverage-penalty valuations achieves an approximation ratio better than $m^{\frac{1}{2}-\epsilon}$ unless NP $\subseteq$ BPP (for every $\epsilon > 0$ and even for $n = 1$).*

Proof: (Sketch) Fix $k = \sqrt{m}$ and $n = 1$. Let A be an affine maximizer as in the statement of the theorem (w.l.o.g. assume that the agent-weights are 1 and outcome-weights are 0). R_A consists of subsets of $[m]$ of size $\sqrt{m}$ that are assigned by A to different inputs. Exactly as in Lemma 2.16, it can be shown that R_A must contain $\Omega(e^{m^\epsilon})$ sets. We now recall the Sauer-Shelah Lemma:

Lemma 3.3 *([30, 33]) For any family Z of subsets of a universe R, there is a subset Q of R of size $\Theta(\frac{\log |Z|}{\log |R|})$ such that for each $Q' \subseteq Q$ there is a $Z' \in Z$ such that $Q' = Z' \cap R$.*

Hence, coming back to the affine maximizer, there is a subset M' of $[m]$ of size $\Theta(\frac{m^\epsilon}{\log m})$ that is "shattered" by R_A. The idea is now to embed a small, but still polynomially large, instance of an NP-complete problem in M'. We shall do this for the problem in which we have a family F' of t subsets of a universe U and wish to determine whether there are $\frac{t}{2}$ sets in F' whose union covers the entire universe (which is known to be NP-complete). The embedding is as follows: We construct a coverage-penalty valuation function v for which each element in M' corresponds to one of the t subsets in F', and all other elements in $[m]$ correspond to empty sets. We set the "penalty set" T to equal M'. Now, observe that the value of the set output by the affine maximizer A equals $|U|$ *iff* there are $\frac{t}{2}$ sets in F' whose union covers U.

The above suggests a *non-uniform* reduction from an NP-hard problem to the problem of calculating the output of the affine maximizer in question. The reason the reduction is non-uniform (and thus it does not establish NP-completeness) is because *we do not know how to find M'* (see [27, 31, 20] on the complexity of making

[9] This may not be true if some of the R_i's are the empty set. However, this is easily handled by simply adding ϵ to $v_{F,T}(S)$ for every S.

Sauer-Shelah Lemma constructive). However, this non-uniform reduction is sufficient to show that if A obtains an approximation ratio better than $m^{\frac{1}{2}-\epsilon}$ in polynomial time then *NP has polynomial-size circuits*, i.e., NP $\subseteq$ P/*poly*.

By using the probabilistic version of the Sauer-Shelah Lemma presented by Ajtai [1] we can turn the reduction described above into a *probabilistic* polynomial-time reduction (thus concluding the proof of the theorem).

Lemma 3.4 *([1]) Let Z be a family of subsets of a universe R that is regular (i.e., all subsets in Z are of equal size) and $Q \geq 2^{|R|^\alpha}$ (for some $0 < \alpha \leq 1$). There are integers q, l (where $|R|, q$ and l are polynomially related) such that if we randomly choose q pairwise-disjoint subsets of R, $Q_1, ..., Q_q$, each of size l, then, w.h.p., for every function $f : [q] \rightarrow \{0, 1\}$ there is a subset $Z' \in Z$ for which $|Z' \cap Q_j| = f(j)$ for all $j \in [q]$.*

The probabilistic polynomial-time reduction from our NP-complete problem is very similar to the non-uniform reduction shown above. The key idea is finding pairwise-disjoint subsets of $[m]$ as in the statement of Lemma 3.4, and then associating each subset of the universe in the NP-complete problem with *all* elements in one of the pairwise-disjoint subsets. □

3.2 Connection to Complexity Theory

This technique has an interesting projection in complexity theory: Call a language $L \subseteq \{0,1\}^*$ *exponentially dense* if there is some $\alpha > 0$, and some integer N, such that for any integer $n > N$ it holds that $|L \cap \{0,1\}^n| \geq 2^{n^\alpha}$. For a language $L \subseteq \{0,1\}^*$, define SAT$_L$ to be the problem: "Given a CNF, is there a truth assignment *in* L that satisfies it?" The proof technique implies that:

Theorem 3.5 *Let L be any exponentially dense language. If* SAT$_L$ *is in P, then NP $\subseteq$ P/poly.*

Observe that we do not know how to relax the computational hardness assumption to NP $\subseteq$ BPP (e.g., via the probabilistic version of the Sauer-Shelah Lemma). For the problem CIRCUIT SAT, however, we can prove this stronger result via a different technique:

Theorem 3.6 *Let L be any exponentially dense language. If* CIRCUIT SAT$_L$ *is in P, then NP $\subseteq$ BPP.*

Proof: (Sketch) To solve a given CNF ϕ on n variables, start with a large enough N so that L contains at least 2^{n^2} strings of length N. Now *hash* these N bits (by a circuit computing a sampled universal hashing function) into n bits. With very high probability, the 2^{n^2} bitstrings in L of length N will cover, after hashing, all 2^n bitstrings of length n. Then feed these n bits into a verifier circuit for ϕ. It is not hard to see that, with high probability, this overall circuit, with N inputs, has a satisfying truth assignment in L if and only if ϕ is satisfiable. □

It would be very interesting to extend this idea (or the ideas in the proof of Lemma 3.4) to SAT$_L$.

3.3 Extension to All Truthful Algorithms

We extend our lower bound to all truthful algorithms by relying on the following simple observation: All the submodular functions that are constructed as part of the proof of the Characterization Lemma (Lemma 2.3 in Section 2) are, in fact, succinctly described. We can therefore define a class of succinctly described valuation functions C, that contains the families of functions constructed in Lemma 2.3 *and* all coverage-penalty valuations.

