Computation and Incentives in Combinatorial Public Projects

Dave Buchfuhrer* Michael Schapira† Yaron Singer

Abstract

The Combinatorial Public Projects Problem (CPPP) is an abstraction of resource allocation problems in which agents have preferences over alternatives, and an outcome that is to be collectively shared by the agents is chosen so as to maximize the social welfare. We explore CPPP from both a computational perspective and a mechanism design perspective. We examine CPPP in the hierarchy of complement-free (subadditive) valuation classes and present positive and negative results for both unrestricted and truthful computation.

1 Introduction

The *Combinatorial Public Project Problem* (CPPP), introduced in [13], is an abstraction of resource allocation environments in which an outcome, designated to be collectively shared by a group of agents, is chosen so as to maximize the agents' social welfare. An instance of CPPP consists of n agents, m resources, a valuation function $v_i : 2^{[m]} \to \mathcal{R}_{\geq 0}$ for each agent i, and an integer $k \in [m]$. The goal is to choose a set S of k resources for which the total social welfare $\Sigma_i v_i(S)$ is maximized. CPPP captures committee elections, network design, etc. (see [13, 16]).

Exploring the boundary of tractability in CPPP is a natural question from a computational perspective. Furthermore, this boundary plays a crucial role in algorithmic mechanism design [12]: for social welfare maximization problems, such as CPPP, computational tractability implies computationally-efficient truthful implementation via the celebrated VCG mechanisms. Intractability, on the other hand, can make truthful implementations with good approximation ratios impossible to obtain. This was recently demonstrated in [13], where it was shown for the first time, in the context of CPPP with submodular agents, that even if constant approximation ratios are achievable if one only cares about computational efficiency *or* truthfulness, combining *both* desiderata can lead to non-constant lower bounds.

Our aim here is to enable a computational and mechanism design understanding of CPPP. For unrestricted agent valuations, CPPP is known to be NP-hard to approximate well [16]. Hence, seeking interesting special cases of CPPP for which reasonable approximation ratios are attainable is natural. We consider the case where agents' valuations are *complement-free*, *i.e.*, cases in which agents' valuations are *subadditive* set functions. The class of complement-free, or subadditive, valuations encapsulates a rich hierarchy of valuation functions [9, 11] (see Fig. 1.1), that has been the focal point of the study of approximability in combinatorial auctions (see, *e.g.*, [2, 5, 6, 9, 17]).

In our study of the *computational feasibility* of CPPP, we search for two thresholds:

1. the point in the complement-free hierarchy of agents' valuations at which CPPP ceases to be tractable, and hence, for which computationally-efficient truthful implementation is no longer achievable via VCG mechanisms;

2. the point at which CPPP ceases to be approximable within a constant factor (*i.e.*, not in APX), and so CPPP cannot be well approximated even from a purely computational perspective (disregarding incentives).

In our study of *incentives* in CPPP, we focus both

*Supported by NSF CCF-0346991, CCF-0830787 and BSF 2004329.

†Supported by NSF grant 0331548.

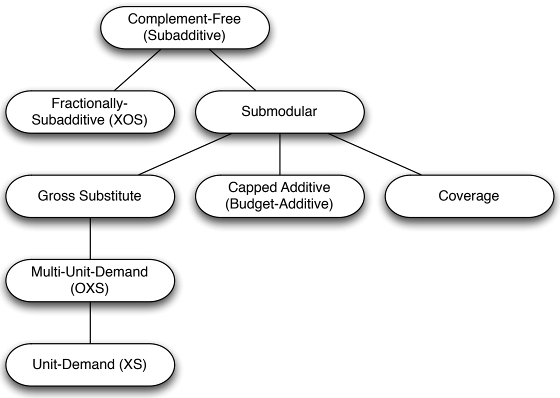

Figure 1.1: The Complement-Free Hierarchy

on general truthful mechanisms and on the broad class of "*VCG-based*", or "*maximal-in-range*", mechanisms, that have received much attention in algorithmic mechanism design recently (see [3, 2, 8, 13]).

CPPP is closely related to combinatorial auctions, that have gained the status of the paradigmatic problem in algorithmic mechanism design, and have been extensively studied across the computer science, microeconomics and operations research disciplines over past decades. To date, despite much research, big questions regarding the interplay between computation and truthfulness in combinatorial auctions, and in general, remain wide open. We believe that CPPP is a more suitable arena for exploring the fundamental connections between computation and truthfulness. Indeed, the advances in [13] and [7], and our results here, demonstrate the amenability of CPPP to an algorithmic mechanism design examination.

1.1 Our Results

We now briefly and informally survey our main results and their implications:

Is CPPP tractable? For CPPP with n agents, we show that even for the lowest (most restricted) class of valuations in the complement-free hierarchy, finding an *optimal* outcome is NP-hard. Specifically, CPPP is hard even for "*unit-demand*" valuations, in which every agent is only interested in getting a single resource. Moreover, going up just one step higher in the hierarchy, CPPP becomes hard even for a constant number of agents.

On the positive side, we present an optimal (and truthful via VCG payments) algorithm for an interesting special case of CPPP.

Is CPPP approximable? Our main inapproximability result is the following: We consider the class of *fractionally-subadditive* valuations, that is contained in the class of complement-free/subadditive valuations [5, 11]. We show that, unlike the case of CPPP with submodular valuations [13], for fractionally-subadditive valuations, no constant approximation ratio is achievable (unless P is in quasi-NP). Our result is nearly tight [16], answers an open question from [16], and is the first non-constant *computational* complexity lower bound for this class (and for subadditive valuations in general).

We present many other positive and negative approximability results: We show that the $1 - \frac{1}{e}$ approximation ratio for CPPP with submodular valuations [13] is tight even for the class of unit-demand valuations. In contrast, we present improved ratios for well-studied subclasses of submodular valuations.

Truthful mechanisms for CPPP. We present both algorithmic and hardness results for truthful computation. In particular, we present an inapproximability result for truthful mechanisms for CPPP that both strengthens and greatly simplifies the result in [13]. Surprisingly, our result holds even for the case of a single agent, thus raising intriguing question in algorithmic mechanism design. We also present a truthful constant-approximation mechanisms for interesting special cases of CPPP.

Finally, we present several inapproximability results for the class of "*VCG-based*", or "*maximal-in-range*", truthful mechanisms. In particular, we show that no constant approximation ratio is achievable for such mechanisms even for CPPP with unit demand valuations. Interestingly, we show that there exists a constant-factor approximation truthful mechanism for the same environment, thus establishing a gap between VCG-based and general truthful mechanisms.

Our results are summarized in the tables below

(Fig. 1.2 and 1.3, which also suggest interesting directions for future research).

1.2 Organization of the Paper

Each of the sections 2-6 focuses on exactly one class in the complement-free hierarchy. In Sec. 2 we present our results for unit-demand valuations. Sec. 3, 4, 5 and 6, deal with multi-unit-demand valuations, capped additive valuations, coverage valuations, and fractionally-subadditive valuations, respectively. We conclude and discuss our results in Sec. 7. Proofs of results that do not appear in the main body of the paper can be found in the appendix.

2 Unit-Demand: Hardness and Truthfulness

Unit-demand valuations. The simple class of *unit demand* valuations, in which every agent is only interested in getting a single resource, constitutes the lowest level of the complement-free hierarchy (see [9, 11], where unit-demand valuations are termed "XS").

Definition 2.1 (unit-demand valuations)**.** *A valuation function v is a unit demand valuation if $v(S) = \max_{i \in S} v(\{i\})$, for every $S \subseteq [m]$. Such a valuation is represented by a list of the m values $v(\{j\}), j \in [m]$.*

Our results in this section shall be proven for an even more restrictive class of valuations: unit demand valuation such that $v(\{i\}) \in \{0, 1\}$ for each resource i, and each agent has a value of 1 for at most 2 resources. We shall refer to this subclass of unit-demand valuations as "*2-{0,1}-unit-demand*".

