Black-Box Reductions in Mechanism Design

Zhiyi Huang[1] ⋆, Lei Wang[2], and Yuan Zhou[3]

[1] University of Pennsylvania, Philadelphia PA 19104, USA,
hzhiyi@cis.upenn.edu
[2] Georgia Institute of Technology, Atlanta GA 30332, USA,
lwang@cc.gatech.edu
[3] Carnegie Mellon University, Pittsburgh PA 15213, USA,
yuanzhou@cs.cmu.edu

Abstract. A central question in algorithmic mechanism design is to understand the additional difficulty introduced by truthfulness requirements in the design of approximation algorithms for social welfare maximization. In this paper, by studying the problem of single-parameter combinatorial auctions, we obtain the first black-box reduction that converts *any* approximation algorithm to a truthful mechanism with essentially the *same* approximation factor in a prior-free setting. In fact, our reduction works for the more general class of *symmetric single-parameter* problems. Here, a problem is symmetric if its allocation space is closed under permutations.
As extensions, we also take an initial step towards exploring the power of black-box reductions for general single-parameter and multi-parameter problems by showing several positive and negative results. We believe that the algorithmic and game theoretic insights gained from our approach will help better understand the tradeoff between approximability and the incentive compatibility.

1 Introduction

In an *algorithmic mechanism design* problem, we face an optimization problem where the necessary inputs are private valuations held by self-interested agents. The high-level goal of *truthful* mechanisms is to reveal these valuations via the bids of the agents and to optimize the objective simultaneously. In this paper, we will focus on the objective of social welfare maximization.

It is well known that the *VCG* mechanism ([24, 7, 13]) which maximizes the social welfare exactly is truthful. As usual in computer science, computational tractability is a necessary requirement. However, VCG is not computationally efficient in general. And unfortunately, the simple combination of approximation algorithms and VCG usually fails to preserve truthfulness. This raises the important open question (see [21]) of whether the design of truthful mechanisms is fundamentally harder than the design of approximation algorithms for social welfare maximization.

⋆ This work is supported in part by the National Science Foundation under Grant No. 0716172 and ONR MURI Grant N000140710907.

Recently, several positive results indicated that one can always convert an approximation algorithm to a truthful mechanism with the same approximation factor in the *Bayesian* setting where the distributions of the agents are public knowledge (see [15, 3, 14]). However, not much is known in the *prior-free* setting where no distribution is known.

In this paper, by studying the problem of *single-parameter combinatorial auctions*, we show the first black-box reduction that converts any approximation algorithm to a universal truthful mechanism with the same approximation factor in the prior-free setting.

In the single-parameter combinatorial auction problem, we are given a set $\mathcal{J}$ of m items and a *public* valuation function $f : 2^{\mathcal{J}} \to \mathbf{R}$. Assume that f is given via an oracle which takes a set S as input and returns $f(S)$. In addition, we have n agents each of whom has a *private* multiplier v_i^* such that the item set S provides $v_i^* f(S)$ amount of utility to agent i. The goal is to design a truthful mechanism which maximizes $\sum_i v_i f(S_i)$, where $S_1 \cdots S_n$ is a partition of $\mathcal{J}$.

This problem has its motivation in the TV ad auctions where the items are time slots and each agent is an advertiser whose private multiplier is her *value-per-viewer*. In [12], the authors provided a logarithmic approximate truthful mechanism for this problem under the assumption that f is submodular. However, the optimal approximation algorithm for the underlying social welfare maximization has a ratio of $1 - 1/e$ given by Vondrak ([25]). By our result, applying Vondrak's algorithm as a black-box, we immediately obtain a truthful mechanism with the optimal constant approximation ratio.

Main result. In fact, our black-box reduction not only works for this particular problem but for a broad class of *symmetric single parameter* problems. Formally, a mechanism design problem (with n agents) is *single-parameter* if each feasible allocation is represented as an n-dimensional real vector $\boldsymbol{x}$, and each agent i has a private value v_i such that her valuation of allocation $\boldsymbol{x}$ is given by $v_i x_i$. We further define that a problem is *symmetric* if the set of feasible allocations is closed under permutations: if $\boldsymbol{x}$ is feasible, so is $\pi \circ \boldsymbol{x}$ for any permutation π. Here $\pi \circ \boldsymbol{x}$ is defined as the vector $(x_{\pi(1)}, ..., x_{\pi(n)})$.

Theorem 1. *For a symmetric single-parameter mechanism design problem Π, suppose we are given an α-approximate ($\alpha > 1$) algorithm $\mathcal{A}$ as a black-box, then for any constant $\epsilon > 0$, we can obtain a polynomial time truthful mechanism with approximation factor $\alpha(1 + \epsilon)$.*

Many interesting mechanism design problems such as position auctions in sponsored search are in the class of symmetric single-parameter problems. In particular, it contains the problem of single-parameter combinatorial auctions that we are interested in.

Corollary 1. *For the single-parameter submodular combinatorial auction problem, there is an optimal 1-1/e approximate truthful mechanism.*

Our construction is based on the technique of *maximum-in-range*. Here, a maximum-in-range mechanism outputs the allocation maximizing the social welfare over a fixed range of allocations. Using the algorithm $\mathcal{A}$ as a black-box, we construct a range $\mathcal{R}$ such that social welfare maximization over $\mathcal{R}$ is efficient. And we will prove that the approximation factor obtained is essentially α.

