

On the Complexity of Optimal Lottery Pricing and Randomized Mechanisms

Xi Chen*, Ilias Diakonikolas[†], Anthi Orfanou*, Dimitris Paparas*, Xiaorui Sun* and Mihalis Yannakakis*

*Computer Science Department

Columbia University, New York, NY 10027

Email: {xichen, anthi, paparas, xiaoruisun, mihalis}@cs.columbia.edu

[†]School of Informatics

University of Edinburgh

Email: ilias.d@ed.ac.uk

Abstract

We study the optimal lottery problem and the optimal mechanism design problem in the setting of a single unit-demand buyer with item values drawn from independent distributions. Optimal solutions to both problems are characterized by a linear program with exponentially many variables.

For the menu size complexity of the optimal lottery problem, we present an explicit, simple instance with distributions of support size 2, and show that exponentially many lotteries are required to achieve the optimal revenue. We also show that, when distributions have support size 2 and share the same high value, the simpler scheme of item pricing can achieve the same revenue as the optimal menu of lotteries. The same holds for the case of two items with support size 2 (but not necessarily the same high value).

For the computational complexity of the optimal mechanism design problem, we show that unless the polynomial-time hierarchy collapses (more exactly, $P^{\text{NP}} = P^{\#P}$), there is no universal efficient randomized algorithm to implement an optimal mechanism even when distributions have support size 3.

I. INTRODUCTION

Optimal pricing problems have been studied intensively during the past decade, under various settings and from both algorithmic and complexity-theoretic perspectives. They are closely related to problems that arise from the area of optimal Bayesian multidimensional mechanism design, e.g., see [Tha04], [HK05], [GHK⁺05], [BK06], [DHFS06], [BB06], [MV06], [CHK07], [BBM08], [Bri08], [Pav10], [BCKW10], [CMS10], [CD11], [HN12], [DW12], [HN13], [LY13], [DDT13], [WT14], [DDT14a], [BILW14], [CDP⁺14], [DDT14b], [Yao15]. The latter is well-understood under the single-parameter setting where Myerson's classic result shows that a simple, deterministic mechanism can achieve as much revenue as any sophisticated, randomized mechanism [Mye81]. The general case with multiple items, however, turns out to be more complex. Much effort has been devoted to understanding both the structure and complexity of optimal mechanisms, and to developing simple and computationally efficient mechanisms that are approximately optimal.

In this paper, we consider the following setting of monopolist lottery pricing where a buyer is interested in n heterogeneous items offered by a seller. We focus on the case when the buyer is *unit-demand* (i.e., only interested in obtaining at most one of the items) and *quasi-linear* (i.e., her utility is $v - p$ if she receives an item of value v to her and makes a payment of p to the seller). The seller is given full access to a probability distribution \mathcal{D} from which the buyer's valuations $\mathbf{v} = (v_1, \dots, v_n)$ for the items are drawn, and can exploit \mathcal{D} to choose a *menu* (a set) M of *lotteries* that maximizes her expected revenue (i.e., payment from the buyer). Here a lottery is of the form (\mathbf{x}, p) , where $p \in \mathbb{R}$ is its price and $\mathbf{x} = (x_1, \dots, x_n)$ is a non-negative vector that sums to at most 1, with each x_i being the probability of the buyer receiving item i if this lottery is purchased (the buyer receives no item with probability $1 - \sum_i x_i$). After a menu M is chosen, the buyer draws a valuation vector \mathbf{v} from \mathcal{D} and receives a lottery that maximizes her expected utility $\sum_i x_i \cdot v_i - p$, or the empty lottery $(\mathbf{0}, 0)$ by default if every lottery in M has a negative utility.

Given \mathcal{D} , its optimal menus are characterized by a linear program in which we associate with each \mathbf{v} in $\mathcal{D} := \text{supp}(\mathcal{D})$ a tuple of $n + 1$ variables to capture the lottery that the buyer receives at \mathbf{v} (see Section II). We will refer to it as the *standard linear program* for the optimal lottery problem. In particular, for the case when \mathcal{D} is correlated and given explicitly (i.e., given as a tuple of valuation vectors and their probabilities), one can find an optimal menu by solving the standard linear program in polynomial time [BCKW10].

We focus on the case when $\mathcal{D} = \mathcal{D}_1 \times \dots \times \mathcal{D}_n$ is a product distribution and each v_i is drawn independently from \mathcal{D}_i . The standard linear program in this case has exponentially many variables (even when each \mathcal{D}_i has support size 2), so one cannot afford to solve it directly. We are interested in the following two questions:

Menu size complexity: How many lotteries are needed to achieve the optimal revenue?

Computational complexity: How difficult is it to compute¹ an optimal menu of lotteries?

While much progress has been made when the buyer is additive (see discussions on related work in Section I-A), both questions remain widely open for the setting of a single unit-demand buyer with item values drawn from a product distribution (for correlated distributions see discussions later in Section I-A). For example, no explicit instance is known previously to require exponentially many lotteries to achieve the optimal revenue in this setting. (A trivial upper bound on the menu size is $|D|$ since otherwise at least one lottery in the menu is never used.)

Our first result is an explicit, simple product distribution \mathcal{D} , for which exponentially many lotteries are needed ($\Omega(|D|)$ indeed) to achieve the optimal revenue. Let \mathcal{D}' denote the distribution supported on $\{1, 2\}$, with probabilities $(1 - p, p)$, and let \mathcal{D}'' denote the distribution supported on $\{0, n + 2\}$, with probabilities $(1 - p, p)$, where $p = 1/n^2$. We prove the following theorem in Section IV.

Theorem I.1. *When n is sufficiently large, any optimal menu for*

$$\mathcal{D}^* = \mathcal{D}' \times \mathcal{D}' \times \cdots \times \mathcal{D}' \times \mathcal{D}''$$

over n items must have $\Omega(2^n)$ many different lotteries.

Note that all distributions in \mathcal{D}^* are the same except one. We show that this is indeed necessary. Before stating our result, we review the optimal item pricing problem. The setup is the same, but now the seller can only assign a price $p_i \in \mathbb{R}$ to each item i . Once the prices are fixed, the buyer draws \mathbf{v} from \mathcal{D} and then buys an item i that maximizes her utility $v_i - p_i$ (or buys no item if $v_i - p_i$ is negative for all i). The problem is to find a tuple of prices that maximizes the seller's expected revenue. Equivalently, an item pricing is a menu in which each lottery is of the special form (\mathbf{e}_i, p) , for some unit vector \mathbf{e}_i (so the menu size is at most n). In general, lotteries can extract strictly higher revenue than the optimal item pricing, as shown in [Tha04], which motivated much of the subsequent work.

We show that lotteries do not help when \mathcal{D}_i 's have support size 2 and share the same high value.

Theorem I.2. *If $\mathcal{D} = \mathcal{D}_1 \times \cdots \times \mathcal{D}_n$ and $\text{supp}(\mathcal{D}_i) = \{a_i, b\}$ with $a_i < b$ for all $i \in [n]$, an optimal item pricing achieves the same expected revenue as that of an optimal menu of lotteries.*

Therefore, the exponential lower bound on the menu size in Theorem I.1 cannot hold for support-size-2 distributions that share the same high value. The proof of Theorem I.2 also implies that an optimal menu in this case can be computed in polynomial time. For the special case of two items, we show that the condition of \mathcal{D}_1 and \mathcal{D}_2 sharing the same high value can be dropped.

Theorem I.3. *If both \mathcal{D}_1 and \mathcal{D}_2 have support size 2, then an optimal item pricing for $\mathcal{D}_1 \times \mathcal{D}_2$ achieves the same expected revenue as that of an optimal menu of lotteries.*

We also give three-item support-size-2 and two-item support-size-3 instances where lotteries help.

Next we describe our result on the problem of *computing* an optimal menu of lotteries. Although our distribution \mathcal{D}^* in Theorem I.1 trivially rules out any polynomial-time algorithm that lists explicitly all lotteries in an optimal menu, there is indeed a deterministic polynomial-time algorithm that, given any $\mathbf{v} \in D$, outputs a lottery $\ell_{\mathbf{v}}$ such that $\{\ell_{\mathbf{v}} : \mathbf{v} \in D\}$ is an optimal menu for \mathcal{D}^* . We are interested in the question of whether a *universal* efficient algorithm that computes an optimal menu in this fashion exists for product distributions.

This question is motivated by a folklore connection between the optimal lottery problem and the optimal mechanism design problem. Consider the same setting, where a unit-demand buyer with values drawn from \mathcal{D} is interested in n items offered by a seller. Here a *mechanism* is a (possibly randomized) map \mathcal{B} from the set D to $([n] \cup \{\text{nil}\}) \times \mathbb{R}$, where $\mathcal{B}(\mathbf{v}) = (b, p)$ means that the buyer is assigned the item b (or no item if $b = \text{nil}$) and pays p to the seller. The optimal mechanism design problem is then to find an *individually rational* and *truthful* mechanism \mathcal{B} (see definitions in Section II) that maximizes the expected revenue of the seller.

Let $\bar{\mathcal{B}}(\mathbf{v}) = (\mathbf{x}(\mathbf{v}), \bar{p}(\mathbf{v}))$ denote the expected outcome of \mathcal{B} on \mathbf{v} , i.e., $x_i(\mathbf{v})$ is the probability of $\mathcal{B}(\mathbf{v})$ assigning item i and $\bar{p}(\mathbf{v})$ is the expected payment. It follows trivially from definitions of the two problems that, under the same distribution \mathcal{D} , \mathcal{B} is an optimal mechanism iff $(\bar{\mathcal{B}}(\mathbf{v}) : \mathbf{v} \in D)$ is an optimal menu. Therefore, the standard linear program for the lottery problem also characterizes optimal mechanisms.