Intractability. The following shows that CPPP is hard even with 2-{0,1}-unit-demand valuations:

Theorem 2.1. CPPP *with n unit-demand valuations is NP-hard to solve optimally.*

Theorem 2.2. *No algorithm for* CPPP *with n unit-demand valuations has an approximation ratio of $1 - \frac{1}{e} + \epsilon$ unless P=NP (for any constant $\epsilon > 0$).*

We note that the above hardness of approximation result is tight (a simple greedy algorithm obtains an approximation ratio of exactly $1 - \frac{1}{e}$). Observe that if there is only a constant number c of agents, one need only consider $\binom{m}{\min(c,k)} \in poly(m)$ sets of resources in order to find one which maximizes the social welfare, and hence CPPP with a constant number of unit-demand agents can be solved in polynomial time.

VCG-based mechanisms. We next consider the class of *VCG-based*, or *maximal-in-range* (MIR), mechanisms. Informally, MIR mechanisms output, for each possible input, the *optimal* outcome within a *fixed* set of outcomes. That is, a MIR mechanism M has *fixed* collection R of possible outcomes (subsets of resources of size k) and, for each n-tuple of agents' valuations $(v_1, \ldots, v_n)$, chooses a subset $r \in R$ that maximizes the social welfare $\Sigma_i v_i$ over R. The collection R is called "M's *range*".

In [16], a computationally-efficient $\frac{1}{\sqrt{m}}$-approximation MIR mechanism for CPPP with subadditive valuations is presented. We show that this approximation ratio is tight for MIR mechanisms even for 2-{0,1}-unit-demand valuations.

Theorem 2.3. *No computationally-efficient MIR mechanism can approximate* CPPP *with n 2-{0,1}-unit-demand valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

General truthful mechanisms. Theorem 2.3 shows that no constant-approximation MIR mechanisms exist even for CPPP with 2-{0,1}-unit-demand valuations. In contrast, we present a simple $\frac{1}{2}$-approximation non-MIR truthful mechanism for 2-{0,1}-unit-demand valuations, thus establishing a large gap between what is achievable via MIR and general truthful mechanisms.

Theorem 2.4. *There exists a computationally-efficient and truthful mechanism for* CPPP *with 2-{0,1}-unit-demand valuations that has an approximation ratio of $\frac{1}{2}$.*

3 Multi-Unit-Demand: Optimal Mechanism for 2 Agents

Multi-unit-demand valuations. Multi-unit-demand valuations (termed "OXS" in [9, 11]) are a

valuation class	no. of agents	appx. ratio r
unit-demand	constant	$r = 1$
	n	$r = 1 - \frac{1}{e}$ [New]
multi-unit-demand	$1, 2$	$r = 1$ [New]
	3	$2/3$ [New] $\leq r < 1$ [New]
	≥ 4	$1 - \frac{1}{e}$ [10] $\leq r < 1$ [New]
	≥ 10	$1 - \frac{1}{e}$ [10] $\leq r < 1 - \epsilon$ (FPTAS)[New]
	n	$r = 1 - \frac{1}{e}$ [New]
capped additive	1	$r = 1$
	constant ≥ 2	$r = 1 - \epsilon$ (FPTAS) [New]
	n	$r = 1 - \frac{1}{e}$ [New]
fractionally-subadditive	constant	$r = 1$
	n	$\max\{\frac{1}{n}, \frac{1}{\sqrt{m}}\}$ [16] $\leq r \leq 2^{-\frac{\log^{1-\gamma} n}{6}}$ [New]

Figure 1.2: Computational Results

valuation class	no. of agents	Truthful appx. ratio r	VCG-based appx. ratio r
2-{0,1} unit-demand	n	$1/2 \leq r < 1$ [New]	$r = \frac{1}{\sqrt{m}}$ [New]
unit-demand	n	?	
multi-unit-demand	3	$2/3 \leq r < 1$ [New]	
	n	?	
capped-additive	≥ 2	?	
coverage	1	$r = \frac{1}{\sqrt{m}}$ [New]	
fractionally-subadditive	n	?	

Figure 1.3: Truthful Mechanism Results. Question marks indicate that the only bounds known are a $\frac{1}{\sqrt{m}}$ lower bound based on the VCG-based mechanism in [16] and a purely computational upper-bound.

generalization of unit-demand valuations.

Definition 3.1 (multi-unit-demand). *A valuation function v is a multi-unit-demand valuation if there exist unit demand valuations $\{v^1, \ldots, v^w\}$ such that, for every $S \subseteq [m]$,*

$$v(S) = \max_{P=\{P^1,\ldots,P^w\}} \sum_{r \in [w]} v^r(P^r)$$

where the maximum is taken over all w-partitions $P = \{P^1, \ldots, P^w\}$ of S. Such a valuation agent is represented by a list of the w unit demand valuations.

How hard is CPPP with multi-unit-demand valuations? Unit-demand valuations are a subclass of multi-unit-demand valuations, and so our negative results in Sec. 2 for *CPPP* with n agents extend to multi-unit-demand valuations. What about a constant number of agents? One can easily show that CPPP with a single multi-unit-demand valuation is optimally solvable in a computationally-efficient manner using maximum matching on a bipartite graph. Below, we shall prove that CPPP with 2 multi-unit-demand valuations is tractable. We now present the following hardness result for CPPP with 3 multi-unit-demand valuations:

Theorem 3.1. CPPP *with* 3 *multi-unit-demand valuations is NP-hard to solve optimally.*

Theorem 3.1 leaves open the possibility of a PTAS for any constant number of agents. We rule out this possibility by presenting a hardness of approximation result for 10 (or more) agents.

Theorem 3.2. *There exists a positive constant ϵ such that it is NP-hard to approximate the social welfare in* CPPP *with* 10 *multi-unit-demand valuations within a ratio of* $1-\epsilon$.

Optimal mechanism for CPPP **with** 2 **agents.** We now show CPPP with 2 multi-unit-demand valuations can be optimally solved in a computationally-efficient manner via minimum cost matching. Thus, the use of VCG payments implies the existence of an optimal truthful mechanism.

Theorem 3.3. *There exists an optimal computationally-efficient and truthful mechanism for* CPPP *with* 2 *multi-unit-demand valuations.*

Our mechanism uses minimum cost flow, for which integral solutions can be found in polynomial time [14]. Minimum cost flow is similar to network flow, except that each edge has a cost and the goal is to find a flow of value f with minimum total edge cost. We now formally describe the mechanism:

Input: an instance of CPPP with two agents $1, 2$, where each agent i has a multi-unit-demand valuation v_i such that each v_i is composed of w_i unit-demand valuations $v_i^1, \dots, v_i^{w_i}$ (see Def. 3.1).

The mechanism:

1. **Step I: add "dummy" unit-demand valuations** (that equal 0 for all subsets of resources) if necessary to ensure that $w_1 = w_2 = w \geq k$.

2. **Step II: create a minimum-cost flow network (see Fig. 3.1)**. In addition to the source and target nodes s and t, the network contains node $p_{i,r}$ corresponding to agent i's rth unit-demand valuation v_i^r, and two nodes $q_{1,j}$ and $q_{2,j}$ for each resource $j \in [m]$. The edge set contains an edge from s to each node $p_{1,j}$, and an edge from each node $p_{2,j}$ to t. In addition, for each $j \in [m]$, create an edge from $q_{1,j}$ to $q_{2,j}$. Set the cost of each of these edges to be 1.

   Let v_{max} be a positive real value that is strictly higher than both agents' values for each single resource (say, $v_{max} = \max_{i\in[2],j\in[m]} v_i(\{j\})+1$). Create, for each $j \in [m], r \in [t]$, an edge from $p_{1,r}$ to $q_{1,j}$ of cost $v_{max} - v_1^r(\{j\})$ and an edge from $q_{2,j}$ to $p_{2,r}$ of cost $v_{max} - v_2^r(\{j\})$. (Observe that all costs are positive integers.)