In our reduction, we make no assumption on the black-box algorithm $\mathcal{A}$. In addition, while the black-box algorithm may be randomized, our reduction does not introduce any further randomization. If the algorithm is deterministic, then our mechanism is deterministically truthful.

Extensions: positive and negative results. A natural extension of our result is to consider the general (possibly asymmetric) single-parameter mechanism design problems. By a novel relation between mechanism design and constraint satisfaction problems, we derive a significant lower bound of the approximability of maximum-in-range mechanisms for general single-parameter problems, which to some extent suggests the difficulty in designing factor-preserving black-box reductions.

However, we are able to generalize our algorithmic technique to some special *multi-parameter* settings. We study the *constant-dimension* and *symmetric* mechanism design problem. We generalize our construction in the symmetric single-parameter case to this problem and obtain a black-box reduction that converts any algorithm into a truthful and quasi-poly-time mechanism with essentially the same approximation guarantee. Alternatively, we can obtain a black-box reduction that converts any algorithm into a truthful and polynomial time mechanism with logarithmic degradation in the approximation factor.

Related work. There has been a significant amount of work related to black-box reductions in mechanism design. In the single-parameter setting, the first black-box reduction was given by Briest et al. [4]. The authors studied the single-parameter binary optimization problem and they showed that any algorithm which is an FPTAS can be converted to a truthful mechanism that is also an FPTAS. Secondly, Babaioff et al. [1] studied the single-value combinatorial auction problem and they constructed a black-box reduction that converts an algorithm to a truthful mechanism with the approximation factor degraded by a logarithmic factor. Finally, the recent work by Goel et al. [12] provided a black-box reduction with a super constant degrade in approximation factor for partially public combinatorial auction.

For multi-parameter problems, there is no factor-preserving black-box reduction in general (e.g. [22]). This motivates the study of *truthfulness in expectation*, which is a weaker notion of incentive compatibility. Here, a randomized mechanism is truthful in expectation, if truth telling maximizes an agent's expected payoff. The initial effort in black-box reduction for multi-parameter problems is due to Lavi and Swamy [18], they showed a method to convert a certain type of algorithms called integrality-gap-verifiers to truthful in expectation mechanisms with the same approximation factors. Recently, Dughmi and Roughgarden [9] studied the class of packing problems. Via an elegant black-box reduction and smooth analysis, they showed that if a packing problem admits an FPTAS, then it admits a truthful in expectation mechanism that is an FPTAS as well. Balcan et al.[2] considered black-box reductions from the revenue maximization aspect. By the technique of sample complexity in machine learning, they gave revenue-preserving reductions from truthful mechanism design to the algorithmic pricing problems. At last, Dughmi et al. [10] introduce a method to convert convex rounding scheme into truthful in expectation mechanism and achieve an optimal $(1-\frac{1}{e})$-approximation for the combinatorial auction problem when the valuations are a special type of submodular functions.

The previous discussion is about *prior-free* mechanism design. Another important area in algorithmic game theory is the *Bayesian* mechanism design where each agent's valuation is drawn from some publicly known prior distribution. Hartline and Lucier [15] studied this problem in the single-parameter setting. They constructed a clever black-box reduction that converts any non-monotone algorithm into a monotone one without compromising its social welfare. Following this work, Bei and Huang [3] and Hartline et al. [14] independently showed such black-box reductions in the multi-parameter setting as well.

2 Preliminaries

In this section, we will outline the basic concepts in mechanism design relevant to our paper.

Truthfulness. Let $\mathcal{X}$ be the set of all feasible allocations, and $v_i(\boldsymbol{x})$ be the private valuation of agent i if allocation $\boldsymbol{x}\in\mathcal{X}$ is picked. A typical goal of a mechanism is to reveal agents' private valuation functions via their bids and optimize the obtained social welfare simultaneously. Formally, suppose we are given n agents and let $\boldsymbol{v}=(v_1,...,v_n)$ be the valuation functions reported by the agents. Based on this, a (deterministic) mechanism M will specify an allocation $\boldsymbol{x}(\boldsymbol{v})\in\mathcal{X}$ and a payment $\boldsymbol{p}(\boldsymbol{v})$. We say M is *deterministically truthful*(or truthful), if the following conditions hold: for any $i,\boldsymbol{v}_{-i}$ and any v_i,v_i', we have $v_i(\boldsymbol{x}(v_i,\boldsymbol{v}_{-i}))-p_i(v_i,\boldsymbol{v}_{-i})\geq v_i(\boldsymbol{x}(v_i',\boldsymbol{v}_{-i}))-p_i(v_i',\boldsymbol{v}_{-i})$.

When a mechanism is randomized, there are two notions of truthfulness: (1) *Universal truthfulness*: A universally truthful mechanism is a probability distribution over deterministically truthful mechanisms; (2) *Truthfulness in expectation*: A mechanism is truthful in expectation if an agent maximizes her *expected utility* by being truthful. Here, an agent's utility is defined as her valuation minus payment. It is easy to see that every deterministically truthful mechanism is universally truthful and every universally truthful mechanism is truthful in expectation.