¹See Theorem I.4 for the exact meaning of “computing” an optimal menu here.

By exploring further ideas behind the proof of Theorem I.1, we show that there exists no efficient universal algorithm to implement an optimal mechanism even when \mathcal{D}_i 's have support size 3, unless $P^{NP} = P^{\#P}$:

Theorem I.4. *Unless $P^{NP} = P^{\#P}$, there exists no algorithm $\mathcal{A}(\cdot, \cdot)$ with the following two properties:*

- 1) \mathcal{A} is a randomized polynomial-time algorithm that always terminates in polynomial time.²
- 2) Given any instance $I = (n, \mathcal{D}_1, \dots, \mathcal{D}_n)$ to the optimal mechanism design problem and any

$$\mathbf{v} \in \text{supp}(\mathcal{D}_1) \times \dots \times \text{supp}(\mathcal{D}_n),$$

where each \mathcal{D}_i has support size 3, $\mathcal{A}(I, \mathbf{v})$ always outputs a pair in $([n] \cup \{\text{nil}\}) \times \mathbb{R}$, such that $\mathcal{B}_I : \mathbf{v} \mapsto \mathcal{A}(I, \mathbf{v})$ is an optimal mechanism for the instance I .

Remark I.5. *It is worth pointing out that our results in Theorem I.1 and Theorem I.4 hold also in the model where lotteries (mechanisms) are required to be “complete”, i.e. every lottery (\mathbf{x}, p) in the menu satisfies $\sum_{i \in [n]} x_i = 1$ (so the buyer always receives an item upon purchasing a lottery).*

A. Related Work

We briefly review related work in the language of the optimal mechanism design problem.

For the unit-demand single-buyer setting considered here, Thanassoulis [Tha04] showed that, unlike the single-parameter setting where the optimal mechanism is deterministic [Mye81], an optimal mechanism for two items drawn independently and uniformly from $[5, 6]$ must involve randomization. In [BCKW10], Briest et al. showed that when \mathcal{D} is correlated and is given explicitly, one can solve the standard linear program to compute an optimal mechanism in polynomial time. Moreover, they showed that the ratio of expected revenues obtained by an optimal randomized mechanism (or lottery pricing) and an optimal deterministic mechanism (or item pricing) can be unbounded in instances with four items. This was later shown to hold for two items by Hart and Nisan [HN13] (they also showed it for the setting of a single additive buyer). In contrast, for the case of product distributions, Chawla et al. showed in [CMS10] that the ratio is at most 4. In [DHN14], Dughmi et al. studied the sampling and representation complexity of mechanisms in a black box access model for the distribution \mathcal{D} , and showed that there is a correlated distribution for which any approximately revenue-maximizing mechanism requires exponential representation complexity. They also improved previous upper bounds on the menu size needed to extract at least $(1 - \epsilon)$ -fraction of the optimal revenue. When there are two items drawn independently from distributions that meet certain conditions, Pavlov [Pav10] characterized optimal mechanisms under both unit-demand and additive settings.

The problem of finding an optimal deterministic mechanism (or an optimal item pricing) in the unit-demand setting with a product distribution was shown to be NP-complete in [CDP⁺14], and this holds even when the item distributions have support size 3 or are identical. An optimal item pricing can be computed in polynomial time for support size 2.

For the additive single-buyer setting, Manelli and Vincent presented in [MV06] an example where randomization results in a strictly higher expected revenue. Much progress has been made on either characterizing optimal mechanisms or developing simple, computationally efficient mechanisms that are approximately optimal [HN12], [LY13], [DDT13], [BILW14], [WT14], [DDT14b], [Yao15]. Among these results, the two that are most relevant to ours are the one by Hart and Nisan [HN13], and the one by Daskalakis et al. [DDT14a]. In [HN13], Hart and Nisan introduced the notion of *menu size*. They showed that there exists a (continuous) correlated distribution for which no mechanism of finite menu size can achieve a positive fraction of the optimal revenue. In [DDT14a], Daskalakis et al. showed that there cannot be an efficient universal algorithm that implements an optimal mechanism for product distributions, even when all items have support size 2, unless $P^{\#P} \subseteq ZPP$. We compare our proof of Theorem I.4 and the proof of [DDT14a] in Section I-B.

B. Ideas Behind the Proofs

The main difficulty in proving both Theorem I.1 and Theorem I.4 is to understand and characterize optimal solutions to the standard linear program (denoted by $LP(I)$) for certain instances I . For Theorem I.1, we need to show that *every* optimal solution to $LP(I)$ with distribution \mathcal{D}^* must have exponentially many different lotteries; for Theorem I.4 we need to embed an instance of a $\#P$ -hard problem in I , and show that every optimal solution to $LP(I)$ helps us solve the $\#P$ -hard problem. However, characterizing optimal solutions to $LP(I)$ is very challenging due to its exponentially many variables and constraints which result in a highly complex geometric object for which our current understanding is still limited (e.g. compared to the literature on the additive setting). Compared to the optimal item pricing problem under the same unit-demand setting, where NP-completeness was recently established in [CDP⁺14], there is a significant difference in their underlying structures. The

²The algorithm is allowed to make random choices with arbitrary rational polynomial-time computable probabilities.

item pricing problem has a richer combinatorial flavor; characteristics of the lottery pricing problem are mostly “continuous”, as suggested by its linear program formulation.

The high-level approach behind proofs of Theorem I.1 and Theorem I.4 is similar to that of [DDT14a]. We simplify the problem by first relaxing the standard linear program $LP(I)$ to a smaller linear program $LP'(I)$ on the same set of variables $(u(\mathbf{v}), \mathbf{q}(\mathbf{v}) : \mathbf{v} \in D)$ but only subject to a subset of carefully picked constraints of $LP(I)$. (Here $\mathbf{q}(\mathbf{v})$ denotes a tuple of n variables with $q_i(\mathbf{v})$ being the probability of buyer receiving item i in the lottery for \mathbf{v} ; $u(\mathbf{v})$ is the utility of the buyer at \mathbf{v} to replace the role of price of the lottery: as it will become clear in Section II, this simplifies $LP(I)$.) Then we focus on a highly restricted family of instances I and characterize the set of optimal solutions to $LP'(I)$, taking advantages of the relaxed and simplified $LP'(I)$ as well as special structures of I . Finally we show that every optimal solution to $LP'(I)$ is also a feasible and thus, optimal solution to the original standard linear program $LP(I)$ as well, and always satisfies the desired properties (e.g., has exponentially many different lotteries, for the purpose of Theorem I.1, or can be used to solve the $\#P$ -hard instance embedded in it, for the purpose of Theorem I.4).

The similarity between our proof techniques and those of Daskalakis et al. [DDT14a], however, stops here due to a subtle but crucial difference between the two linear programs. In our standard $LP(I)$, the allocation variables $\mathbf{q}(\mathbf{v})$ must sum to at most 1 because the buyer is unit-demand. For the additive setting, on the other hand, there is no such constraint on the sum of $q_i(\mathbf{v})$ but the only constraint is that $q_i(\mathbf{v}) \in [0, 1]$ for all i . It turns out that this difference requires a completely different set of ideas and constructions to carry out the plan described above for the unit-demand setting, which we sketch below.

Recall the two distributions \mathcal{D}' and \mathcal{D}'' in the statement of Theorem I.1, supported on $\mathcal{D}' = \{1, 2\}$ and $\mathcal{D}'' = \{0, n+2\}$, respectively. Consider the i.i.d. instance I with n items drawn from \mathcal{D}' each. We make the following observation: an i.i.d. instance as I always has a “symmetric” optimal solution in which $q_i(\mathbf{v})$ only depends on the value of v_i and the number of 2’s in \mathbf{v} ,³ and such a solution tends to have many different lotteries. For example, if in such an optimal solution $q_i(\mathbf{v}) \neq q_j(\mathbf{v})$ when $v_i = 2, v_j = 1$ and \mathbf{v} has $(n/2)$ many 2’s, then all such exponentially many \mathbf{v} ’s would have distinct lotteries. Inspired by this, we analyze $LP(I)$ (by a careful relaxation) and obtain a complete characterization of its optimal solutions. Each of its optimal solution is (almost) uniquely determined by $\mathbf{q}(\mathbf{1})$ of the all-1 vector at the bottom. Moreover, our characterization shows that there are exponentially many different lotteries when $\mathbf{q}(\mathbf{1})$ has full support. However, $\mathbf{q}(\mathbf{1})$ does not necessarily have full support; indeed any $\mathbf{q}(\mathbf{1})$ that sums to 1 results in the same optimal revenue. In fact, by Theorem I.2 $LP(I)$ has an optimal item pricing, i.e., an optimal solution with only n lotteries (this solution is nonsymmetric).

Our next idea is to add another item with its value drawn from the second distribution \mathcal{D}'' (which breaks the symmetry of the overall instance) in order to enforce the full support of $\mathbf{q}(\mathbf{1})$ in every optimal solution to $LP(I')$, where I' denotes the new instance with $n+1$ items. To this end, we study a relaxation $LP'(I')$ of $LP(I')$ and obtain a complete characterization of its optimal solutions. We show that every optimal solution to $LP'(I')$ satisfies all the constraints of $LP(I')$ and thus, $LP'(I')$ and $LP(I')$ have the same set of optimal solutions. Furthermore, every optimal solution to $LP'(I')$ must have $\mathbf{q}(\mathbf{1})$ being the uniform distribution over the first n items, which in turn implies that almost all valuations buy a different lottery. This then finishes the proof of Theorem I.1.