   Set the capacities of all edges to be 1.

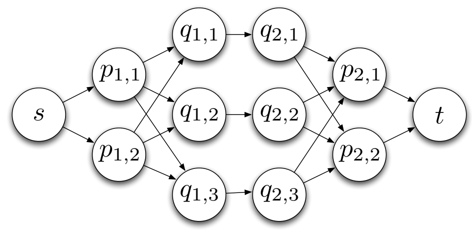

Figure 3.1: example of minimum-cost flow network construction for $w = 2$ and $m = 3$.

3. **Step III: compute a minimum-cost flow** f **with flow value** k **and integer flow along each edge** (*i.e.*, the flow along each edge is in $\{0, 1\}$).

4. **Step IV: set** S **to be the subset of** $[m]$ **such that** $j \in S$ **iff the flow in** f **along the edge from** $q_{1,j}$ **to** $q_{2,j}$ **is positive**. Observe that the k units of flow in f emanating from s must traverse exactly k edges of the form $(q_{1,j}, q_{2,j})$, and hence $|S| = k$.

5. **Step V: output the set of resources** S **and the VCG payments**. The VCG payment for agent 1 can be easily calculated by repeating Steps I through IV with v_1 set to 0 to arrive at a set T, which results in a payment of $v_2(T) - v_2(S)$. The payment for agent 2 can be found by reversing the roles of the agents.

Clearly, the mechanism is computationally efficient (recall that the computation of minimum-cost flow is tractable [14]). We are left with showing that the mechanism outputs the social-welfare maximizing subset of resources (the truthfulness of the mechanism then immediately follows from the VCG payment scheme).

Lemma 3.4. *The mechanism for* CPPP *with* 2 *multi-unit-demand valuations outputs a subset of resources of size k that maximizes the social welfare.*

Proof. Observe that the k units of flow in f emanating from s, and the k units of flow going into t traverse edges that have a total cost of $2k$, and that the k units of flow along the edges from $q_{1,j}$ nodes to $q_{2,j}$ nodes traverse edges that have a total cost of k. Hence, the total cost of these edges is $3k$ regardless of how the flow f is achieved.

Consider $j \in S$ (computed in Step IV of the mechanism). Observe that because there is 1 unit of flow traversing the edge $(q_{1,j}, q_{2,j})$, there must be exactly one incoming edge leading to node $q_{1,j}$, and exactly one outgoing edge leaving node $q_{1,j}$, on which the flow in f is 1. Consider a specific edge $(q_{1,j}, q_{2,j})$ and let $(p_{1,r}, q_{1,j})$ and $(q_{2,j}, p_{2,r'})$ be the edges through which the flow in f equals 1. Observe that the total cost of these two edges is $2v_{max} - v_1^r(\{j\}) - v_2^{r'}(\{j\})$. We define $c : S \to \mathbb{Z}^+$ to be the function that maps each $j \in S$ to the total cost of the incoming and outgoing edges to $q_{1,j}$ and $q_{2,j}$ (not including the edge between them).

Now, for some pair of partitions of S $P_1 = (P_1^1, \ldots, P_1^w)$ and $P_2 = (P_2^1, \ldots, P_2^w)$,

$$\begin{aligned}\sum_{j\in S} c(j) &= 2kv_{max} - \sum_{r=1}^{w} v_1^r(P_1^r) - \sum_{r=1}^{w} v_2^r(P_2^r) \\ &\geq 2kv_{max} - \max_{P=(P^1,\ldots P^w)} \sum_{r=1}^{w} v_1^r(P^r) \\ &\quad - \max_{P=(P^1,\ldots,P^w)} \sum_{r=1}^{w} v_2^r(P^r) \\ &= 2kv_{max} - v_1(S) - v_2(S)\end{aligned}$$

(the maxima in the above equations are taken over w-partitions of S)

Therefore, the total cost of flow f (including the edges leaving s, the edges entering t and the edges leading from the $q_{1,j}$'s to the $q_{2,j}$'s) is at least $2kv_{max} + 3k$ minus the social welfare of the set S. This lower bound is tight, as choosing the set maximizing the social welfare and the incoming and outgoing flows that correspond to the unit-demand valuations that maximize each v_i guarantees a total cost of exactly $2kv_{max} + 3k$ minus the maximum social welfare. Hence, the computation of the k-flow of minimum cost determines the value of the social-welfare maximizing outcome, and the set S produced achieves this maximum. □

Note that we can use this mechanism in a randomized fashion with 3 or more agents by selecting 2 of the agents uniformly at random, then running the mechanism on them.

Corollary 3.5. *There is a randomized, universally truthful mechanism for CPPP with 3 multi-unit-demand agent that achieves a* 2/3 *approximation of the social welfare in expectation.*

VCG-based mechanisms. Theorem 3.3 shows that there exists a computationally-efficient VCG mechanism for CPPP with 2 multi-unit-demand valuations. In contrast, we prove the following hardness of approximation result for CPPP with 3 or more multi-unit-demand valuations.

Theorem 3.6. *No computationally-efficient MIR mechanism can approximate* CPPP *with* 3 *multi-unit-demand valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

4 FPTAS for CPPP With Capped Additive Valuations

Capped additive valuations. Intuitively, a capped additive valuation is a valuation function that is additive (the value for each bundle of resources is the additive sum of the per-resource values) but cannot exceed some threshold.

Definition 4.1 (additive valuations)**.** *A valuation function v is* additive *(linear) if $v(S) = \Sigma_{j\in S}\ v(\{j\}$ for every $S \subseteq [m]$. Such a valuation is represented by a list of the m values $v(\{j\}), j \in [m]$.*

Definition 4.2 (capped additive valuations)**.** *A valuation function v is a* capped additive valuation *if there exists an additive valuation a, and a real value $B > 0$, such that, for each $S \subseteq [m]$, $v(S) = \min\{a(S), B\}$. Such a valuation is represented by* *the budget value b, followed by a list of the m values $v(\{i\}), i \in [m]$.*

NP-hardness and an FPTAS. 2-{0,1}-unit-demand valuations are a subclass of capped additive valuations (where $B = 1$), and so our negative results in Sec. 2 for $CPPP$ with n agents extend to capped additive valuations. What about a constant number of agents? Observe that finding the optimal outcome for a single agent is trivially in P (simply take the k most valued resources). We now show that, even with 2 agents, this is no longer the case.

Theorem 4.1. CPPP *with* 2 *capped additive valuations is NP-hard.*

However, using dynamic programming we obtain an FPTAS for any constant number of agents.

Theorem 4.2. *There exists an FPTAS for* CPPP *with a constant number of capped additive valuations.*

VCG-based mechanisms. We now show that MIR mechanisms cannot obtain constant approximation ratios even for CPPP with 2 capped additive valuations.

Theorem 4.3. *No computationally-efficient MIR mechanism can approximate* CPPP *with* 2 *capped additive valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

5 Coverage: Truthfulness With One Agent is Hard!

Coverage valuations. Intuitively, in a coverage valuation each resource represents a set of elements in some universe U, and the value of each set of resources $S \subseteq [m]$ equals the cardinality of the subset of U that is covered by its resources (that is, by the subsets of U represented by the resources in S).

Definition 5.1 (coverage valuations)**.** *A valuation function v is a coverage valuation if there exists a universe of elements U, subsets $S_1, \ldots, S_m \subseteq U$, and a real number $\alpha > 0$ such that $v(S) = \alpha|\bigcup_{j\in S} S_j|$, for every $S \subseteq [m]$. Such a valuation is represented by a list of the m sets $S_1, \ldots, S_m$.*

The hardness of being truthful with a single agent. [13] shows that no computationally-efficient and truthful mechanism for CPPP with 2 submodular valuations can obtain an approximation ratio better than $\frac{1}{\sqrt{m}}$ (while a constant non-truthful approximation exists). [16] proves the tightness of this result in [13] by presenting a $\frac{1}{\sqrt{m}}$-approximation mechanism that is both computationally-efficient and truthful.