It is easy to see that an approximation of $1 - \frac{1}{e}$ is achievable for the CPPP when the valuation functions are in C (this is because coverage-penalty valuations are arbitrarily close to nondecreasing submodular valuations and all other valuations in C are nondecreasing and submodular). Because of the way we defined C one can show (by carefully applying the proof of Lemma 2.3 to this case) that any truthful algorithm for the CPPP with valuations in C is an affine-maximizer. We can now use Theorem 3.2 to obtain our result.

4 Discussion and Open Problems

What are the limitations of our techniques? It would be very interesting to explore for which mechanism design problems one can prove characterizations similar to that shown in Lemma 2.3 (for the CPPP), thus proving unconditional lower bounds. It would also be very interesting to extend the proof technique of Theorem 3.2 to other succinctly described settings. We believe that such computational hardness results can be shown for many other mechanism design problems, such as combinatorial auctions. Finally, a big computational-complexity-related open question is proving that mechanism design is computationally intractable under weaker computational assumptions. We suspect that this is possible, but the precise way of doing this eludes us at the moment.

Acknowledgements

We thank Moshe Babaioff, Liad Blumrosen, Venkatesan Guruswami, Luca Trevisan, Gregory Valiant and Umesh Vazirani for helpful discussions. We thank the anonymous reviewers for valuable comments.

References

[1] M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *STOC*, pages 10–19, 1998.

[2] A. Archer, C. Papadimitriou, K. Talwar, and E. Tardos. An approximate truthful mechanism for combinatorial auctions with single parameter agent. In *Proceedings of the 14th Annual ACM Symposium on Discrete Algorithms (SODA)*, 2003.

[3] Y. Bartal, R. Gonen, and N. Nisan. Incentive compatible multi unit combinatorial auctions. In *TARK 03*, 2003.

[4] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.

[5] S. Dobzinski and N. Nisan. Limitations of VCG-based mechanisms. In *STOC*, 2007.

[6] S. Dobzinski, N. Nisan, and M. Schapira. Approximation algorithms for combinatorial auctions with complement-free bidders. In *STOC*, 2005.

[7] S. Dobzinski and M. Schapira. An improved approximation algorithm for combinatorial auctions with submodular bidders. In SODA 06'.

[8] S. Dobzinski and M. Sundararajan. On characterizations of truthful mechanisms for combinatorial auctions and scheduling. In *ACM conference on electronic commerce (EC)*, 2008.

[9] U. Feige. On maximizing welfare when the utility functions are subadditive. In STOC 06'.

[10] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998.

[11] U. Feige, V. Mirrokni, and J. Vondrak. Maximizing non-monotone submodular functions. In *FOCS*, 2007.

[12] U. Feige and J. Vondrak. Approximation algorithms for allocation problems: Improving the factor of $1 - 1/e$. In FOCS 06'.

[13] J. Feigenbaum and 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*, pages 1–13, New York, 2002. ACM.

[14] T. Groves. Incentives in teams. *Econometrica*, pages 617–631, 1973.

[15] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[16] R. Lavi, A. Mu'alem, and N. Nisan. Towards a characterization of truthful combinatorial auctions. In *FOCS*, 2003.

[17] R. Lavi, A. Mu'alem, and N. Nisan. Two simplified proofs for Roberts' Theorem, 2004. Submitted.

[18] B. Lehmann, D. Lehmann, and N. Nisan. Combinatorial auctions with decreasing marginal utilities. In *ACM conference on electronic commerce (EC)*, 2001.

[19] V. Mirrokni, M. Schapira, and J. Vondrak. Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions, 2008. Working paper.

[20] E. Mossel and C. Umans. On the complexity of approximating the VC dimension. *J. Comput. Syst. Sci.*, 65(4):660–671, 2002.

[21] A. Mu'alem and M. Schapira. Setting lower bounds on truthfulness. In *SODA 2007*.

[22] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions ii. *Math. Programming Study 8*, pages 73–87, 1978.

[23] N. Nisan. The communication complexity of approximate set packing and covering. In ICALP 2002.

[24] N. Nisan and A. Ronen. Computationally feasible VCG-based mechanisms. In *ACM Conference on Electronic Commerce*, 2000.

[25] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behaviour*, 35:166 – 196, 2001. A preliminary version appeared in STOC 1999.

[26] N. Nisan and I. Segal. The communication requirements of efficient allocations and supporting prices. *JET*, 2006.

[27] C. H. Papadimitriou and M. Yannakakis. On limited nondeterminism and the complexity of the V-C dimension. *Journal of Computer and System Sciences*, 1996.

[28] K. Roberts. The characterization of implementable choice rules. In J.-J. Laffont, editor, *Aggregation and Revelation of Preferences. Papers presented at the 1st European Summer Workshop of the Econometric Society*, pages 321–349. North-Holland, 1979.

[29] T. Roughgarden. An algorithmic game theory primer. In *Proceedings of the 5th IFIP International Conference on Theoretical Computer Science (TCS). An invited survey.*, 2008.

[30] N. Sauer. On the density of families of sets. *J. Comb. Theory, Ser. A*, 13(1):145–147, 1972.

[31] M. Schäfer. Deciding the Vapnik-Cervonenkis dimension is Σ_3^p-complete. *Journal of Computer and System Sciences*, 58:177–182, 1999.

[32] M. Schapira and Y. Singer. Inapproximability of combinatorial public projects. Working paper, 2008.

[33] S. Shelah. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific J Math*, 41:247–261, 1972.

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

[35] J. Vondrak. Optimal approximation for the submodular welfare problem in the value oracle model. In *STOC*, 2008.

[36] A. C.-C. Yao. Some complexity questions related to distributive computing. In *ACM Symposium on Theory of Computing (STOC)*, pages 209–213, 1979.