Single-parameter mechanism design. In a *single-parameter* mechanism design problem, each allocation is represented as an n-dimensional real vector $\boldsymbol{x}$ (where n is the number of agents), and each agent i has a private value v_i such that her valuation of allocation $\boldsymbol{x}$ is given by $v_i x_i$. It is known [20] that for a single-parameter problem, a mechanism is truthful if and only if (1) the allocation rule is *monotone*: suppose $v_i\leq v_i'$, then $x_i(v_i,\boldsymbol{v}_{-i})\leq x_i(v_i',\boldsymbol{v}_{-i})$; (2) each agent i's payment is determined by $p_i(\boldsymbol{v})=v_i x_i(v_i,\boldsymbol{v}_{-i})-\int_0^{v_i}x_i(t,\boldsymbol{v}_{-i})dt$.

Maximum-in-range mechanisms. The *maximum-in-range* technique is a general approach in the field of mechanism design. It works as follows: The mechanism fixes a range $\mathcal{R}$ of allocations *without* any knowledge of the agents' valuations. Given any $\boldsymbol{v}$, let $\boldsymbol{x}^*$ =argmax$_{\boldsymbol{x}\in\mathcal{R}} \sum_j v_j(\boldsymbol{x})$ and $\boldsymbol{x}^*_{-i}$ =argmax$_{\boldsymbol{x}\in\mathcal{R}} \sum_{j\neq i} v_j(\boldsymbol{x})$ respectively. Now define payment p_i of agent i to be $\sum_{j\neq i} v_j(x^*_{-i}) - \sum_{j\neq i} v_j(x^*)$. It is now not difficult to see that with this payment function, it is in best interest of every agent to report their true valuations, irrespective of what others report. The major challenge in designing maximum-in-range mechanism is to balance between the size of the range and the approximation factor obtained. A larger range can obtain better approximation but yield greater computational complexity.

3 Symmetric Single-Parameter Mechanism Design

Recall that a single-parameter mechanism design problem is symmetric if the allocation space $\mathcal{X}$ is closed under permutations: if $\boldsymbol{x} \in \mathcal{X}$, then $\pi \circ \boldsymbol{x} = (x_{\pi(1)}, ..., x_{\pi(n)}) \in \mathcal{X}$ for any permutation π. In this section, we will prove Theorem 1: For a symmetric single-parameter problem Π, given any constant $\epsilon > 0$ and any α-approximate algorithm $\mathcal{A}$ as a black-box, we design a polynomial time truthful mechanism with approximation factor $(1+\epsilon)\alpha$.

Our construction is based on the *maximum-in-range* technique. Given an algorithm $\mathcal{A}$, we define a range $\mathcal{R}$ by applying $\mathcal{A}$ as a black-box on a carefully chosen collection of *typical bid vectors*. Our mechanism is then maximum-in-range over $\mathcal{R}$. We will show: (1) To maximize social welfare over $\mathcal{R}$ for a given bid vector, we only need to examine polynomially many allocations in $\mathcal{R}$, hence our mechanism is efficient; (2) Every bid vector can be mapped to a typical bid with approximately the same social welfare, hence our mechanism performs almost as well as the algorithm $\mathcal{A}$. This proves the approximation factor.

Now we describe our range construction in detail for a given symmetric single-parameter problem Π, black-box algorithm $\mathcal{A}$ and constant $\epsilon > 0$.

3.1 Construction of the range

Let $V = \mathbf{R}^n_+$ be the collection of all possible bid vectors. Next we will provide a three-step procedure choosing a subset $T \subseteq V$ as our collection of *typical bids*.

The first step is *normalization*: By properly reordering the agents and scaling their bids, we only consider the set T_0 of bids where $\boldsymbol{v} \in T_0$ if and only if $1 = v_1 \geq ... \geq v_n$; The second step is *discretization*. In this step, our goal is to obtain a finite set of bid vectors that approximately represent the whole valuation space V. To do this, given any vector $\boldsymbol{v} \in T_0$, we first apply the operation of *tail cutting*: We choose a small value u (e.g. $1/n^M$ for some constant M) and round all the entries smaller than u to 0; then, we discretize the interval $[u, 1]$ by considering $Q = \{\eta^k : k \geq 0\} \cap [u, 1]$ where $\eta < 1$ is a fixed constant. We will round down each of the remaining entries of $\boldsymbol{v}$ after the tail cutting to the closest value in Q. If we do the above for each $\boldsymbol{v} \in T_0$, we obtain a finite set of vectors T_1; The final step is *equalization*. We fix a constant $\beta > 1$ and partition $[n]$ into $\log_\beta n$ groups. For each vector in T_1, we equalize its entries within each group by setting them to be the value of the largest entry in the group. We then obtain the set of vectors T, and each vector in T is called a *typical bid*.

1: **Normalization.** Let $T_0 = \{\boldsymbol{v} : 1 = v_1 \geq ... \geq v_n\}$;
2: **Discretizing.** Let $Q = \{\eta^k : 0 \leq k \leq \lceil \log_{1/\eta}(n^M) \rceil\}$ where $M \geq \log_2 \frac{8}{\epsilon}$ is a constant. For any real value z, define $\lfloor z \rfloor_\eta = \eta^{\lceil \log_\eta z \rceil} \in Q$. Then we define a function $D : T_0 \mapsto T_0$ as follows: for each $\boldsymbol{v} \in T_0$ and for each i, define
$$D(\boldsymbol{v})_i = \begin{cases} \lfloor v_i \rfloor_\eta & v_i \geq u = \frac{1}{n^M} \\ 0 & \text{otherwise} \end{cases}$$
Let $T_1 = D(T_0)$;
3: **Equalization.** Let $n_k = \lfloor \beta^k \rfloor$ where $\beta > 1$ is a fixed constant and $0 \leq k \leq \lfloor \log_\beta n \rfloor$. Define a function $E : T_1 \to T_1$ as follows: for each $\boldsymbol{v} \in T_1$ and $1 \leq i \leq n$, $E(\boldsymbol{v})_i$ is set to be v_{n_k} when $n_k \leq i < n_{k+1}$. At last, let $T = E(T_1)$.