One might suspect that the hardness of truthful computation in [13] stems from the conflict of interests between the two agents. When there is only one agent, the interests of the mechanism designer and the single agent are trivially aligned; both strive to better the agent's outcome (that is also the total social welfare). We now give the first evidence that, surprisingly, algorithmic mechanism design can be non-trivial even in single-player environments. We believe that this result raises intriguing questions in algorithmic mechanism design regarding the "right" solution concept for such environments.

Inapproximability result. We strengthen the result in [13] (which we prove for $n = 1$ and for coverage valuations). Our proof greatly simplifies the long and complicated proof in [13].

Theorem 5.1. *No computationally-efficient and truthful mechanism for* CPPP *with one coverage valuation has an approximation ratio within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

Sketch. We first present a simple characterization of truthful mechanisms for CPPP with a single agent, that can easily be generalized to hold for all 1-player mechanism design environments. Our characterization shows that every truthful mechanism is an *affine maximizer* (see [15, 13]).

Lemma 5.2. *If M is a truthful mechanism for* CPPP *with a single agent* 1*, then there exists a collection O of subsets of resources of size k, and a real number $w_o \in R$ for each $o \in O$ such that, for each valuation function v_1 of agent* 1*, the outcome $o(v_1) \in O$ that M outputs for v_1 is in $argmax_{o\in O}(v_1(o) - w_o)$.*

Importantly, the proof of the above lemma is trivial and does not follow in the footsteps of Roberts [15] (as in all proofs of this genre, including that in [13]).

Next, we show that, w.l.o.g., we can safely ignore the outcome weights (the w_o's) and restrict our attention to MIR mechanisms. We now use the connections between MIR mechanisms and the VC dimension (pointed out in [13]) to prove the following lemma (that concludes the proof of the theorem):

Lemma 5.3. *No MIR mechanism for* CPPP *with with one coverage valuation achieves an approximation ratio of at least $m^{-(\frac{1}{2}-\epsilon)}$ unless NP $\subseteq$ P/poly.*

□

6 Fractionally-Subadditive: Inapproximability Result

Fractionally-subadditive valuations. Intuitively, a valuation is fractionally-subadditive (termed "XOS" in [9, 11]) if it is the maximum of a collection of additive (linear) valuations (valuations where the value of each set is simply the additive sum of the per-resource values.

Definition 6.1 (fractionally subadditive)**.** *A valuation function v is* fractionally subadditive*) if there exist additive valuations $\{a^1, \ldots, a^t\}$ s.t. $v(S) = \max_{j\in t} v^j(S)$. Such a valuation is represented by a list of the t valuations $v^j, j \in [t]$.*

Constant number of agents We show that CPPP with a constant number of fractionally-subadditive valuations allows for a polynomial time algorithm.

Theorem 6.1. CPPP *with a constant number of fractionally-subadditive valuations can be solved in polynomial time.*

Inapproximability result. We now give a reduction from LABEL-COVER$_{max}$ to CPPP with n fractionally-subadditive valuations which preserves an approximation gap. First, we define LABEL-COVER$_{max}$ and discuss the complexity of its approximation. A LABEL-COVER$_{max}$ instance consists of a regular bipartite graph $G = (V_1, V_2, E)$, a set of n labels $N = \{1, \ldots, n\}$ and for each edge $e \in E$ a partial function $\Pi_e : N \to N$. We say that the edge $e = \{x, y\}$ for $x \in V_1, y \in V_2$ is satisfied if x is labeled with l_1 and y with l_2 such that $\Pi_e(l_1) = l_2$. The goal of LABEL-COVER$_{max}$ is to find an assignment of labels to the nodes in V_1 and V_2 such that each node has exactly one label and as many edges as possible are satisfied. It was shown in [1] that LABEL-COVER$_{max}$ is quasi-NP-hard to approximate.

Theorem 6.2 ([1])**.** *For any sufficiently small constant $\gamma > 0$, it is quasi-NP-hard to distinguish between the following two cases in* LABEL-COVER$_{max}$*: (1) YES case: all edges are covered, and (2) NO case: at most a $2^{-\log^{1-\gamma} n}$ fraction of the edges are covered, where n is the size of the* LABEL-COVER$_{max}$ *instance.*

We make use of Theorem 6.2 to show a similar hardness result for CPPP with fractionally-subadditive valuations.

Theorem 6.3. *Obtaining an approximation ratio of $2^{\frac{\log^{1-\gamma} b}{6}}$ for* CPPP *with fractionally-subadditive valuations where b is the size of the* CPPP *instance is quasi-NP-hard.*

Proof. We prove this using a gap-preserving reduction from LABEL-COVER$_{max}$: We are given an instance of LABEL-COVER$_{max}$ consisting of a graph $G = (V_1, V_2, E)$, a set of labels N and a set of partial functions Π_e for each $e \in E$. We create a CPPP instance with $|V_1|$ agents, one corresponding to each node in V_1. The resource set is $V_2 \times N$. We now define the fractionally-subadditive valuation v_i of each agent i. For every label $l \in N$, we define the additive valuation function $a_{i,l}$.

$$a_{i,l}(\{(j, l')\}) = \begin{cases} 1, & \{i, j\} \in E \text{ and } \Pi_{\{i,j\}}(l) = l' \\ 0, & \text{otherwise} \end{cases} .$$

So $a_{i,l}(S)$ represents how many edges incident with i are covered if we choose label l for vertex $i \in V_1$ and the best label from the set $\{l' : (j, l') \in S\}$ for vertex $j \in V_2$.

The fractionally-subadditive valuation of agent i is defined by

$$v_i(S) = \max_{l\in N}\{a_{i,l}(S)\}.$$

So agent i gets the value for the best possible choice of a single label for vertex i given the label choices for V_2 implied by S. We set the size of the set of resources to be chosen in our CPPP instance to be $|V_2|$.

If the LABEL-COVER$_{max}$ instance is a YES case, we can find a set of resources with social welfare $|E|$. Simply take any labeling that covers every edge and for every $j \in V_2$, choose the resource (j, l'), where j is labeled by l' in the labeling. Call this set S. Clearly, $v_i(S)$ equals the degree of node i, as if we choose l such that i is labeled by l, $\Pi_{\{i,j\}}(l) = l'$ for each $(j, l') \in S$. So the social welfare given these resources is $|E|$.

We now show that if the LABEL-COVER$_{max}$ instance is a NO case, the maximum social welfare is bounded by $2^{-\frac{\log^{1-\gamma} n}{6}}|E|$ for sufficiently large n. Note that if n' is the size of the LABEL-COVER$_{max}$ instance, our construction guarantees $n \leq (n')^2$. So our bound is at least

$$2^{-\frac{\log^{1-\gamma}[(n')^2]}{6}}|E| \geq 4 \cdot 2^{-\frac{\log^{1-\gamma} n'}{3}}|E|$$

for sufficiently large n'. In order to simplify our expressions in the rest of the proof, let $\alpha = 2^{-\frac{\log^{1-\gamma} n}{6}}$. Using the above bound, we see

$$\alpha \geq 4 \cdot 2^{-\frac{\log^{1-\gamma} n'}{3}}. \quad (1)$$

Suppose by way of contradiction that we reduced from a NO case, but the maximum social welfare is at least $\alpha|E|$.