The proof of Theorem I.4 is based on similar ideas but is much more delicate and involved. The goal here is to embed a subset-sum-type $\#P$ -hard problem in I . Let g_1, \dots, g_n denote the input integers of the $\#P$ -hard problem. Roughly speaking, we are given a subset H of $\{g_1, \dots, g_n\}$ of size $n/2$, and are asked to decide whether the sum $\sum_{i \in H} g_i$ is at least as large as the median of $\binom{n}{n/2}$ many such sums derived from all subsets of $\{g_1, \dots, g_n\}$ of size $n/2$ (note that the exact definition of the problem is more involved; see Section V-B for details).

We consider an instance I with $n+2$ items, where item i is supported on $\{a_i, \ell_i, h_i\}$ for each $i \in [n]$ with $a_i \approx 1, \ell_i < h_i$, and $\ell_i \approx h_i \approx 2$. The other two items $n+1$ and $n+2$ are supported on $\{0, s\}$ and $\{0, t\}$, respectively, for some s and t that satisfy $t \gg s \gg 1$. The probabilities of item i taking values a_i, ℓ_i and h_i are $1-p-r, p$ and r , respectively, for each $i \in [n]$; item $n+1$ takes value 0 with probability $1-\delta$, and s with probability δ ; item $n+2$ takes value 0 with probability $1-\delta^2$, and t with probability δ^2 . The parameters p, r and δ satisfy $1 \gg p \gg r \gg \delta$. For now, we do not pin down exact values of the parameters a_i, s, h_i, t , but only assume that they satisfy certain mild conditions; the rest of parameters ℓ_i, p, r, δ , on the other hand, are assigned specific values.

The first step of our proof is to characterize the set of optimal solutions of a carefully chosen relaxation $LP'(I)$ of $LP(I)$, assuming that parameters a_i, s, h_i, t satisfy the conditions specified. To this end we partition the set D of all valuation vectors into four types $T_1 \cup T_2 \cup T_3 \cup T_4$, where T_i denotes the set of type- i vectors: T_1 consists of vectors \mathbf{v} with $v_{n+1} = v_{n+2} = 0$, T_2 consists of vectors \mathbf{v} with $v_{n+1} = s$ and $v_{n+2} = 0$, T_3 consists of vectors \mathbf{v} with $v_{n+2} = t$ and $v_{n+1} = 0$, and T_4 consists

³Indeed any optimal solution can be “symmetrized” into such a solution without any loss in revenue; a general symmetrization procedure can be found in [DW12]. However, there may be an optimal nonsymmetric solution that is much more compact than its symmetrization. This is the case, for example, with this instance of n items drawn from \mathcal{D}' each, before we add the additional $(n+1)$ th item.

of vectors \mathbf{v} with $v_{n+1} = s$ and $v_{n+2} = t$. The bottom vectors of T_i 's play a crucial role in our characterization. Let

$$\mathbf{a} = (a_1, \dots, a_n, 0, 0), \quad \mathbf{c}_2 = (a_1, \dots, a_n, s, 0), \quad \mathbf{c}_3 = (a_1, \dots, a_n, 0, t) \quad \text{and} \quad \mathbf{c}_4 = (a_1, \dots, a_n, s, t).$$

We also let $\rho : T_2 \cup T_3 \cup T_4 \rightarrow T_1$ denote the map with $\rho(\mathbf{v}) = (v_1, \dots, v_n, 0, 0)$.

Then our characterization shows that any optimal solution of $\text{LP}'(I)$ is (almost) uniquely determined by $\mathbf{q}(\mathbf{a})$, $u(\mathbf{c}_2)$, $u(\mathbf{c}_3)$ and $u(\mathbf{c}_4)$. This is done by a sequence of lemmas, each imposing a condition called **CONDITION-TYPE- i** on type- i vectors in optimal solutions of $\text{LP}'(I)$. They are established in reverse order: We start by proving the condition on type-2 and 4 vectors first, followed by type-3 vectors, and finally type-1 vectors. The proof of **CONDITION-TYPE-1** is among the technically most challenging part of the paper. In particular, **CONDITION-TYPE- i** for $i = 2, 3, 4$ requires that

$$u(\mathbf{v}) = \max \{u(\rho(\mathbf{v})), u(\mathbf{c}_i)\}, \quad \text{for each } \mathbf{v} \in T_i. \quad (1)$$

Given the characterization, we start pinning down parameters a_i, s, h_i, t . By setting a_i and s carefully, we show that in any optimal solution to $\text{LP}'(I)$, the first n entries of $\mathbf{q}(\mathbf{a})$ sum to 1 and are almost uniform, i.e., $q_i(\mathbf{a}) \approx 1/n$ for $i \in [n]$. Next by setting h_i to encode the input integer g_i of the $\#P$ -hard problem, **CONDITION-TYPE-1** implies that utilities of type-1 vectors (more exactly, a carefully chose subset of type-1 vectors) encode the desired sums of $(n/2)$ -subsets of $\{g_1, \dots, g_n\}$, in every optimal solution to $\text{LP}(I)$. Finally, $u(\mathbf{c}_3)$ is tightly controlled by our choice of t , and we can set it to an appropriate value so that $u(\mathbf{c}_3)$ encodes exactly the median of sums obtained from all $(n/2)$ -subsets of $\{g_1, \dots, g_n\}$. Combining these with (1) we conclude that the $\#P$ -hard problem can be solved by simply comparing $u(\mathbf{c}_3)$ with $u(\mathbf{v})$, in any optimal solution of $\text{LP}'(I)$, at a specific type-3 vector $\mathbf{v} \in T_3$ such that $u(\rho(\mathbf{v}))$ encodes the sum $\sum_{i \in H} g_i$ for the input set H .

In the last step of the proof, we use our characterization to show that every optimal solution of the relaxation $\text{LP}'(I)$ must also be a feasible and thus, optimal solution to the standard linear program $\text{LP}(I)$. This finishes the proof of Theorem I.4.

For Theorems I.2 and I.3 our method for showing that randomization does not help in these settings, is done by identifying suitable convex combinations of the revenues of item pricings which upper bound the revenues of all lotteries. Note that this proof method is complete for this purpose by the properties of linear programming; the problem here is to show existence of suitable coefficients for the convex combinations.

Organization. We first give formal definitions of both problems and present the standard linear program $\text{LP}(I)$ in Section II. We then prove Theorem I.2 in Section III, Theorem I.1 in Section IV, and Theorem I.4 in Section V. Proofs of Theorem I.3 and lemmas used in the extended abstract can be found in the full paper.

II. PRELIMINARIES

Consider an instance $I = (n, \mathcal{D}_1, \dots, \mathcal{D}_n)$, where a seller offers n items, indexed by $[n] = \{1, \dots, n\}$, to a unit-demand buyer whose valuation (v_1, \dots, v_n) of items is drawn from n independent discrete distributions \mathcal{D}_i , $i \in [n]$. Each distribution \mathcal{D}_i is given explicitly in I , including both its support $D_i = \text{supp}(\mathcal{D}_i)$ and the probability of each value in D_i . Let

$$\mathcal{D} = \mathcal{D}_1 \times \dots \times \mathcal{D}_n \quad \text{and} \quad D = D_1 \times \dots \times D_n = \text{supp}(\mathcal{D}).$$

A solution to the optimal lottery problem is a menu (or a finite set) M of lotteries (\mathbf{x}, p) , where each lottery consists of a non-negative *allocation vector* $\mathbf{x} = (x_1, \dots, x_n)$ that satisfies $\sum_i x_i \leq 1$ and a *price* $p \in \mathbb{R}$. Since x_i is the probability of the buyer receiving item i , with valuation $\mathbf{v} = (v_1, \dots, v_n)$, the expected utility of the buyer receiving a lottery (\mathbf{x}, p) is

$$\sum_{i \in [n]} x_i \cdot v_i - p.$$

(Note that the allocation vector \mathbf{x} does not necessarily sum to 1 and the buyer receives no item with probability $1 - \sum_{i \in [n]} x_i$; we refer to a lottery (\mathbf{x}, p) as a *complete* lottery if \mathbf{x} sums to 1.)

Given a menu M of lotteries, the buyer draws her valuation vector \mathbf{v} of items from \mathcal{D} , and then receives a lottery (\mathbf{x}, p) that maximizes her expected utility with respect to \mathbf{v} . (So when there is a tie, the seller can assign the buyer, among those that maximize the buyer's utility, a lottery with the maximum price.⁴) If all lotteries in M have a negative utility, the buyer receives the empty lottery $(\mathbf{0}, 0)$ by default; this corresponds to the buyer choosing to buy nothing. For each $\mathbf{v} \in D$, let

$$\Pr[\mathbf{v}] = \prod_{i \in [n]} \Pr_{\mathcal{D}_i}[v_i]$$

⁴As in the case of deterministic pricing [CDP⁺14], the supremum achievable revenue is independent of the tie-breaking rule. Furthermore, the maximum price (equivalently, maximum expected value) tie-breaking rule has the property that the supremum can be achieved.