Now we provide the detailed description. In the following, we fix constants $\beta > 1$ and $\eta < 1$ such that $\frac{\beta}{\eta} = 1 + \epsilon/2$. For a bid vector $\boldsymbol{v}$, let $\boldsymbol{x}^{\mathcal{A}}(\boldsymbol{v})$ be the allocation obtained by applying algorithm $\mathcal{A}$ on $\boldsymbol{v}$. Since the allocation space is closed under permutations, we may assume $\boldsymbol{x}^{\mathcal{A}}(\boldsymbol{v})_1 \geq \boldsymbol{x}^{\mathcal{A}}(\boldsymbol{v})_2 \geq ... \geq \boldsymbol{x}^{\mathcal{A}}(\boldsymbol{v})_n$. At last, let $\mathcal{R}_0 = \{\boldsymbol{x}^{\mathcal{A}}(\boldsymbol{v}) : \boldsymbol{v} \in T\}$ and we finally define our range as $\mathcal{R} = \{\pi \circ \boldsymbol{x} : \boldsymbol{x} \in \mathcal{R}_0, \pi \in \Pi_n\}$ where Π_n consists of all permutations over n elements.

Now we analyze the performance of our mechanism. Since the mechanism is maximum-in-range, it is truthful. We will show that it has polynomial running time and an approximation factor of $\alpha(1+\epsilon)$.

Running time. We show that the social welfare maximization over $\mathcal{R}$ is solvable within polynomial time, hence our maximum-in-range mechanism is efficient. We will first show $|\mathcal{R}_0|$ is polynomial in n.

Lemma 1. $|\mathcal{R}_0| \leq |T| \leq n^{1/\log_2 \beta + M/\log_2(1/\eta)}$.

Proof. The first inequality follows from the definition of $\mathcal{R}_0$. Now we prove the second one. Observe that for each vector $\boldsymbol{v}$ in T_1, $E(\boldsymbol{v})$ is uniquely determined by the values $\{v_{n_k} : 0 \leq k \leq \lfloor \log_\beta n \rfloor\} \subseteq Q \bigcup \{0\}$. Moreover, we have that $v_{n_{k-1}} \geq v_{n_k}$ for all k. Therefore, let H be the class of non-increasing functions from $\{0, 1, ..., \lfloor \log_\beta n \rfloor\}$ to $Q \bigcup \{0\}$, thus $|T| \leq |H|$. Since $|Q| = \lceil \log_{1/\eta}(n^M) \rceil$, It is not difficult to see, $|H| \leq \binom{\lfloor \log_\beta n \rfloor + \lceil \log_{1/\eta}(n^M) \rceil}{\lceil \log_{1/\eta}(n^M) \rceil} \leq 2^{\lfloor \log_\beta n \rfloor + \lceil \log_{1/\eta}(n^M) \rceil} \leq n^{1/\log_2 \beta + M/\log_2(1/\eta)}$.

Now we are ready to prove the running time guarantee. Let $\mathsf{opt}_{\mathcal{R}}(\boldsymbol{v})$ be the allocation maximizes the social welfare over $\mathcal{R}$ for the given bid vector $\boldsymbol{v}$. Let σ be the permutation such that $v_{\sigma(1)} \geq ... \geq v_{\sigma(n)}$. Obviously, for each $\boldsymbol{x} \in \mathcal{R}_0$, we have $\boldsymbol{v} \cdot (\sigma^{-1} \circ \boldsymbol{x}) \geq \boldsymbol{v} \cdot (\pi \circ \boldsymbol{x})$ for all permutation π. Therefore, $\mathsf{opt}_{\mathcal{R}}(\boldsymbol{v}) \in \{\sigma^{-1} \circ \boldsymbol{x} :$

$\boldsymbol{x} \in \mathcal{R}_0\}$. By Lemma 1, $|\{\sigma^{-1} \circ \boldsymbol{x} : \boldsymbol{x} \in \mathcal{R}_0\}| = |\mathcal{R}_0| \leq n^{1/\log_2 \beta + M/\log_2 \eta}$, this implies that $\mathsf{opt}_{\mathcal{R}}(\boldsymbol{v})$ can be found in polynomial time.

Approximation factor. We show that the approximation factor of our mechanism is $\alpha(1+\epsilon)$. Given any bid vector $\boldsymbol{v}$, by reordering and scaling properly, we may assume $\boldsymbol{v} \in T_0$, then we consider the typical bid $E(D(\boldsymbol{v}))$. We show that for any sorted allocation $\boldsymbol{x}$, the social welfare $\boldsymbol{v} \cdot \boldsymbol{x}$ is $(1+\epsilon)$-approximated by $E(D(\boldsymbol{v})) \cdot \boldsymbol{x}$, hence an α-approximate solution for social welfare maximization with respect to $E(D(\boldsymbol{v}))$ is an $\alpha(1+\epsilon)$-approximate solution for $\boldsymbol{v}$. This proves the desired approximation guarantee.