Let S be a set of resources in the CPPP instance with a social welfare of at least $\alpha|E|$. Recall also that each agent i's fractionally-subadditive valuation v_i is defined as the pointwise maximum over a set of additive valuations. Let $a_{i,l}$ be the additive valuation in the set of valuations making up v_i for which $a_{i,l}(S)$ is maximized (and so $v_i(S) = a_{i,l}(S)$). If we fix a choice of j, $a_{i,l}$ assigns a value of 1 to at most one of the resources (j, l') for $l' \in N$. Moreover, $a_{i,l}$ can only assign value to a resource (j, l') if $\{i, j\} \in E$. We say that an edge between vertex $i \in V_1$ and vertex $j \in V_2$ is satisfied by the set S if $(j, \Pi_{\{i,j\}}(l)) \in S$. Observe that the total social welfare value of S equals the number of edges satisfied by S.

Let d be the number of incoming edges of a vertex in V_2. Since G is a regular bipartite graph, $d = \frac{|E|}{|V_2|}$. Let V_2' denote all vertices $v \in V_2$ in which the number of edges incident on v satisfied by S is at least $\frac{\alpha}{2}d$. A counting argument shows that $|V_2'| \geq \frac{\alpha}{2}|V_2|$. If $|V_2'|$ were less than $\frac{\alpha}{2}|V_2|$, the number of satisfied edges incident upon vertices in V_2' is at most $|V_2'|d < \frac{\alpha}{2}|E|$, and the number of satisfied edges incident upon vertices outside of V_2 would be less than $|V_2|\frac{\alpha}{2}d = \frac{\alpha}{2}|E|$. So summing these, we would see that the number of satisfied edges is less than $\alpha|E|$, a contradiction. So $|V_2'| \geq \frac{\alpha}{2}|V_2|$.

If S contains ℓ resources of the form (j, l) for a fixed j and ℓ different values $l \in N$, we say that j is labeled ℓ times by S. Since there are $|S| = |V_2|$ resources, at most $\frac{\alpha}{4}|V_2|$ of the nodes $j \in V_2$ are labeled more than $\frac{4}{\alpha}$ times by S. So letting V_2'' be the subset of V_2' which is labeled at most $\frac{4}{\alpha}$ times by S, $|V_2''| \geq \frac{\alpha}{4}|V_2|$.

Since S labels each $j \in V_2''$ at most $\frac{4}{\alpha}$ times, and S satisfies at least $\frac{\alpha}{2}d$ edges incident upon each vertex in V_2'', we can find a single $s_j \in S$ that satisfies at least $\frac{\alpha/2}{4/\alpha}d = \frac{\alpha^2}{8}d$ of the edges incident upon j. So if we label each $j \in V_2''$ according to S_j and label each $i \in V_1$ by the l such that $v_i(S) = a_{i,l}(S)$, we have a labeling that satisfies at least $|V_2''|\frac{\alpha^2}{8}d = \frac{\alpha^3}{32}|E|$ edges, regardless of how the vertices in $V_2 - V_2''$ are labeled. This contradicts that we had a NO case, as we can see by (1) that $\frac{\alpha^3}{32}|E| > 2^{-\log^{1-\gamma} n'}|E|$.

Thus, we see that the maximum social welfare of our CPPP is at least $|E|$ if we reduced from a YES case and at most $\alpha|E|$ if we reduced from the NO case. Therefore it is quasi-NP-hard to achieve an approximation ratio of $\alpha = 2^{-\frac{\log^{1-\gamma} n}{6}}$. □

7 Discussion and Open Questions

In our exploration of CPPP we have presented positive and negative results for truthful and unrestricted computation. Our results highlight interesting phenomena in algorithmic mechanism design, and improve our understanding of the tractability-intractability boundary for this natural computational and economic environment.

The focus in algorithmic mechanism design is on the tension between computation and truthfulness.

Our results for CPPP identify extremely simple combinatorial environments where the two desiderata clash, and that are therefore a natural arena for the investigation of the complex interplay between computational efficiency and incentive compatibility.

We leave many important questions wide open. We still lack a good understanding of the power of computationally-efficient and truthful mechanisms for CPPP (see Fig. 1.3), and leave bridging the approxmability gaps between the upper and lower bounds in Fig. 1.2 as an open question.

References

[1] Sanjeev Arora and Carsten Lund. Hardness of approximations. pages 399–446, 1997.

[2] Shahar Dobzinski, Noam Nisan, and Michael Schapira. Approximation algorithms for combinatorial auctions with complement-free bidders. In *STOC*, 2005.

[3] Shahar Dobzinski and Mukund Sundararajan. On characterizations of truthful mechanisms for combinatorial auctions and scheduling. In *ACM conference on electronic commerce*, 2008.

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

[5] Uriel Feige. On maximizing welfare when the utility functions are subadditive, 2006. In STOC.

[6] Uriel Feige and Jan Vondrak. Approximation algorithms for allocation problems: Improving the factor of $1 - 1/e$, 2006. In FOCS.

[7] Anupam Gupta, Katrina Ligett, Frank McSherry, Aaron Roth, and Kunal Talwar. Differentially private approximation algorithms. *CoRR*, abs/0903.4510, 2009.

[8] Ron Lavi, Ahuva Mu'alem, and Noam Nisan. Towards a characterization of truthful combinatorial auctions. In *FOCS*, 2003.

[9] Benny Lehmann, Daniel Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. In *ACM conference on electronic commerce*, 2001.

[10] 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.

[11] Noam Nisan. Bidding and allocation in combinatorial auctions. In *ACM Conference on Electronic Commerce*, 2000.

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

[13] Christos H. Papadimitriou, Michael Schapira, and Yaron Singer. On the hardness of being truthful. In *FOCS*, 2008.

[14] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization; Algorithms and Complexity.* Dover Publications, 1998.

[15] Kevin Roberts. The characterization of implementable choice rules. In *Aggregation and Revelation of Preferences*, pages 321–349. 1979.

[16] Michael Schapira and Yaron Singer. Inapproximability of combinatorial public projects. In Christos H. Papadimitriou and Shuzhong Zhang, editors, *WINE*, volume 5385 of *Lecture Notes in Computer Science*, pages 351–361. Springer, 2008.

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

A Unit-Demand Valuations

We show that approximating the social welfare of a combinatorial public project with n unit-demand agents to a factor of $1 - 1/e + \epsilon$ is NP-hard for any $\epsilon > 0$ with a reduction from MAX-t-COVER.

Definition A.1 (MAX-t-COVER)**.** *MAX-t-COVER takes as input a collection of subsets $\mathcal{F}$ of a set A and an integer t. The goal is to find t sets in $\mathcal{F}$ which have a union of maximum cardinality.*

It was shown in [4] that MAX-t-COVER cannot be approximated to within $1 - 1/e + \epsilon$ for any constant $\epsilon > 0$ unless $P = NP$.

Theorem A.1. *For any constant $\epsilon > 0$, is NP-hard to approximate the social welfare of CPPP with n unit-demand agents to within $1 - 1/e + \epsilon$.*

Proof. We will now show an approximation preserving reduction from MAX-t-COVER. Consider a MAX-t-COVER instance over set A with $\mathcal{F} = \{S_1, \ldots, S_\ell\}$ and number of sets to be chosen t. We create a CPPP instance with resource set $\mathcal{F}$ and $|A|$ agents, one corresponding to each element of A. The agent corresponding to element a values each item $S_i \in \mathcal{F}$ as

$$v_a(S_i) = \begin{cases} 1, & a \in S_i \\ 0, & \text{otherwise} \end{cases} .$$

So the value for agent a is 1 if a is covered by the chosen set and 0 otherwise. Thus, the social welfare is number of covered items, or the cardinality of the union of the chosen sets. By setting the number of resources allowed to be chosen to $k = t$, we see that if we can approximate the social welfare to within any factor α, we get an α-approximation of MAX-t-COVER as well. So by [4], an approximation of $1 - 1/e + \epsilon$ is not achievable. □

Note that the above proof required $|\mathcal{F}|$ agents, each with very simple 0/1 valuation functions. It may be tempting to try to reduce the number of agents at the expense of using more complicated valuation functions. This can only go so far though, as if there are only a constant number c of agents, one need only consider $\binom{m}{\min(c,k)} \in poly(m)$ sets of resources in order to find one which maximizes the social welfare.