denote the probability of $\mathbf{v} \sim \mathcal{D}$. Let $\mathbf{q}(\mathbf{v}) = (q_1(\mathbf{v}), \dots, q_n(\mathbf{v}))$ denote the allocation vector, and let $p(\mathbf{v})$ denote the price of the lottery that the buyer receives at \mathbf{v} . Then the menu is given by

$$M = \{(\mathbf{q}(\mathbf{v}), p(\mathbf{v})) : \mathbf{v} \in D\}.$$

The goal of the optimal lottery problem is to find an M that maximizes the expected revenue of the seller:

$$\text{REV}(M) = \sum_{\mathbf{v} \in D} \Pr[\mathbf{v}] \cdot p(\mathbf{v}).$$

The only conditions are to make sure that 1) the utility

$$u(\mathbf{v}) = \sum_{i \in [n]} v_i \cdot q_i(\mathbf{v}) - p(\mathbf{v})$$

of the buyer is always nonnegative and 2) $(\mathbf{q}(\mathbf{v}), p(\mathbf{v}))$ is indeed a lottery in M that maximizes the utility $u(\mathbf{v})$ of the buyer. We can therefore express the optimal lottery problem as a linear program over variables $(q(\mathbf{v}), p(\mathbf{v}) : \mathbf{v} \in D)$, or alternatively we can replace the price variables by the utility variables $u(\mathbf{v})$, using

$$p(\mathbf{v}) = \sum_{i \in [n]} v_i \cdot q_i(\mathbf{v}) - u(\mathbf{v}).$$

This gives us the following linear program $\text{LP}(I)$:

$$\text{Maximize } \sum_{\mathbf{v} \in D} \Pr[\mathbf{v}] \cdot \left(\sum_{i \in [n]} v_i \cdot q_i(\mathbf{v}) - u(\mathbf{v}) \right) \text{ subject to} \quad (2)$$

$$u(\mathbf{v}) \geq 0, \quad q_i(\mathbf{v}) \geq 0, \quad \text{and} \quad \sum_{i \in [n]} q_i(\mathbf{v}) \leq 1, \quad \text{for all } \mathbf{v} \in D \text{ and } i \in [n].$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \sum_{i \in [n]} (v_i - w_i) \cdot q_i(\mathbf{v}), \quad \text{for all } \mathbf{v}, \mathbf{w} \in D. \quad (3)$$

We refer to it as the standard linear program and denote it by $\text{LP}(I)$. Given any optimal solution $(u(\mathbf{v}), \mathbf{q}(\mathbf{v}) : \mathbf{v} \in D)$ to $\text{LP}(I)$, we refer to the number of lotteries in the menu it induces as its *menu size*.

For the optimal mechanism design problem (with a single unit-demand buyer), the setting is exactly the same (and so are the instances I). A randomized mechanism is a randomized algorithm \mathcal{B} that, given any $\mathbf{v} \in D$, returns a pair (a, p) , where $a \in [n] \cup \{\text{nil}\}$ is the item assigned to the buyer (or no item is assigned to the buyer, if $a = \text{nil}$) and $p \in \mathbb{R}$ is the payment from the buyer. Given \mathcal{B} , let $\bar{\mathcal{B}}(\mathbf{v}) = (\mathbf{x}(\mathbf{v}), \bar{p}(\mathbf{v}))$ denote the expected outcome of \mathcal{B} on \mathbf{v} , where $x_i(\mathbf{v})$ is the probability that $\mathcal{B}(\mathbf{v})$ assigns item i and $\bar{p}(\mathbf{v})$ is the expected payment.

We say \mathcal{B} is *individually rational* if the buyer always has a nonnegative utility if she reports truthfully:

$$\sum_{i \in [n]} v_i \cdot x_i(\mathbf{v}) - \bar{p}(\mathbf{v}) \geq 0, \quad \text{for all } \mathbf{v} \in D.$$

We say \mathcal{B} is *truthful* if the buyer has no incentive to misreport:

$$\sum_{i \in [n]} v_i \cdot x_i(\mathbf{v}) - \bar{p}(\mathbf{v}) \geq \sum_{i \in [n]} v_i \cdot x_i(\mathbf{w}) - \bar{p}(\mathbf{w}), \quad \text{for all } \mathbf{v}, \mathbf{w} \in D.$$

The goal of the problem is to find an individually rational and truthful mechanism that maximizes the expected revenue

$$\sum_{\mathbf{v} \in D} \Pr[\mathbf{v}] \cdot \bar{p}(\mathbf{v}).$$

From the definitions, \mathcal{B} is an optimal mechanism if and only if the set of lotteries

$$\{\bar{\mathcal{B}}(\mathbf{v}) : \mathbf{v} \in D\} = \{(\mathbf{x}(\mathbf{v}), \bar{p}(\mathbf{v})) : \mathbf{v} \in D\}$$

induced from \mathcal{B} is an optimal solution to the optimal lottery problem, that is, \mathcal{B} is an optimal mechanism if and only if the tuple $(u(\mathbf{v}), \mathbf{x}(\mathbf{v}) : \mathbf{v} \in D)$ it induces is an optimal solution to the standard linear program $\text{LP}(I)$, where we have similarly replaced $\bar{p}(\mathbf{v})$ with the utility $u(\mathbf{v})$ of the buyer.

III. DISTRIBUTIONS WITH SUPPORT $\{a_i, b\}$

In this section, we outline the proof of Theorem I.2. Suppose that the n items $i = 1, \dots, n$ have distributions with support $\{a_i, b\}$ of size 2, where $0 \leq a_i < b$, with the same high value b . Let q_i denote the probability that item i has value $v_i = a_i$ (and $1 - q_i$ that $v_i = b$). We show that lotteries do not offer any advantage over deterministic item pricing. A consequence is that in this case we can solve the optimal lottery problem in polynomial time.

Fix an optimal set of lotteries L^* . For each subset $S \subseteq [n]$ of items we let $\mathbf{v}(S)$ be the valuation in which items in S have value b and the rest have value a_i . Let L_S be the lottery of L^* that the buyer buys for $\mathbf{v}(S)$, and p_S be the price of L_S . Let

$$L_\emptyset = (x_1, \dots, x_n, p_\emptyset)$$

be the lottery for the valuation $\mathbf{v}(\emptyset) = (a_1, \dots, a_n)$. Note that $\sum_i x_i \leq 1$ and $p_\emptyset \leq \sum_i a_i x_i$. Let $x_0 = 1 - \sum_i x_i$.

Let R^* denote the expected revenue achieved by L^* . We show that R^* is bounded from above by a convex combination of expected revenues of a set of $n+1$ item pricings. Specifically, consider the following set of $n+1$ item pricings π_0, \dots, π_n : pricing π_0 assigns price b to all items; for each $i \in [n]$, pricing π_i assigns price a_i to item i and b to all other items. Let R_i be the expected revenue of item pricing π_i for each i . Then we show that

Lemma III.1. $R^* \leq \sum_{i=0}^n x_i R_i$.

Proof: Consider a valuation $\mathbf{v}(S)$ for a subset $S \neq \emptyset$. The utility of lottery L_\emptyset for $\mathbf{v}(S)$ is

$$\sum_{i \notin S} a_i x_i + b \sum_{i \in S} x_i - p_\emptyset \geq \sum_{i \notin S} a_i x_i + b \sum_{i \in S} x_i - \sum_{i \in [n]} a_i x_i = \sum_{i \in S} (b - a_i) x_i.$$

The utility of the lottery L_S that is bought under $\mathbf{v}(S)$ must be at least as large as that of L_\emptyset . The value of the lottery L_S is at most b , thus $b - p_S \geq \sum_{i \in S} (b - a_i) x_i$, hence

$$p_S \leq b - \sum_{i \in S} (b - a_i) x_i.$$

Therefore, the total optimal expected revenue R^* is (here we let $\Pr[S]$ denote the probability of $\mathbf{v}(S)$ for convenience):

$$\begin{aligned} R^* &= \sum_{\emptyset \neq S \subseteq [n]} p_S \cdot \Pr[S] + p_\emptyset \cdot \Pr[\emptyset] \\ &\leq \sum_{\emptyset \neq S \subseteq [n]} \left[b - \sum_{i \in S} (b - a_i) x_i \right] \Pr[S] + \sum_{i \in [n]} a_i x_i \cdot \Pr[\emptyset] \\ &= b(1 - \Pr[\emptyset]) - \sum_{i \in [n]} (b - a_i) x_i (1 - q_i) + \sum_{i \in [n]} a_i x_i \cdot \Pr[\emptyset]. \end{aligned}$$

Consider next the $n+1$ item pricings π_0, \dots, π_n . The expected revenue R_0 of π_0 is $b(1 - \Pr[\emptyset])$. Under the pricing π_i , for each $i \in [n]$, the revenue is b if $v_i = a_i$ and $v_j = b$ for some $j \neq i$, and is a_i in all other cases (i.e., if $v_i = b$, or if all $v_j = a_j$). So the expected revenue R_i of π_i is

$$R_i = b(q_i - \Pr[\emptyset]) + a_i(1 - q_i + \Pr[\emptyset]).$$

Consider the convex combination $\sum_{i=0}^n x_i R_i$ of the expected revenues of the $n+1$ pricings $\pi_i, i = 0, \dots, n$. We have:

$$\begin{aligned} \sum_{i=0}^n x_i R_i &= x_0 b(1 - \Pr[\emptyset]) + b \sum_{i \in [n]} x_i (q_i - \Pr[\emptyset]) + \sum_{i \in [n]} a_i x_i (1 - q_i + \Pr[\emptyset]) \\ &= b \sum_{i=0}^n x_i (1 - \Pr[\emptyset]) - b \sum_{i \in [n]} x_i (1 - q_i) + \sum_{i \in [n]} a_i x_i (1 - q_i) + \sum_{i \in [n]} a_i x_i \Pr[\emptyset] \\ &= b(1 - \Pr[\emptyset]) - \sum_{i \in [n]} (b - a_i) x_i (1 - q_i) + \sum_{i \in [n]} a_i x_i \Pr[\emptyset]. \end{aligned}$$

Thus, $R^* \leq \sum_{i=0}^n x_i R_i$. This finishes the proof of the lemma. \blacksquare

Therefore, for at least one of the pricings π_i , we must have $R^* \leq R_i$ (since $\sum_{i=0}^n x_i = 1$), and Theorem I.2 follows.