Now we provide the detail. We first show that by considering $D(\boldsymbol{v})$ instead of $\boldsymbol{v} \in T_0$, the social welfare is rounded down by at most a factor of $\eta(1-\epsilon/4)$.

Lemma 2. *For any $\boldsymbol{v} \in T_0$ and any allocation $\boldsymbol{x}$ s.t. $x_1 \geq ... \geq x_n$, we have $D(\boldsymbol{v}) \cdot \boldsymbol{x} \leq \boldsymbol{v} \cdot \boldsymbol{x} \leq \frac{1}{\eta(1-\epsilon/4)} D(\boldsymbol{v}) \cdot \boldsymbol{x}$.*

Proof. The first inequality holds by definition. Now we prove the second one. We first show the following lemma which says that the social welfare affected by "tail cutting" is bounded by a fraction of $\epsilon/4$. The proof of the lemma is deferred to full version due to space reasons.

Lemma 3. $\sum_{i:v_i \geq 1/n^M} v_i x_i \geq (1-\epsilon/4) \sum_{i=1}^n v_i x_i$.

Let $\boldsymbol{v'} = D(\boldsymbol{v})$, it is easy to see: $\boldsymbol{v'} \cdot \boldsymbol{x} = \sum_{i:v_i \geq 1/n^M} v'_i x_i \geq \eta \sum_{i:v_i \geq 1/n^M} v_i x_i \geq \eta(1-\frac{\epsilon}{4}) \boldsymbol{v} \cdot \boldsymbol{x}$.

Secondly, we show that the social welfare increases by at most a factor of β by considering $E(\boldsymbol{v})$ instead of $\boldsymbol{v}$ for any $\boldsymbol{v} \in T_1$.

Lemma 4. *For any $\boldsymbol{v} \in T_1$ and any allocation $\boldsymbol{x}$ s.t. $x_1 \geq \cdots \geq x_n$, we have $\boldsymbol{v} \cdot \boldsymbol{x} \leq E(\boldsymbol{v}) \cdot \boldsymbol{x} \leq \beta \boldsymbol{v} \cdot \boldsymbol{x}$.*

Proof. The first inequality is implied by the definition of E. We will prove the second one. Let $\boldsymbol{v'} = E(\boldsymbol{v})$ and $L = \lceil \log_{1/\eta} n \rceil$. Since $\boldsymbol{v}, \boldsymbol{v'} \in T_1$, we have that $v_i, v'_i \in Q$ for each i. Thus, if we let $l_i = \log_\eta v_i$ and $l'_i = \log_\eta v'_i$ respectively for each i, then l_i's and l'_i's are non-decreasing sequences. By our construction, it is easy to see: (1) for all $1 \leq i \leq n$, $l'_i \leq l_i$; (2) for all $l \in [0, L]$, $|\{i : l'_i \leq l\}| \leq \beta\, |\{i : l_i \leq l\}|$. Since $x_1 \geq ... \geq x_n$, we have the following:

Lemma 5. *For any $l \in [0, L]$ and $1 \leq i \leq n$, $\sum_{i:l'_i \leq l} x_i \leq \beta \sum_{i:l_i \leq l} x_i$.*

Observe that if we define $W_l = \eta^l$ for $0 \leq l \leq L$ and $W_{L+1} = 0$, then $\sum_{i=1}^n x_i v_i = \sum_{i=1}^n x_i W_{l_i} = \sum_{i=1}^n x_i \sum_{l=l_i}^{L} (W_l - W_{l+1}) = \sum_{l=0}^{L} (W_l - W_{l+1}) \sum_{i:l_i \leq l} x_i$.

Similarly, we have $\sum_{i=1}^n x_i v'_i = \sum_{l=0}^{L} (W_l - W_{l+1}) \sum_{i:l'_i \leq l} x_i$. By Lemma 5, we have $\boldsymbol{v'} \cdot \boldsymbol{x} \leq \beta \boldsymbol{v} \cdot \boldsymbol{x}$.

By Lemma 2 and Lemma 4, we have the following:

Corollary 2. *For any* $\boldsymbol{v} \in T_0$ *and any allocation* $\boldsymbol{x}$ *such that* $x_1 \geq \cdots \geq x_n$*, we have:* $\eta(1-\epsilon/4)\boldsymbol{v}\cdot\boldsymbol{x} \leq E(D(\boldsymbol{v}))\cdot\boldsymbol{x} \leq \beta\boldsymbol{v}\cdot\boldsymbol{x}$*.*