Theorem A.2. *No computationally-efficient MIR mechanism can approximate* CPPP *with n 2-$\{0,1\}$-unit-demand valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

Proof. We begin by noting that in [13] it was shown that any algorithm for CPPP which achieves an approximation ratio of at least $m^{1/2-\epsilon}$ has a range of size $\Omega(e^{m^\epsilon})$. This proof required that for any $V \subseteq [m]$, it is possible to create a set of agents such that the social welfare is $v(S) = |V \cap S|$. This is easy to do with n 2-{0,1}-unit-demand agents, resulting in the following useful lemma:

Lemma A.3. *Any maximal-in-range mechanism for CPPP with n 2-$\{0,1\}$-unit-demand agents which achieves an approximation ratio of at least $m^{1/2-\epsilon}$ must have a range of size $\Omega(e^{m^\epsilon})$.*

From this, we can use the Sauer-Shelah lemma to see that the range has a VC dimension at least m^α for some constant $\alpha > 0$. This large range allows us to perform reductions similar to the ones we usein our NP-hardness proofs to show inapproximability. We begin with the modified unit-demand reduction.

As shown above, any maximal-in-range mechanism which approximates better than $m^{1/2-\epsilon}$ must have a range with VC-dimension at least m^α. Re-order the items such that the m^α corresponding to this VC-dimension are the set $[m^\alpha]$. We show a reduction from vertex cover with $m^\alpha/2$ edges. Let $k' < k$ be the target size of the vertex cover. The first $|V|$ items correspond to the vertices.

The first $2|E| = m^\alpha$ agents correspond 2 to each edge, and have valuation 1 if the corresponding edge is covered by one of the vertices corresponding to an item chosen from $[m^\alpha]$, we have $m - m^\alpha$ agents, one corresponding to each item outside of $[m^\alpha]$ where the agent corresponding to item i has valuation

$$v_i(S) = \begin{cases} 1, & i \in S \\ 0, & \text{otherwise} \end{cases} .$$

If a single edge is unsatisfied, more social welfare can be obtained by adding an item from $[m^\alpha]$, where some item adds at least 2 to the social welfare than by adding an item from $[m] - [m^\alpha]$, which only contributes 1. So if the minimum vertex cover has size k^*, the maximum social welfare is $2|E| + (k - k')$. Furthermore, $\mathcal{M}$ will find this maximum, as it's range includes every subset of $[m^\alpha]$, padded out with arbitrary elements from $[m] - [m^\alpha]$ to reach size k. Thus,

$\mathcal{M}$ can be used to find the size of the minimum vertex cover and therefore cannot run in polynomial time unless $NP \subset P/poly$. ☐

Theorem A.4. *There exists a computationally-efficient and truthful mechanism for* CPPP *with 2-{0,1}-unit-demand valuations that has an approximation ratio of* $\frac{1}{2}$.

Proof. **The mechanism :**

1. for each resource j let $s_j = |\{i : v_i(\{j\}) = 1\}|$.

2. sort the m resources in decreasing order by the value of s_j, breaking ties in favor of resources with lower indices.

3. output the set S consisting of the k first resources in the above ordering.

This is clearly an efficient algorithm, as sort only requires $O(n \log n)$ time, and in this case bucket sort can be used to achieve a linear time mechanism.

The mechanism as described essentially allows agents to vote for 2 resources, then chooses the k with the most votes. An agent only benefits from adding votes to the 2 resources that he actually desires, as adding other items to the top k does not improve his social welfare. As the two resources are desired equally, there is no advantage to voting for one of the resources the bidder desires and not the other. So there is never an incentive for an agent to not declare his valuation truthfully.

We will now see that this has an approximation ratio of 1/2. Every agent has a value of either 0 or 1 for the chosen set. If an agent has a value of 1, we call that agent satisfied. For each resource j, let s_j be the number of agents satisfied by j. For any set T, $\sum_{j \in T} s_j$ is an upper bound on the social welfare of T. Clearly, S maximizes $\sum_{j \in S} s_j$ for sets of size k, so $\sum_{j \in S} s_j$ is an upper bound on the maximum social welfare. Furthermore, each agent is satisfied by at most 2 items in S, so the social welfare of S is at least $\sum_{j \in S} s_j/2 = 1/2 \sum_{j \in S} s_j$, which is at least 1/2 the maximum social welfare. ☐

B Multi Unit-Demand Valuations

Theorem B.1. CPPP *with* 3 *multi-unit-demand valuations is NP-hard to solve optimally.*

Proof. We reduce from 3-Dimensional Matching (3DM). Given a 3DM instance $M \subseteq [q] \times [q] \times [q]$, the goal is to determine whether there exists a set $M' \subset M$ of size q such that no to members of M' share a coordinate. Our reduction is as follows. The set of items is M. The number of items to be chosen is $k = q$.

The ith agent values set S by the number of different values for the ith coordinate in set S. This valuation is multi-unit-demand because it can be built out of the q unit-demand valuations that value 1 to any set containing an item with a j in the ith coordinate, for any $1 \leq j \leq q$. By partitioning the items by their jth coordinate, we see that the corresponding multi-unit-demand valuation is just the sum of these unit-demand valuations, which is the number of different values of the ith coordinate.

The maximum social welfare of this auction is $3q$ iff the 3DM instance is positive. Clearly, any set M' of size q will have social welfare $3q$ iff none of the items in the set share a coordinate, as the maximum value of q is achieved by each agent none of the items share the coordinate corresponding to that agent. ☐

Definition B.1 (MAX-3SAT-5)**.** *MAX-3SAT-5 is the maximization version of 3SAT in which each variable appears in exactly 5 clauses.*

[4] showed that there exists a constant $\epsilon > 0$ it is NP-hard to distinguish between the case that all clauses are satisfiable and that a $1 - \epsilon$ fraction of the clauses are satisfiable in a MAX-3SAT-5 instance.

Theorem B.2. *There exists a positive constant* ϵ *such that it is NP-hard to approximate the social welfare of CPPP with* 10 *multi-unit-demand agents to a ratio of* $1 - \epsilon$.

Proof. Consider an instance of MAX-3SAT-5 with clauses $c_1, \ldots, c_\ell$. Because each clause has 3 variables and each variable is contained in 5 clauses, there are

$3\ell/5$ variables $v_1, \ldots, v_{3\ell/5}$. We will start by reducing to an instance with n unit-demand agents, then demonstrate that these agents can be compressed into 10 multi-unit-demand agents without changing the social welfare function.

There are $6\ell/2$ items, 2 corresponding to each variable. For each variable v_i, we will have two items labeled i and $\bar{i}$. Choosing i corresponds to setting v_i to true, while choosing $\bar{i}$ corresponds to setting v_i to false. We allow $k = 3\ell/5$ items to be chosen, so that one value can be chosen for each variable.

There are two classes of agents. The first class has ℓ agents, one corresponding to each clause. The agent corresponding to clause c_i has value 1 for each item j such that v_j is in c_i and 1 for each item $\bar{j}$ such that $\neg v_j$ is in c_i. Thus, these agents have value 1 if their clause is satisfied and 0 otherwise.

The second class of agents has $5 \cdot (3\ell/5) = 3\ell$ agents, 5 for each variable. The 5 agents for each variable are identical, and the agents corresponding to v_i have value 1 for items i and $\bar{i}$ and 0 for all other items. If there is some item i for which both i and $\bar{i}$ are chosen, then by the pigeonhole principle, there is some j for which neither j nor $\bar{j}$ is chosen. This leads to a loss of 5 to the social welfare from these agents compared to replacing one of $i, \bar{i}$ with one of $j, \bar{j}$, while having both i and $\bar{i}$ adds at most 5 to the social welfare of the clause agents compared to keeping just one of these. So these agents allow us to modify any choice of items such that only one of $i, \bar{i}$ is chosen for each i without reducing the social welfare. Thus, we will assume WLOG that all choices correspond to an assignment to the variables of the MAX-3SAT-5 instance.