IV. A SUPPORT-2 INSTANCE WITH EXPONENTIALLY MANY LOTTERIES

In this section we prove Theorem I.1. Consider an instance I of the lottery problem with $n+1$ items, where each item has two possible values. The first n items are identical: item $i \in [n]$ is supported over $\{1, 2\}$ with probabilities $(1-p, p)$. The “special” item $n+1$ is supported over $\{0, s\}$, with probabilities $(1-p, p)$ as well, where

$$p = 1/n^2 \quad \text{and} \quad s = 2 + n = 2 + 1/(np).$$

For each subset $S \subseteq [n+1]$, we let $\mathbf{v}(S)$ denote the valuation vector where each item $i \in S$ has the high value (2 if $i \in [n]$, s if $i = n+1$), and each $i \notin S$ has the low value (1 or 0). Similarly, we let $\mathbf{q}(S)$ and $u(S)$, for each $S \subseteq [n+1]$, denote the allocation and utility variables of the standard linear program $\text{LP}(I)$. The revenue from a valuation $\mathbf{v}(S)$ is

$$\text{REV}(S) = \sum_{i \in [n+1]} v_i(S) \cdot q_i(S) - u(S),$$

and the objective is to maximize the total revenue,

$$\text{REV} = \sum_{S \subseteq [n+1]} \Pr[S] \cdot \text{REV}(S).$$

The standard linear program $\text{LP}(I)$ for the lottery problem on input I over allocation variables $q_i(S)$ and utility variables $u(S)$, $S \subseteq [n+1]$, can be written as follows:

$$\text{Maximize} \quad \sum_{S \subseteq [n+1]} \Pr[S] \cdot \left(\sum_{i \in [n+1]} v_i(S) \cdot q_i(S) - u(S) \right) \quad \text{subject to} \quad (4)$$

$$\begin{aligned} u(S) &\geq 0, \quad q_i(S) \geq 0, \quad \text{and} \quad \sum_{i \in [n+1]} q_i(S) \leq 1, & \text{for all } S \subseteq [n+1] \text{ and } i \in [n+1]. \\ u(S) - u(T) &\leq \sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S), & \text{for all } S, T \subseteq [n+1]. \end{aligned} \quad (5)$$

We partition the subsets of $[n+1]$ into two types: the set T_1 of *type-1* sets and the set T_2 of *type-2* sets, where

$$T_1 = \{S \subseteq [n+1] : n+1 \notin S\} \quad \text{and} \quad T_2 = \{S \subseteq [n+1] : n+1 \in S\}.$$

We consider the partial order among the subsets of $[n+1]$ and its Hasse diagram (transitive reduction) G .

We will define a relaxation $\text{LP}'(I)$ of the standard $\text{LP}(I)$, characterize its optimal solutions, and then show that an optimal solution to $\text{LP}'(I)$ must also be feasible and thus, optimal, for $\text{LP}(I)$. The relaxed $\text{LP}'(I)$ has the same set of variables and objective function as $\text{LP}(I)$, but only (some of the) constraints (5) between adjacent sets, and between type-1 sets and \emptyset :

$$\text{Maximize} \quad \sum_{S \subseteq [n+1]} \Pr[S] \cdot \left(\sum_{i \in [n+1]} v_i(S) \cdot q_i(S) - u(S) \right) \quad \text{subject to}$$

Part 1: $u(S) \geq 0$, $q_i(S) \geq 0$, and $\sum_{j \in [n+1]} q_j(S) \leq 1$, for all $S \subseteq [n+1]$ and $i \in [n+1]$.

Part 2: Constraints between (some of the) adjacent sets:

2a. $q_i(S) \geq u(S) - u(S \setminus \{i\})$, for all $S \subseteq [n+1]$, $i \in S$ and $i \neq n+1$;

2b. $u(S) \geq u(S \setminus \{i\})$, for all $S \subseteq [n+1]$ and $i \in S$.

Part 3: Constraints between type-1 sets and \emptyset :

$$u(S) \geq \sum_{i \in S} q_i(\emptyset), \quad \text{for all } S \in T_1.$$

Figure 1. Relaxed Linear Program $\text{LP}'(I)$ in the Proof of Theorem I.1

A. Characterization of Optimal Solutions to $LP'(I)$

We give a complete characterization of optimal solutions to $LP'(I)$ in three steps. First of all, we show how to determine the optimal allocations $\mathbf{q}(S)$, for all $S \neq \emptyset$, once the utilities $u(\cdot)$ are all set (Lemma IV.2). Then we show how to determine the optimal utilities $u(\cdot)$, given $\mathbf{q}(\emptyset)$ (Lemma IV.3). Finally, we show that every optimal solution to $LP'(I)$ must have $\mathbf{q}(\emptyset)$ being the uniform distribution over the first n items (Lemma IV.4). The characterization is summarized in Corollary IV.5.

For the first step, we note that variables of $\mathbf{q}(S)$ for each $S \neq \emptyset$ appear in $LP'(I)$ only in Parts 1 and 2a of the constraints. Given a utility function $u : 2^{[n+1]} \rightarrow \mathbb{R}_{\geq 0}$, the relaxed $LP'(I)$ with respect to the remaining variables $\mathbf{q}(S)$ decomposes into independent linear programs $LP(S : u)$ for the different subsets S :

$$\begin{aligned} & \text{Maximize } \sum_{j \in [n+1]} v_j(S) \cdot q_j - u(S) \quad \text{subject to} \\ & q_j \geq 0, \text{ for all } j \in [n+1]; \quad \sum_j q_j \leq 1; \quad q_i \geq u(S) - u(S \setminus \{i\}), \text{ for all } i \in S \cap [n]. \end{aligned}$$

This is summarized in the following lemma.

Lemma IV.1. *Let $(u(\cdot), \mathbf{q}(\cdot))$ be a solution to $LP'(I)$. Then for each $S \neq \emptyset$, $\mathbf{q}(S)$ satisfies all the constraints of $LP'(I)$ that involve $\mathbf{q}(S)$ if and only if it is a feasible solution to $LP(S : u)$. Furthermore, if $(u(\cdot), \mathbf{q}(\cdot))$ is an optimal solution to $LP'(I)$, then $\mathbf{q}(S)$ must be an optimal solution to $LP(S : u)$ for all $S \neq \emptyset$.*

The next lemma characterizes optimal solutions to the simple linear program $LP(S : u)$.

Lemma IV.2. *Suppose that $LP(S : u)$ is feasible for some $S \neq \emptyset$ and some utility function $u(\cdot)$. Then optimal solutions to $LP(S : u)$ are characterized as follows:*

- 1) *If $S \in T_1$, then a solution \mathbf{q} of $LP(S : u)$ is optimal if and only if $q_i = 0$ for all $i \notin S$, $q_i \geq u(S) - u(S \setminus \{i\})$ for all $i \in S$, and $\sum_{i \in S} q_i = 1$.*
- 2) *If $S \in T_2$, then a solution \mathbf{q} of $LP(S : u)$ is optimal if and only if $q_i = 0$ for all $i \notin S$, $q_i = u(S) - u(S \setminus \{i\})$ for all $i \in S \cap [n]$, and $q_{n+1} = 1 - \sum_{i \in S \cap [n]} q_i$.*

For the second step, the next lemma tells us how to determine from $\mathbf{q}(\emptyset)$, optimal values for all the utilities $u(S)$.

Lemma IV.3. *Any optimal solution of $LP'(I)$ must satisfy*

$$u(\{n+1\}) = \min_{i \in [n]} q_i(\emptyset) \quad \text{and} \quad u(S) = \sum_{i \in S \cap [n]} q_i(\emptyset), \quad \text{for every } S \neq \{n+1\}.$$

We sketch the main ideas of the proof. First note that any optimal solution satisfies

$$u(S) \geq \sum_{i \in S} q_i(\emptyset), \quad \text{for all } S \in T_1$$

and

$$u(S) \geq \max \left(\sum_{i \in S \cap [n]} q_i(\emptyset), u(\{n+1\}) \right), \quad \text{for all } S \in T_2.$$

Call a set S *tight*, if the corresponding inequality is tight. If there are non-tight sets, form another solution $(u'(\cdot), \mathbf{q}'(\cdot))$ by setting $u'(S) = u(S) - \epsilon$ if S is not tight for some small enough $\epsilon > 0$, $u'(S) = u(S)$ if S is tight, and setting $\mathbf{q}'(S)$ to be a vector of an optimal solution to $LP(S : u')$ for each $S \neq \emptyset$, while $\mathbf{q}'(\emptyset) = \mathbf{q}(\emptyset)$. We argue that the new solution is feasible and yields strictly more revenue. This is done by a careful charging argument showing that although some sets may yield a smaller revenue, other sets compensate for that, so that the net overall change is positive. This implies that all sets must be tight in any optimal solution.

We next show that if $u(\{n+1\}) \neq \min_{i \in [n]} q_i(\emptyset)$ then we can modify the solution by increasing or decreasing the utilities of $\{n+1\}$ as well as certain other sets by a small amount $\epsilon > 0$, to strictly increase the overall revenue; the relation between the probability p and s is important here to establish that $u(\{n+1\})$ should be neither greater nor smaller than $\min_{i \in [n]} q_i(\emptyset)$. Lemma IV.3 then follows.