Now we prove the approximation guarantee of our mechanism. Given any bid vector $\boldsymbol{v}$, without loss of generality, we may assume $\boldsymbol{v} \in T_0$. Let $\boldsymbol{z}^*$ be the optimal solution of social welfare maximization for $\boldsymbol{v}$ and $\boldsymbol{x}^*$ be the solution output by our mechanism. In addition, let $\boldsymbol{y}^*$ be the optimal solution for the typical bid $E(D(\boldsymbol{v}))$. Then, by Corollary 2, we have $\boldsymbol{v}\cdot\boldsymbol{x}^* \geq \frac{1}{\beta}E(D(\boldsymbol{v}))\cdot\boldsymbol{x}^*$. Since our algorithm is maximum-in-range, allocation $\boldsymbol{x}^*$ is at least as good as the allocation by algorithm $\mathcal{A}$ with respect to typical bid vector $E(D(\boldsymbol{v}))$. Hence, we have $E(D(\boldsymbol{v}))\cdot\boldsymbol{x}^* \geq \frac{1}{\alpha}E(D(\boldsymbol{v}))\cdot\boldsymbol{y}^*$. Further, by optimality of $\boldsymbol{y}^*$ and Corollary 2, we have $E(D(\boldsymbol{v}))\cdot\boldsymbol{y}^* \geq E(D(\boldsymbol{v}))\cdot\boldsymbol{z}^* \geq \eta\left(1-\frac{\epsilon}{4}\right)\boldsymbol{v}\cdot\boldsymbol{z}^*$.

In all, we have $\boldsymbol{v}\cdot\boldsymbol{x}^* \geq \frac{\eta}{\alpha\beta}\left(1-\frac{\epsilon}{4}\right)\boldsymbol{v}\cdot\boldsymbol{z}^*$. Since we choose β and η such that $\beta/\eta = 1+\epsilon/2$, we have $\boldsymbol{v}\cdot\boldsymbol{z}^* \leq \alpha(1+\epsilon/2)\boldsymbol{v}\cdot\boldsymbol{x}^*/(1-\epsilon/4) \leq \alpha(1+\epsilon)\boldsymbol{v}\cdot\boldsymbol{x}^*$. This completes our analysis.

4 General SPMD : Limitation of MIR Mechanisms

In the previous section, we study the black-box reductions in the symmetric single-parameter setting. In this section, we give some negative results for factor-preserving black-box reduction in general single-parameter mechanism design (SPMD). We derive a significant approximability gap between maximum-in-range mechanisms and approximation algorithms in the most general single-parameter setting. To do this, we establish a novel relation between SPMD and maximum constraint satisfaction problems (MaxCSP), and show that the approximation ratio of a MIR mechanism for some SPMD problem can be arbitrarily worse than that of the best approximation algorithm for the "corresponding" MaxCSP problem.

Specifically, for every MaxCSP problem Γ that is NP-Hard, we set up a corresponding SPMD problem Γ', mapping (which can be done in polynomial time) each instance $\mathcal{I} \in \Gamma$ to a profile of agent valuation $v_{\mathcal{I}}$, while $\mathsf{opt}_{\Gamma}(\mathcal{I}) = \mathsf{opt}_{\Gamma'}(v_{\mathcal{I}})$. For every (efficient) MIR mechanism, we show that unless NP $\subseteq$ P/poly, the approximation guarantee of the mechanism (on Γ') can be no better than that of a random assignment for the corresponding MaxCSP problem Γ, and therefore is arbitrarily worse than the guarantee of the best approximation algorithms, for some carefully chosen Γ.

For the sake of exposition, we choose Γ to be Max k-AllEqual (which can be any MaxCSP problem, although the gap between performance of MIR mechanisms and that of approximation algorithms might be different). The Max k-AllEqual problem is defined as follows.

Definition 1 (Max k-AllEqual)**.** *Given a set* C *of clauses of the form* $l_1 \equiv l_2 \equiv \cdots \equiv l_k$ *(*k *constant), where each literal* l_i *is either a Boolean variable* x_j *or its negation* $\bar{x_j}$*. The goal is to find an assignment to the variables* x_i *so as to maximize the number of satisfied clauses.*

The MAX k-ALLEQUAL problem is NP-Hard. In fact, it is NP-Hard to approximate MAX k-ALLEQUAL problem within a factor of $2^{c\sqrt{k}}/2^k$ for some constant $c > 0$, according to [23, 16, 11]. On the algorithmic side, there is an $\Omega(k/2^k)$-approximation algorithm for MAX k-ALLEQUAL shown in [6]. The algorithm based on SDP-relaxation and randomized rounding, but it can be efficiently derandomized by embedding the SDP solution to a low dimensional space, via derandomized Johnson-Lindenstrauss transformation [17, 8, 19].

CSP-based hard instance for MIR mechanisms. We describe the corresponding SPMD problem for MAX k-ALLEQUAL as follows. For a MAX k-ALLEQUAL problem with n variables, we set up $M = (2n)^k$ agents in $M_{\text{MAX } k\text{-ALLEQUAL}}$, each corresponding to a clause $c : l_1 \equiv l_2 \equiv \cdots \equiv l_k$. Recall that in a SPMD problem, each agent's valuation is a single real number that specifies its utility for being served. We conclude the description of our hard instance by defining the winner set, i.e. the set of feasible subset of agents being served. For any Boolean assignment $x : [n] \to \{\mathsf{true}, \mathsf{false}\}$, let $C(x) \subseteq [M]$ be the set of clauses that are satisfied by x. We define the set of feasible allocation functions $Y \subseteq \{\boldsymbol{y} : [M] \to \{0,1\}\}$ to be $Y = \{\mathbf{1}_{C(x)} | x : [n] \to \{\mathsf{true}, \mathsf{false}\}\}$.

Given a MAX k-ALLEQUAL instance $\mathcal{I}$ with set C of clauses, define the valuation function for the agents, $\boldsymbol{v}_{\mathcal{I}} = \boldsymbol{v} : [M] \to \{0,1\}$, to be the indicator function of C, i.e. $\boldsymbol{v}(c) = \mathbf{1}_C(c) = \begin{cases} 1 & c \in C \\ 0 & \text{otherwise} \end{cases}$.