Let δ be a positive constant such that it is NP-hard to distinguish between the case that all clauses are satisfiable in a MAX-3SAT-5 instance and that only a $1-\delta$ fraction are. Choose any positive constant $\epsilon < \delta/4$. If all ℓ clauses are satisfiable, then by choosing the items corresponding to the assignment that satisfies them, we get a maximum of ℓ social welfare from the clause agents and 3ℓ social welfare from the rest for a total of 4ℓ. So if we approximate the social welfare to $1-\epsilon$, we get a social welfare of at least $(1-\epsilon)4\ell > 4\ell - \delta\ell$. As argued above, we can assume that 3ℓ comes from the second class of agents described, so more than $(1-\delta)\ell$ social welfare comes from the clause agents, telling us that there is an assignment satisfying more than a $1-\delta$ fraction of the clauses, allowing us to distinguish between the case that all clauses are satisfiable and the case that at most a $1-\delta$ fraction are. This is NP-hard, so approximating the social welfare to within $1-\epsilon$ is NP-hard as well.

Finally, we show how these 4ℓ unit-demand agents can be compressed into 10 multi-unit-demand agents. We do so by combining groups of unit-demand agents that don't value any of the same items into a single multi-unit-demand agent. Then the multi-unit-demand value to that agent of any set is the sum of the values of each of the individual unit-demand agents, as we can partition the items according to which agent value which items. Note that we can assume WLOG that there is no i for which the clause agents only value either i or $\bar{i}$ as we can simply remove any such i, then perform a $1-\epsilon$ approximation and add the i back knowing whether to choose i or $\bar{i}$ and resulting in an improved approximation. Thus, for each item i or $\bar{i}$, there are only 4 clause agents that value the item.

We combine the unit-demand agents greedily, starting with the clause agents. For each unit-demand agent, simply add its valuation to any multi-unit-demand agent which does not yet have value for any of the items the unit-demand agent values. Since each item is valued by at most 3 other clause agents and there are three items valued by any clause agent, there are at most $3 \cdot 3 = 9$ multi-unit-demand clause agents with values for these 3 items. Thus, one of the 10 multi-unit-demand agents can accommodate the value of this unit-demand agent.

Now, we add the second class of agents, which for each i value items i and $\bar{i}$. There are 5 such agents for each i. Similarly, for each i, there are 5 unit-demand agents which value either i or $\bar{i}$. So the 5 agents from the second class can be added to the valuations of the 5 multi-unit-demand agents that do not yet value items i or $\bar{i}$ from the clause valuations.

Thus, we can compress these valuations into 10 multi-unit-demand agents in such a way that a $1-\epsilon$ approximation allows for distinguishing between the case that all clauses can be satisfied and that at most a $1-\delta$ fraction can. $\square$

This demonstrates that, unlike in combinatorial auctions, the number of multi-unit-demand agents can determine the complexity of social welfare maximization in CPPP.

Corollary B.3. *There is a randomized, universally truthful mechanism for CPPP with 3 multi-unit-demand agent that achieves a $2/3$ approximation of the social welfare in expectation.*

Proof. Consider choosing 2 of the agents uniformly at random, then running the mechanism from Theorem 3.3 on them. This is universally truthful, as the agents not selected are ignored and thus have no incentive to lie, and the mechanism is truthful for the two selected agents. For 3 agents, this gives an expected 2/3 approximation of the social welfare, as the ignored agent contributes 1/3 of the maximum social welfare in expectation. $\square$

Theorem B.4. *No computationally-efficient MIR mechanism can approximate* CPPP *with* 3 *multi-unit-demand valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

Proof. The proof here is essentially the same as that of Theorem 2.3, in that the proof of NP-hardness can be modified to make use of the smaller range by setting each of the items outside of $[m^\alpha]$ to add 1/2 to the social welfare. In this case, we can fold this value in to the multi-unit-demand valuation of one of the agents without affecting the social welfare gain from the items in $[m^\alpha]$. Running $\mathcal{M}$ in this case yields a social welfare of $3q + (k-q)/2$ iff there is a set of q items from $[m^\alpha]$ with social welfare $3q$, corresponding to a 3-dimensional matching. So $\mathcal{M}$ cannot run in polynomial time unless $NP \subset P/poly$. $\square$

C Capped Additive Valuations

Theorem C.1. *The Combinatorial Public Projects problem with 2 budget additive agents is NP-hard.*

Proof. We reduce from Subset Sum, where we are given a set of positive integers $v_1, \dots, v_\ell$ and a target t, and the goal is to find a subset of $v_1, \dots, v_\ell$ that sums to t. Given an instance of Subset Sum, we construct an instance to our problem with $m = 2\ell$ resources, $k = \ell$, and 2 agents with valuations $v_1(S) = \min\{\sum_{j\in S} \bar{2}v_{1j}, 2t\}$ and $v_2(S) = \min\{\sum_{j\in S} \bar{v}_{2j}, B\}$, where $B = k \cdot \max_j v_j$ and $\bar{v}_{1j}, \bar{v}_{2j}$ are:

$$\bar{v}_{1j} = \begin{cases} 2v_j, & j \le \ell \\ 0, & \text{otherwise} \end{cases}$$

$$\bar{v}_{2j} = \begin{cases} B/k - v_i, & j \le m \\ B/k, & \text{otherwise} \end{cases} .$$

Observe that if there exists a subset S s.t. $\sum_{i\in S} a_i = t$, by choosing the set of resources $S' = S \cup \{\ell+1 \dots 2\ell - |S|\}$ we have $v_1(S') + v_2(S') = B + t$.

Conversely, consider a subset of resources in our problem of size m with social welfare of at least $B+t$. Consider the set of items with index at most ℓ. If the corresponding items summed to more than t, then agent 2 would have total value less than $B-t$, while agent 1 would have value of only $2t$, for a total value of less than $B+t$. If the corresponding items summed to less than t, then the social welfare would be B, plus the sum of the corresponding set, which is less than $B+t$. So the subset must have a sum of exactly t. $\square$

Theorem C.2. *No computationally-efficient MIR mechanism can approximate* CPPP *with* 2 *capped additive valuations within $m^{-(\frac{1}{2}-\epsilon)}$ (for any constant $\epsilon > 0$) unless $NP \subset P/poly$.*

Proof. Since the reduction for budget additive agents did not rely on minimizing the number of items required to match the social welfare of the set of all items, we cannot rely on the same trick here as in the above two theorems. Instead, we rely on the structure of the reduction. The number of items which are valued by agent 1 at 0 and agent 2 at B/k doesn't affect the proof (as long as it's larger than ℓ and at least k), so we just add $m - m^\alpha$ more of these. The particular value of k also doesn't matter, as long as it's at least ℓ, so losing control of how k relates to m^α isn't an issue. Thus, using the same reduction after this modification, we see that $\mathcal{M}$ can be used to solve subset sum instances of size m^α, and is therefore does not run in polynomial time unless $NP \subset P/poly$. $\square$

Theorem C.3. *For the Combinatorial Public Projects problem with a constant number of budget additive agents there is a FPTAS.*

Proof. We will use a dynamic programming procedure. For $b = \max_{i \in n} b_i$, we divide the interval $[0, b]$ into $\frac{n \cdot m}{\epsilon}$ segments, each of length $\frac{\epsilon b}{n \cdot m}$, and denote $p(x) = \lfloor x \cdot mn/\epsilon b \rfloor$. We will maintain an n dimensional table with $(\frac{n \cdot m}{\epsilon})^n$ entries, denoted A, where in each entry $a_{ij\ldots k}$ we will store a subset S for which $p(v_n(S)) = k, p(v_2(S)) = j, \ldots p(v_n(S)) = k$, if such a subset exists. For convenience, for a given subset S we will denote $A(S)$ to be its corresponding entry in the table.