From the above lemmas we know how to derive from $\mathbf{q}(\emptyset)$ all the utilities and all the $\mathbf{q}(S)$, $S \neq \emptyset$, in an optimal solution, so we can calculate the revenue as a function of $\mathbf{q}(\emptyset)$. We can then determine the optimal value of $\mathbf{q}(\emptyset)$ that maximizes the

revenue. Let $q_{\min} = \min_{i \in [n]} q_i(\emptyset)$. The revenue can be calculated to have the form

$$\text{REV} = c_0 + (c_1 - c_2) \sum_{i \in [n]} q_i(\emptyset) + c_3 \cdot q_{\min},$$

where c_0 is a constant, independent of the solution,

$$\begin{aligned} c_1 &= (1-p)^{n+1} - (1-p)p \approx 1, & c_2 &= (s-1)p^2 = \Theta(1/n^3) \ll c_1, & \text{and} \\ c_3 &= (s-2)np^2(1-p)^{n-1} - p(1-p)^n = p^2(1-p)^{n-1} > 0. \end{aligned}$$

Because $(c_1 - c_2) > 0$ and $c_3 > 0$, in order to maximize the revenue we want to maximize $\sum_{i \in [n]} q_i(\emptyset)$ and q_{\min} . Both these quantities are maximized simultaneously if we have $\sum_{i \in [n]} q_i(\emptyset) = 1$ and $q_{\min} = 1/n$. Thus, we have:

Lemma IV.4. *Any optimal solution to $\text{LP}'(I)$ satisfies*

$$q_{n+1}(\emptyset) = 0 \quad \text{and} \quad q_i(\emptyset) = 1/n, \quad \text{for all } i \in [n].$$

Combining the previous lemmas yields the following characterization of optimal solutions to $\text{LP}'(I)$.

Corollary IV.5. *Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ has the following form:*

- $\mathbf{q}(\emptyset) = (1/n, \dots, 1/n, 0)$.
- $u(S) = |S \cap [n]|/n$ for all $S \neq \{n+1\}$; $u(\{n+1\}) = 1/n$.
- For $S \in T_1 \setminus \{\emptyset\}$: $q_i(S) = 0$ for $i \notin S$; $q_i(S) \geq 1/n$ for $i \in S$; $\sum_{i \in S} q_i(S) = 1$.
- For $S \in T_2$ and $|S| \leq 2$: $\mathbf{q}(S) = (0, \dots, 0, 1)$.
- For $S \in T_2$ and $|S| > 2$: $q_i(S) = 0$ for $i \notin S$; $q_i(S) = 1/n$ for $i \in S \cap [n]$; $q_{n+1}(S) = 1 - |S \cap [n]|/n$.

B. Returning to the Standard Linear Program $\text{LP}(I)$

Now that we have characterized the set of optimal solutions of the relaxed $\text{LP}'(I)$ in Corollary IV.5, we can check that they are also feasible, and hence also optimal, in the full standard $\text{LP}(I)$.

Lemma IV.6. *Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ is a feasible (and optimal) solution to $\text{LP}(I)$.*

Proof: Consider any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ of $\text{LP}'(I)$. It suffices to show that (5) in $\text{LP}(I)$ holds for every pair of subsets $T, S \subseteq [n+1]$. The case when $S = \emptyset$ is easy to check, since we have $u(S) - u(T) = -u(T)$,

$$\sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S) = -\frac{|T \cap [n]|}{n},$$

and $u(T) \geq |T \cap [n]|/n$ for all T . Below we assume that $S \neq \emptyset$.

We claim that it suffices to prove (5) for $T, S \subseteq [n+1]$ that satisfy $T \subseteq S$ and $S \neq \emptyset$. To see this, consider any T and S with $S \neq \emptyset$, let $T' = T \cap S \subseteq S$, and suppose that T', S satisfy (5):

$$u(S) - u(T') \leq \sum_{i \in [n+1]} (v_i(S) - v_i(T')) \cdot q_i(S).$$

Note that $u(T) \geq u(T')$ by the monotonicity of u , hence $u(S) - u(T) \leq u(S) - u(T')$. Further, $v_i(T)$ and $v_i(T')$ differ only on elements $i \in T \setminus T' = T \setminus S$, but $q_i(S) = 0$ for all such i since $S \neq \emptyset$ by Corollary IV.5. Therefore, we have

$$\sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S) = \sum_{i \in [n+1]} (v_i(S) - v_i(T')) \cdot q_i(S),$$

and (5) holds for T, S as well.

Consider two sets $T \subset S \subseteq [n+1]$. If S is not one of the sets $\{n+1\}$ or $\{i, n+1\}, i \in [n]$, then the LHS of (5) is

$$u(S) - u(T) \leq \frac{|(S \setminus T) \cap [n]|}{n},$$

which is at most

$$\sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S)$$

because $q_i(S) \geq 1/n$ for all $i \in (S \setminus T) \cap [n]$. If S is $\{n+1\}$ or $\{i, n+1\}$, $i \in [n]$, then either $T = \emptyset$, in which case

$$u(S) - u(T) = \frac{1}{n} \leq \sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S) = s,$$

or $|T| = 1$, in which case $u(T) = 1/n$ and we have

$$u(S) - u(T) = 0 \leq \sum_{i \in [n+1]} (v_i(S) - v_i(T)) \cdot q_i(S).$$

Thus, (5) is satisfied in all cases. This finishes the proof of the lemma. \blacksquare

Finally, we show that any optimal solution to I requires an exponential number of lotteries.

Theorem IV.7. Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to I has $\Theta(2^n)$ different lotteries.

Proof: For all $S \subseteq [n+1]$, except for $\emptyset, [n+1]$ and $\{i, n+1\}$ for $i \in [n]$, the support of $\mathbf{q}(S)$ is equal to S , thus all these lotteries are different. Hence any optimal solution has $2^{n+1} - n - 2$ different lotteries. \blacksquare

V. HARDNESS OF OPTIMAL MECHANISM DESIGN

We prove Theorem I.4. This section is organized as follows. In Section V-A we characterize optimal solutions to a relaxation to the standard linear program $\text{LP}(I)$, denoted by $\text{LP}'(I)$, when parameters of I satisfy certain conditions. In Section V-B, we pin down all the parameters of I to embed in it a $\#P$ -hard problem called COMP. Finally we show in Section V-C that for such I , any optimal solution to $\text{LP}'(I)$ is optimal to $\text{LP}(I)$, and can be used to solve COMP. An efficient universal algorithm for the optimal mechanism design problem would then imply that $P^{\text{NP}} = P^{\#P}$.

A. Linear Program Relaxation

Let I denote an instance of $n+2$ items with the following properties. Each item $i \in [n]$ is supported over $D_i = \{a_i, \ell_i, h_i\}$ with $a_i < \ell_i < h_i$. Probabilities of a_i, ℓ_i and h_i are $1-p-r$, p and r , respectively, where

$$p = 1/2^{n^4} \quad \text{and} \quad r = p/2^{n^2}.$$

Let $\beta = 1/2^n$. The support $\{a_i, \ell_i, h_i\}$ of item $i \in [n]$ satisfies the following conditions:

$$\ell_i = 2 + 3(n-i)\beta, \quad \ell_i + \beta \leq h_i \leq \ell_i + (1 + (1/2^{2n}))\beta, \quad \text{and} \quad |a_i - 1| = O(np). \quad (6)$$

Let $d_i = \ell_i - a_i \approx 1$ and $\tau_i = h_i - \ell_i$. Promises on ℓ_i and h_i guarantee that $\tau_i \approx \beta$ and $\ell_i > h_{i+1} + \beta$ (or $\ell_i \approx h_{i+1} + 2\beta$ more exactly) for all i from 1 to $n-1$. Item $n+1$ takes value 0 with probability $1-\delta$, and s with probability δ ; item $n+2$ takes value 0 with probability $1-\delta^2$, and t with probability δ^2 . So let

$$D_{n+1} = \{0, s\}, \quad D_{n+2} = \{0, t\}, \quad \text{and} \quad D = D_1 \times \cdots \times D_{n+2}.$$

We impose the following conditions on δ, s and t :

$$\delta = \frac{1}{2^{n^6}}, \quad s = \Theta\left(\frac{1}{pn}\right), \quad t = O\left(\frac{\beta}{r^{m+1}m}\right), \quad t = \Omega\left(\frac{\beta}{r^{m+1}m2^n}\right), \quad \text{where } m = \lceil n/2 \rceil. \quad (7)$$

Note that $\delta \ll r \ll p$, and $t = 2^{\Theta(n^5)} \gg s = 2^{\Theta(n^4)} \gg 1$. Precise values of the a_i 's, h_i 's and s and t will be chosen later on in Section V-B, after we have analyzed the structure of the problem. In particular, the h_i 's and t will be used to reflect the instance of the $\#P$ -hard problem that we will embed in I and $\text{LP}(I)$.

We need some notation before describing the relaxation of $\text{LP}(I)$. Given $\mathbf{v} \in D$, we use $S(\mathbf{v})$ to denote the set of $i \in [n]$ such that $v_i \in \{\ell_i, h_i\}$, $S^-(\mathbf{v})$ to denote the set of $i \in [n]$ such that $v_i = \ell_i$, and $S^+(\mathbf{v})$ to denote the set of $i \in [n]$ such that $v_i = h_i$. So we always have $S(\mathbf{v}) = S^+(\mathbf{v}) \cup S^-(\mathbf{v}) \subseteq [n]$.

Next we partition D into T_1, T_2, T_3, T_4 , where T_1 consists of vectors with $v_{n+1} = v_{n+2} = 0$, T_2 consists of vectors with $v_{n+1} = s$ and $v_{n+2} = 0$, T_3 consists of vectors with $v_{n+2} = t$ and $v_{n+1} = 0$, and T_4 consists of vectors with $v_{n+1} = s$ and $v_{n+2} = t$. We call vectors in each T_i type- i vectors. We denote the bottom vector $(a_1, \dots, a_n, 0, 0)$ by \mathbf{a} , $(a_1, \dots, a_n, s, 0)$ by \mathbf{c}_2 , $(a_1, \dots, a_n, 0, t)$ by \mathbf{c}_3 , and (a_1, \dots, a_n, s, t) by \mathbf{c}_4 (so \mathbf{c}_i is the *bottom* of type- i vectors). Since $u(\mathbf{a}) = 0$ in any optimal solution to $\text{LP}(I)$, we fix it to be 0.