Note that here we assume that every clause appears at most once in C. But the hard instance can be easily generalized to weighted case, by letting $\boldsymbol{v}(c)$ be the weight of clause c.

Analysis. It's easy to check the following fact.

Fact 1 $\mathsf{opt}(\mathcal{I}) = \max_{x:[n]\to\{\mathsf{true},\mathsf{false}\}} \left\{\boldsymbol{v} \cdot \mathbf{1}_{C(x)}\right\} = \max_{y \in Y} \{\boldsymbol{v} \cdot \boldsymbol{y}\} = \mathsf{opt}(v_{\mathcal{I}})$.

Now, we prove that there is a significant gap between the approximation guarantee of any MIR mechanism and that of the approximation algorithms. The following theorem shows that MIR mechanism performs $\Omega(k)$ times worse than the approximation algorithm for the corresponding algorithmic task, for any constant $k > 0$.

Theorem 2. *Assuming* NP $\not\subseteq$ P/poly*, there is no polynomial time MIR mechanism with approximation ratio better than* $2(1+\epsilon)/2^k$*, for any constant* $\epsilon > 0$.

The detailed proof of Theorem 2 is deferred to full version, due to space reasons. At high level, the proof of Theorem 2 consists of two steps. Assuming there is an MIR mechanism with range $\mathcal{R}$ achieving $2(1+\epsilon)/2^k$ approximation guarantee, we firstly show that $\mathcal{R}$ needs to be exponentially large. Then we use Sauer-Shelar Lemma to argue that when $\mathcal{R}$ is sufficiently large, it must cover all possible assignments for a constant fraction of the n variables in MAX k-ALLEQUAL, and we can use this mechanism exactly solve MAX k-ALLEQUAL problem on this fraction of variables, which is NP-Hard.

The above technique was first introduced in [5] to show the inapproximability result in combinatorial auctions. However, their construction relies on the complicated private structures of agents' valuations, hence does not apply in our problem. Our approach can be viewed as a novel generalization of their technique in single-parameter mechanism design. To our knowledge, this is the first example of a lower bound on MIR mechanisms for problems that are linear and succinct.

5 Symmetric Multi-Parameter Mechanism Design

As a natural generalization of single-parameter mechnism design, we consider the multi-parameter problems in this section. Due to space reasons, proofs of the theorems in this section are deferred to full version of this paper.

As before, a problem is *symmetric* if $\boldsymbol{S} = (S_1, ..., S_n)$ is a feasible allocation implies that $\pi \circ \boldsymbol{S} = (S_{\pi(1)}, ..., S_{\pi(n)})$ is also a feasible allocation for any permutation π. Moreover, a mechanism design problem is Δ-dimension if the valuation of each agent i can be naturally represented by a Δ-dimension vector $\boldsymbol{u}_i = (u_{i1}, \ldots, u_{i\Delta}) \in \mathbb{R}_+^\Delta$. We let $v(S, \boldsymbol{u})$ denote an agent's value of an allocation S when its valuation function is given by a Δ-dimension vector $\boldsymbol{u}$. We will assume that the problem satisfies the following properties:

- **Monotonicity.** For any $1 \le i \le n$, $\boldsymbol{S}$, $\boldsymbol{u}_i$ and $\boldsymbol{u}'_i$ such that $u_{ij} \ge u'_{ij}$ for any $1 \le j \le \Delta$, we have $v(S_i, \boldsymbol{u}_i) \ge v(S_i, \boldsymbol{u}'_i)$.
- **Sub-linear influence.** For any $1 \le k \le \Delta$, $\beta > 1$, $\boldsymbol{u}$ and $\boldsymbol{u}'$ such that for any $1 \le i \le n$, $u_{ij} = u'_{ij}$ for any $j \ne k$, and $u_{ik} \le \beta u'_{ik}$, we have $\mathsf{opt}(\boldsymbol{u}) \le \beta \mathsf{opt}(\boldsymbol{u}')$.
- **Negligible tail.** For any $\delta > 0$, let $\boldsymbol{u}_i^\delta$ be the tail-truncated values: $u_{ij}^\delta = u_{ij}$ if $u_{ij} \ge \delta \max_{s,t} u_{st}$ and $u_{ij}^\delta = 0$ otherwise. For any constant $\epsilon > 0$, there is a polynomially small $\delta > 0$, so that for any allocation $\boldsymbol{S}$ and any values $\boldsymbol{u}_i$'s, we have $(1+\epsilon)\sum_{i=1}^n v(S_i, \boldsymbol{u}_i^\delta) \ge \sum_{i=1}^n v(S_i, \boldsymbol{u}_i)$.

These assumptions are without loss of generality in many mechanism design problems. For example, consider the following:

- **Multi-item auction.** In multi-item auctions, we consider n agents and m different types of items, each of which has a finite supply. Each agent i has a private m-dimension vector of values $\boldsymbol{u}_i = (u_{i1}, \ldots, u_{im})$. Agent i's value of a bundle S with x_j items of type j, $1 \le j \le m$, is $v(S, \boldsymbol{u}_i) = \sum_{j=1}^m x_j u_{ij}$. This is a m-dimensional problem that satisfies our assumptions.
- **Combinatorial auction.** In combinatorial auctions, we consider n agents and m different items. Each agent i has a private 2^m-dimension vector $\boldsymbol{u}_i$ so that for each subset of items $S \in 2^{[m]}$, agent i's value of bundle S is $v(S, \boldsymbol{u}_i) = u_{iS}$. This is a 2^m-dimensional problem that satisfies our assumptions.