Assume some arbitrary ordering $\{1 \ldots m\}$ over the set of resources, and consider the following procedure. We initialize the table with the empty set in all entries. At stage j, for each subset $S \in A$, s.t. $|S| < k$, let $T = A(S \cup \{j\})$. If $|S \cup \{j\}| \leq |T|$ or $T = \emptyset$, we set $A(S \cup \{j\}) = S \cup \{j\}$. After the mth stage we iterate over al entries in the table, and choose the subset with highest social welfare. The procedure runs in $O(m \cdot (\frac{mn}{\epsilon})^n)$ steps, which is polynomial in m and $1/\epsilon$ as required.

Let O denote the optimal solution, $O_j = \{i \leq j | i \in O\}$. By induction on the stage of the algorithm, we can show that at stage ℓ there is a subset S_ℓ s.t. $S_\ell \in A(O_\ell)$, $|S| \leq |O_\ell|$ and for every agent i we have that $v_i(O_\ell) - v_i(S_\ell) \leq \ell \cdot \frac{\epsilon b}{m \cdot n}$. For $\ell = 1$ the claim is trivial. For a $\ell \leq m$, if $\ell \notin O_\ell$, the claim trivially holds from the inductive hypothesis. Otherwise, there is a subset $S_{\ell-1}$, s.t. $|v_i(O_\ell) - v_i(S_{\ell-1} \cup \{\ell\})| = |v_i(O_{\ell-1} \cup \{\ell\}) - v_i(S_{\ell-1} \cup \{\ell\})| \leq (\ell - 1) \cdot \frac{\epsilon b}{m \cdot n}$ for every i, and $|S_{\ell-1}| \leq |O_{\ell-1}|$. If another subset $S' \neq S \cup \{\ell\}$ is stored in $A(O_\ell)$ then $|v_i(S \cup \{\ell\}) - v_i(S')| \leq \frac{\epsilon b}{m \cdot n}$, $|S'| \leq |S \cup \{\ell\}|$, and the claim holds. □

D Coverage Valuations

Theorem D.1. *No computationally-efficient and truthful mechanism for* CPPP *with one coverage valuation obtains an approximation ratio within* $m^{-(\frac{1}{2}-\epsilon)}$ *(for any constant* $\epsilon > 0$*) unless* $NP \subset P/poly$.

Proof. We first present a simple characterization of truthful mechanisms for CPPP with a single agent, that can easily be generalized to hold for all 1-player mechanism design environments. Our characterization shows that every truthful mechanism is an *affine maximizer* (see [15, 13]).

Lemma D.2. *If* M *is a truthful mechanism for* CPPP *with a single agent* 1*, then there exists a collection* O *of subsets of resources of size* k*, and a real number* $w_o \in R$ *for each* $o \in O$ *such that, for each valuation function* v_1 *of agent* 1*, the outcome* $o(v_1) \in O$ *that* M *outputs for* v_1 *is in* $argmax_{o \in O}(v_1(o) - w_o)$.

Proof. Let O be the collection of outcomes (subsets of $[m]$ of size k) that agent 1 can achieve (*i.e.*, outcomes that the mechanism outputs for some valuation of 1). In truthful mechanisms, the payment of a player is independent of his own valuation function, and can only depend on the outcome and on the valuations of the other players. Because we are dealing with a single-player environment, we can associate each outcome $o \in O$ with the payment that M outputs for that outcome w_o. Now, M's truthfulness implies that, if M outputs the outcome $o \in O$ for the valuation v_1, then it must hold $v_1(o) - w_o \geq v_1(o') - w_{o'}$ for each $o' \in O$ (otherwise, 1 is better off lying and announcing the valuation for which M outputs o'). The lemma follows. □

We now prove an inapproximability result for truthful mechanisms. Let M be a truthful mechanism. We now know that there exists a collection O of subsets of resources of size k, and per-outcome "weights" (the w_o's), such that M exactly optimizes 1's value over O, given the outcome weights. Let $\alpha = \max_{o \in O} 2|w_o| + 1$. Observe that for every $S, T \in O$ such that $v_1(S) \neq v_1(T)$ it holds that $|v_1(S) - v_1(T)| > |w_S - w_T|$. So we can safely ignore the outcome weight, as maximizing $v_1(S) - w_S$ also maximizes $v_1(S)$. Hence, from now on, we need only consider MIR mechanisms (that for each possible v_1 output an outcome $o(v_1) \in O$ that is in $argmax_{o \in O} v_1(o)$). The following lemma concludes the proof of the theorem.

Lemma D.3. *No MIR mechanism for* CPPP *with with one coverage valuation achieves an approximation ratio of at least* $m^{-(\frac{1}{2}-\epsilon)}$ *unless* $NP \subseteq P/poly$.

Proof. Let M be a MIR mechanism that obtains an approximation ratio better than $m^{-(\frac{1}{2}-\epsilon)}$. [13] shows that the VC dimension of M's range (that is, O) must be at least m^α for some constant $\alpha > 0$ (using essentially the arguments used in the proof of Theorem 2.3).

We now show a reduction the NP-hard t-COVER [4] with m^α sets. In t-COVER, the input is m^α subsets of a universe E, $T_1, \ldots, T_{m^\alpha}$, and an integer t, and the objective is to determine whether there are t sets that cover E. We now construct the valuation function of agent 1. We create a universe U that consists of two disjoint copies of E, E_1 and E_2, plus a set of $m-t$ additional elements, E_3. To define the coverage valuation v_1 we need to define the sets $S_1, \ldots, S_m \subseteq U$ (see Def. 5.1). Re-order the resources such that the m^α resources corresponding to this VC-dimension are the set $[m^\alpha]$. For each $j \in m^\alpha$ let the set S_j be the subset of U that covers all elements in E_1 and E_2 that are covered by T_j. For each $j \in \{m^\alpha + 1, \ldots, m\}$ let S_j be a set that covers a single unique element in E_3.

Observe that if the minimal number of sets needed to cover E in t-COVER is r, then any optimal outcome in our $CPPP$ instance is one that contains r resources corresponding to r covering sets in t-COVER, and $k-r$ additional resources from E_3 (chosen arbitrarily). The output of M thus determines the value of r. If $r \leq t$ then there exist t sets in MAX-t-COVER that cover E, otherwise no such t sets exist. Observe that the reduction is polynomial, yet is not uniform (because of the non-constructiveness of the Sauer-Shelah Lemma), and hence our result is dependent on the computational assumption that NP is not contained in P/*poly*. □

□

E Fractionally-Subadditive Valuations

Although multi-unit demand agents are a special case of fractionally subadditive agents in general, this leads to an exponential blowup in description size with our choice of representation. So while 3 multi-unit demand agents creates an NP-hard problem, a constant number of fractionally subadditive agents allow a polynomial time algorithm.

Theorem E.1. *CPPP with a constant number of fractionally-subadditive valuations can be solved in polynomial time.*

Proof. Each fractionally-subadditive valuation is the maximum over linearly many additive valuations. If one of these additive valuations is chosen for each agent, the resulting auction can be trivially solved in polynomial time. This solution gives a lower bound on the maximum social welfare. If the additive valuations chosen happen to be the ones that exhibit the maximum in an optimal allocation, the solution found will also be optimal. Thus, by enumerating over all possible choices, an optimal allocation can be found. If there are c agents with at most ℓ additive valuations each, there are $O(\ell^c) \subseteq poly(\ell)$ choices to enumerate over. Thus, the solution to the auction can be found in polynomial time. □