Given $\mathbf{v} \in D$, we write $\text{BLOCK}(\mathbf{v})$ to denote the set of $\mathbf{w} \in D$ with $S(\mathbf{w}) = S(\mathbf{v})$, $w_{n+1} = v_{n+1}$ and $w_{n+2} = v_{n+2}$; we refer to $\text{BLOCK}(\mathbf{v})$ as the *block* that contains \mathbf{v} . It would also be helpful to view each T_i as a collection of (disjoint) blocks. We say $\mathbf{v} \in D$ is *essential* if $S^+(\mathbf{v}) = \emptyset$ (here the intuition is that within each block, there is a unique essential vector with

the largest mass of probability, given $r \ll p$). We use D' to denote the set of essential vectors, and write $T'_i = T_i \cap D'$ and $T_i^* = T_i \setminus T'_i$ for each i . Given $\mathbf{v} \in D$, we use $\text{LOWER}(\mathbf{v})$ to denote the unique essential vector in $\text{BLOCK}(\mathbf{v})$.

We let $\min(S(\mathbf{v}))$ denote the smallest index in $S(\mathbf{v})$ and let

$$S'(\mathbf{v}) = S(\mathbf{v}) \setminus \{\min(S(\mathbf{v}))\}.$$

Given a vector $\mathbf{v} \in D$, we follow the convention and write $(\mathbf{v}_{-i}, \alpha)$ to denote the vector obtained from \mathbf{v} by replacing v_i with α . We let $\rho : T_2 \cup T_3 \cup T_4 \rightarrow T_1$ denote the map with $\rho(\mathbf{v}) = (v_1, \dots, v_n, 0, 0)$.

The linear program $\text{LP}'(I)$ is presented in Figure 2. It has the same objective function and variables $(u(\mathbf{v}), \mathbf{q}(\mathbf{v}) : \mathbf{v} \in D)$ as $\text{LP}(I)$. Part 1 of $\text{LP}'(I)$ is about type-1 vectors only. It includes constraints between essential type-1 vectors and \mathbf{a} in (8), between type-1 vectors in the same block in (9) and (10), and between type-1 vectors in neighboring blocks in (11) and (13). Each Part $i = 2, 3, 4$ concerns type- i vectors. It includes constraints between $\mathbf{v} \in T_i$ and its corresponding type-1 vector $\rho(\mathbf{v})$ and bottom vector \mathbf{c}_i in (14), (18) and (22), between vectors in the same block in (15), (19) and (23), and between vectors in neighboring blocks in (17), (21) and (25). Also Part 4 has two constraints on $u(\mathbf{c}_4)$ compared with $u(\mathbf{c}_3)$. It is easy to check that $\text{LP}'(I)$ is a relaxation of $\text{LP}(I)$.

In the rest of Section V-A of the full paper we prove a sequence of lemmas that completely characterize optimal solutions to $\text{LP}'(I)$. At the end we show that an optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ is essentially determined by $\mathbf{q}(\mathbf{a})$, the allocation vector of \mathbf{a} , and utilities of the other three bottom vectors: $u(\mathbf{c}_2)$, $u(\mathbf{c}_3)$ and $u(\mathbf{c}_4)$. To this end, we show that every optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ satisfies the following four conditions, one for each type of vectors:

CONDITION-TYPE-1: For each essential vector $\mathbf{v} \in T'_1$ and $\mathbf{v} \neq \mathbf{a}$, we have

$$u(\mathbf{v}) = \sum_{i \in S(\mathbf{v})} d_i \cdot q_i(\mathbf{a}).$$

For each $\mathbf{v} \in T'_1$ and $\mathbf{v} \neq \mathbf{a}$, letting $k = \min(S(\mathbf{v}))$ and $S'(\mathbf{v}) = S(\mathbf{v}) \setminus \{k\}$, we have $q_i(\mathbf{v}) = q_i(\mathbf{a})$ for every $i \in S'(\mathbf{v})$, and $q_k(\mathbf{v}) = 1 - \sum_{i \in S'(\mathbf{v})} q_i(\mathbf{a})$, and all other entries of $\mathbf{q}(\mathbf{v})$ are 0. Moreover, for each nonessential type-1 vector $\mathbf{v} \in T_1^*$, letting $\mathbf{w} = \text{LOWER}(\mathbf{v})$, we have $\mathbf{q}(\mathbf{v}) = \mathbf{q}(\mathbf{w})$ and

$$u(\mathbf{v}) = u(\mathbf{w}) + \sum_{j \in S^+(\mathbf{v})} \tau_j \cdot q_j(\mathbf{w}) = \sum_{i \in S(\mathbf{v})} d_i \cdot q_i(\mathbf{a}) + \sum_{j \in S^+(\mathbf{v})} \tau_j \cdot q_j(\mathbf{w}).$$

CONDITION-TYPE-2, 3, 4: Each type- i vector $\mathbf{v} \in T_i$, $i = 2, 3, 4$, satisfies

$$u(\mathbf{v}) = \max \{u(\rho(\mathbf{v})), u(\mathbf{c}_i)\}.$$

These conditions are established in reverse order: we start by proving the condition on type-2 and 4 vectors first, followed by type-3 and finally type-1 vectors. The proof of CONDITION-TYPE-1 is among the technically most difficult part of the paper. Inspired by these conditions, we introduce the following definition.

Let \mathbf{q} be a nonnegative $(n+2)$ -dimensional vector that sums to at most 1, and let $u_2, u_3, u_4 \geq 0$ with $u_3 \leq u_4 \leq u_3 + s$. We write $\text{Ext}(\mathbf{q}, u_2, u_3, u_4)$ to denote the following set of solutions to $\text{LP}'(I)$. First, we set $(u(\mathbf{v}), \mathbf{q}(\mathbf{v}) : \mathbf{v} \in T_1)$ to be the unique partial solution over type-1 vectors that satisfies both $\mathbf{q}(\mathbf{a}) = \mathbf{q}$ and CONDITION-TYPE-1. Next, $u(\cdot)$ is extended to D by setting $u(\mathbf{c}_i) = u_i$, $i = 2, 3, 4$, and applying CONDITION-TYPE-2,3,4. With $u(\cdot)$ fixed now we set $\mathbf{q}(\mathbf{v})$ of each vector $\mathbf{v} \in T_2 \cup T_3 \cup T_4$ separately to be an optimal solution to a small linear program that maximizes the revenue from \mathbf{v} subject to constraints that involve $\mathbf{q}(\mathbf{v})$ in $\text{LP}'(I)$ (similar to what we proved in Lemmas IV.1 and IV.2 on $\text{LP}(S : u)$ in Section IV).

Our characterization is summarized in the following theorem.

Theorem V.1. *Given any nonnegative vector \mathbf{q} that sums to at most 1, and $u_2, u_3, u_4 \geq 0$ that satisfies $u_3 \leq u_4 \leq u_3 + s$, $\text{Ext}(\mathbf{q}, u_2, u_3, u_4)$ is a nonempty set of feasible solutions to $\text{LP}'(I)$. Furthermore, any optimal solution $(u(\mathbf{v}), \mathbf{q}(\mathbf{v}) : \mathbf{v} \in D)$ to $\text{LP}'(I)$ belongs to $\text{Ext}(\mathbf{q}, u_2, u_3, u_4)$, where $\mathbf{q} = \mathbf{q}(\mathbf{a})$ and $u_i = u(\mathbf{c}_i)$, $i = 2, 3, 4$.*

B. Choices of Parameters and their Consequences

Now we pin down the rest of parameters: a_i, s, h_i, t , and see how they affect optimal solutions of $\text{LP}'(I)$.

Note that the type-1 part of any optimal solution to $\text{LP}'(I)$ is determined by $\mathbf{q}(\mathbf{a})$. We set a_i 's to be

$$a_i = \frac{1 + \sum_{k < i} (1 - p - r)^{k-1} \cdot (p + r)^2 \cdot \ell_k + (1 - p - r)^{i-1} \cdot (p + r) \cdot \ell_i}{(1 - p - r)^n + \sum_{k < i} (1 - p - r)^{k-1} \cdot (p + r)^2 + (1 - p - r)^{i-1} \cdot (p + r)}, \quad (26)$$

Maximize $\sum_{\mathbf{v} \in D} \Pr[\mathbf{v}] \cdot \left(\sum_{i \in [n+2]} v_i \cdot q_i(\mathbf{v}) - u(\mathbf{v}) \right)$ subject to the following constraints:

Part 0. Same constraints on $u(\mathbf{v})$ and $\mathbf{q}(\mathbf{v})$ as in $\text{LP}(I)$:

$$u(\mathbf{v}) \geq 0, \quad q_i(\mathbf{v}) \geq 0, \quad \text{and} \quad \sum_{j \in [n+2]} q_j(\mathbf{v}) \leq 1, \quad \text{for } \mathbf{v} \in D \text{ and } i \in [n+2].$$

Part 1. Constraints on type-1 vectors:

$$u(\mathbf{v}) \geq \sum_{i \in S(\mathbf{v})} d_i \cdot q_i(\mathbf{a}), \quad \text{for } \mathbf{v} \in T'_1; \quad (8)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \tau_i \cdot q_i(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_1, i \in S^+(\mathbf{v}) \text{ and } \mathbf{w} = (\mathbf{v}_{-i}, \ell_i); \quad (9)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \geq \sum_{j \in S^+(\mathbf{v})} \tau_j \cdot q_j(\mathbf{w}), \quad \text{for } \mathbf{v} \in T_1, \mathbf{w} = \text{LOWER}(\mathbf{v}); \quad (10)$$

$$u(\mathbf{v}) \geq u(\mathbf{w}), \quad \text{for } \mathbf{v} \in T_1, i \in S(\mathbf{v}), \mathbf{w} = \text{LOWER}(\mathbf{v}_{-i}, a_i); \quad (11)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \sum_{j \in [n]} (v_j - w_j) \cdot q_j(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_1, i \in S(\mathbf{v}), \mathbf{w} \in \text{BLOCK}(\mathbf{v}_{-i}, a_i). \quad (12)$$