Via techniques similar to those in Section 3, we can show the following theorem. The proofs are deferred to the full version of this paper.

Theorem 3. *For any Δ-dimension symmetric mechanism design problem Π where Δ is a constant, suppose $\mathcal{A}$ is an α-approximate algorithm, then for any constant $\epsilon > 0$, we can get an truthful and $(1+\epsilon)\alpha$-approximate mechanism that runs in quasi-polynomial time given $\mathcal{A}$ as a black-box.*

Alternatively, we can alleviate the running time by having greater degrade in the approximation factor.

Theorem 4. *For any Δ-dimension symmetric mechanism design problem Π where Δ is a constant, suppose $\mathcal{A}$ is an α-approximate algorithm, then for any constant $\epsilon > 0$, we can get a truthful and $\alpha\,\mathrm{polylog}$-approximate mechanism that runs in polynomial time given $\mathcal{A}$ as a black-box.*

Acknowledgments

The authors would like to thank Nina Balcan, Yang Cai, Deeparnab Chakrabarty, Florin Constantin, Gagan Goel, Venkatesan Guruswami, Vijay Vazirani and Jiajin Yu for many useful comments and helpful discussions.

References

1. Moshe Babaioff, Ron Lavi, and Elan Pavlov. Single-value combinatorial auctions and implementation in undominated strategies. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm (SODA)*, pages 1054–1063, New York, NY, USA, 2006. ACM.
2. Maria-Florina Balcan, Avrim Blum, Jason D. Hartline, and Yishay Mansour. Mechanism design via machine learning. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 605–614, 2005.
3. Xiaohui Bei and Zhiyi Huang. Bayesian incentive compatibility via fractional assignments. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), to appear*, 2011.
4. Patrick Briest, Piotr Krysta, and Berthold Vöcking. Approximation techniques for utilitarian mechanism design. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, pages 39–48, New York, NY, USA, 2005. ACM.
5. Dave Buchfuhrer, Shaddin Dughmi, Hu Fu, Robert Kleinberg, Elchanan Mossel, Christos Papadimitriou, Michael Schapira, Yaron Singer, and Chris Umans. Inapproximability for vcg-based combinatorial auctions. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2010.
6. Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Near-optimal algorithms for maximum constraint satisfaction problems. *ACM Trans. Algorithms*, 5:32:1–32:14, July 2009.
7. Edward H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1), September 1971.
8. Ilias Diakonikolas, Daniel M. Kane, and Jelani Nelson. Bounded independence fools degree-2 threshold functions. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2010.

9. Shaddin Dughmi and Tim Roughgarden. Black-box randomized reductions in algorithmic mechanism design. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2010.
10. Shaddin Dughmi, Tim Roughgarden, and Qiqi Yan. From convex optimization to randomized mechanisms: Toward optimal combinatotial auctions for submodular bidders. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC)*, 2011.
11. Lars Engebretsen and Jonas Holmerin. More efficient queries in PCPs for NP and improved approximation hardness of maximum CSP. *Random Structures & Algorithms*, 33(4):497–514, 2008.
12. Gagan Goel, Chinmay Karande, and Lei Wang. Single-parameter combinatorial auctions with partially public valuations. In *Proceedings of the 3rd International Symposium on Algorithmic Game Theory (SAGT)*, 2010.
13. Theodore Groves. Incentives in teams. *Econometrica*, 41(4):617–631, July 1973.
14. Jason Hartline, Robert Kleinberg, and Azarakhsh Malekian. Bayesian incentive compatibility via matchings. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), to appear*, 2011.
15. Jason D. Hartline and Brendan Lucier. Bayesian algorithmic mechanism design. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 301–310, New York, NY, USA, 2010. ACM.
16. Johan Håstad and Avi Wigderson. Simple analysis of graph tests for linearity and pcp. *Random Structures and Algorithms*, 22(2):139–160, 2003.
17. Daniel M. Kane and Jelani Nelson. A derandomized sparse Johnson-Lindenstrauss transform. *Arxiv preprint arXiv:1006.3585*, 2010.
18. Ron Lavi and Chaitanya Swamy. Truthful and near-optimal mechanism design via linear programming. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 595–604, Washington, DC, USA, 2005. IEEE Computer Society.
19. Raghu Meka and David Zuckerman. Pseudorandom generators for polynomial threshold functions. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 427–436, 2010.
20. Roger Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, Feb 1981.
21. Noam Nisan and Amir Ronen. Algorithmic mechanism design. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing (STOC)*, pages 129–140, 1999.
22. Christos Papadimitriou, Michael Schapira, and Yaron Singer. On the hardness of being truthful. In *In 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2008.
23. Alex Samorodnitsky and Luca Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the 32nd Annual ACM symposium on Theory of computing (STOC)*, pages 191–199. ACM, 2000.
24. William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.
25. Jan Vondrak. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the 40th Annual ACM Symposium on Theory of computing (STOC)*, pages 67–74, New York, NY, USA, 2008. ACM.