Part 2. Constraints on type-2 vectors:

$$u(\mathbf{v}) \geq u(\rho(\mathbf{v})) \text{ and } u(\mathbf{v}) \geq u(\mathbf{c}_2), \quad \text{for } \mathbf{v} \in T_2; \quad (14)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \tau_i \cdot q_i(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_2, i \in S^+(\mathbf{v}), \mathbf{w} = (\mathbf{v}_{-i}, \ell_i); \quad (15)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \sum_{j \in [n]} (v_j - w_j) \cdot q_j(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_2, i \in S(\mathbf{v}), \mathbf{w} \in \text{BLOCK}(\mathbf{v}_{-i}, a_i). \quad (16)$$

Part 3: Constraints on type-3 vectors:

$$u(\mathbf{v}) \geq u(\rho(\mathbf{v})) \text{ and } u(\mathbf{v}) \geq u(\mathbf{c}_3), \quad \text{for } \mathbf{v} \in T_3; \quad (18)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \tau_i \cdot q_i(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_3, i \in S^+(\mathbf{v}), \mathbf{w} = (\mathbf{v}_{-i}, \ell_i); \quad (19)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \sum_{j \in [n]} (v_j - w_j) \cdot q_j(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_3, i \in S(\mathbf{v}), \mathbf{w} \in \text{BLOCK}(\mathbf{v}_{-i}, a_i). \quad (20)$$

Part 4: Constraints on type-4 vectors: $u(\mathbf{c}_4) \geq u(\mathbf{c}_3)$, $u(\mathbf{c}_4) - u(\mathbf{c}_3) \leq s \cdot q_{n+1}(\mathbf{c}_4)$, and

$$u(\mathbf{v}) \geq u(\rho(\mathbf{v})) \text{ and } u(\mathbf{v}) \geq u(\mathbf{c}_4), \quad \text{for } \mathbf{v} \in T_4; \quad (22)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \tau_i \cdot q_i(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_4, i \in S^+(\mathbf{v}), \mathbf{w} = (\mathbf{v}_{-i}, \ell_i); \quad (23)$$

$$u(\mathbf{v}) - u(\mathbf{w}) \leq \sum_{j \in [n]} (v_j - w_j) \cdot q_j(\mathbf{v}), \quad \text{for } \mathbf{v} \in T_4, i \in S(\mathbf{v}), \mathbf{w} \in \text{BLOCK}(\mathbf{v}_{-i}, a_i). \quad (24)$$

$$(25)$$

Figure 2. Relaxed Linear Program $\text{LP}'(I)$ in the Proof of Theorem I.4

which meets the promised condition (6) on a_i and makes the revenue from type-1 vectors a function of $\sum_i q_i(\mathbf{a})$. Calculation leads to the following closed form for the revenue obtained from type-1 vectors in solutions in $\text{Ext}(\mathbf{q}, u_2, u_3, u_4)$:

$$\text{a constant} + (1 - \delta)(1 - \delta^2) \cdot \sum_{i \in [n]} q_i(\mathbf{a}).$$

Our choices of a_i 's then make sure that $\sum_{i \in [n]} q_i(\mathbf{a}) = 1$ in any optimal solution to $\text{LP}'(I)$.

Lemma V.2. *Any optimal solution to $\text{LP}'(I)$ has $\sum_{i \in [n]} q_i(\mathbf{a}) = 1$. They share the same revenue from T_1 .*

Next we set s . The following choice meets our promise on s :

$$s = 2 + \frac{1}{(n - 0.5)p} = \Theta\left(\frac{1}{np}\right).$$

Given this, we show that $u(\mathbf{c}_2)$ and $\mathbf{q}(\mathbf{a})$ are uniquely determined in order to maximize the revenue from T_2 .

Lemma V.3. *Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ satisfies*

$$u(\mathbf{c}_2) = d_1 q_1(\mathbf{a}) = \dots = d_n q_n(\mathbf{a}).$$

Before giving our choices of h_i 's and t we introduce the decision problem COMP. An input (\mathcal{G}, H, M) of COMP consists of a tuple $\mathcal{G} = (g_2, \dots, g_n)$ of $n-1$ integers between 1 and $N = 2^n$, a subset $H \subset [2 : n]$ of size $|H| = m = \lceil n/2 \rceil$, and an integer M between 1 and $\binom{n-1}{m}$. Let $\text{Sum}(T) = \sum_{i \in T} g_i$ for $T \subseteq [2 : n]$. Let t^* be the M -th largest integer in the multiset

$$\{\text{Sum}(T) : T \subset [2 : n] \text{ and } |T| = m\}.$$

The problem is to decide whether $\text{Sum}(H) > t^*$ or $\text{Sum}(H) \leq t^*$. We show in the full paper that COMP is $\#P$ -hard.

We are ready to embed an instance (\mathcal{G}, H, M) of COMP in I . By our choices of a_i 's and ℓ_i 's we have $d_1 = \max_{j \in [n]} d_j$. We set $\tau_i = \tau'_i + \beta$ for each $i \in [n]$ with $\tau'_1 = \beta/N^2$ and

$$\tau'_i = \frac{\beta}{N^2} \cdot \frac{d_1 - d_i}{d_1} + g_i \cdot \frac{d_i \beta}{N^4} = O\left(\frac{n\beta^2}{N^2}\right), \quad \text{for each } i > 1,$$

which meet our promise on h_i 's. To see the connection between COMP and I , let R^* denote the set of $\mathbf{v} \in T_3$ with

$$|S(\mathbf{v})| = |S^+(\mathbf{v})| = m + 1 \quad \text{and} \quad 1 \in S^+(\mathbf{v}).$$

Fix any type-3 vector $\mathbf{v} \in R^*$, and let $\mathbf{w} = \rho(\mathbf{v}) \in T_1$. We take a closer look at $u(\mathbf{w})$. Let $(u(\cdot), \mathbf{q}(\cdot)) \in \text{Ext}(\mathbf{q}, u_2, u_3, u_4)$. Then by CONDITION-TYPE-1 and our choices of τ_i 's, we have

$$u(\mathbf{w}) = \sum_{i \in S(\mathbf{v})} d_i \cdot q_i + \tau_1 \cdot \left(1 - \sum_{i \in S'(\mathbf{v})} q_i\right) + \sum_{i \in S'(\mathbf{v})} \tau_i \cdot q_i = C + \frac{\beta u_2}{N^4} \sum_{i \in S'(\mathbf{v})} g_i,$$

where C is a number that does not depend on \mathbf{v} . This is how sums of subsets of $\{g_2, \dots, g_n\}$ are encoded. In particular, the M -th largest element of the multiset $\{u(\rho(\mathbf{v})) : \mathbf{v} \in R^*\}$, denoted by u^* , clearly corresponds to the M -th largest integer t^* in the multiset from COMP.

Let h denote the probability $\Pr[\mathbf{v}]$ of each vector $\mathbf{v} \in R^*$ (note that they share the same $\Pr[\mathbf{v}]$). We set

$$t = 2 + \frac{\beta \delta^2}{h(m+1)(M-0.5)},$$

which allows us to prove the last piece of the characterization:

Lemma V.4. *Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ must satisfy $u(\mathbf{c}_3) = u(\mathbf{c}_4) = u^*$.*

This lemma gives us the desired reduction from COMP to $\text{LP}'(I)$. Let \mathbf{v}_H denote the vector in R^* with

$$S^+(\mathbf{v}_H) = S(\mathbf{v}_H) = \{1\} \cup H.$$

Given that $u(\mathbf{c}_3) = u^*$, we have

- 1) if $\text{Sum}(H) > t^*$, then $u(\mathbf{v}_H) > u(\mathbf{c}_3)$;

2) if $\text{Sum}(H) \leq t^*$, then $u(\mathbf{v}_H) = u(\mathbf{c}_3)$, in any optimal solution to $\text{LP}'(I)$.

Since $\mathbf{q}(\mathbf{v}_H)$ maximizes the revenue from \mathbf{v}_H subject to constraints in Part 3 of $\text{LP}'(I)$ that involve $\mathbf{q}(\mathbf{v}_H)$, we have

- 1) if $\text{Sum}(H) > t^*$, then $q_{n+2}(\mathbf{v}_H) < 1$;
- 2) if $\text{Sum}(H) \leq t^*$ then $q_{n+2}(\mathbf{v}_H) = 1$, in any optimal solution.

C. Returning to the Standard Linear Program

Let (\mathcal{G}, H, M) be an instance of COMP and let I be the instance constructed from (\mathcal{G}, H, M) . In the last step of the proof we show in the full paper that any optimal solution to $\text{LP}'(I)$ is a feasible solution to the standard linear program $\text{LP}(I)$.

Lemma V.5. *Any optimal solution $(u(\cdot), \mathbf{q}(\cdot))$ to $\text{LP}'(I)$ is a feasible solution to $\text{LP}(I)$.*

We combine all the lemmas to prove Theorem I.4.

Proof of Theorem I.4: Since $\text{LP}'(I)$ is a relaxation of $\text{LP}(I)$, a solution to $\text{LP}(I)$ is optimal if and only if it is optimal to $\text{LP}'(I)$. Thus, our characterization of optimal solutions to $\text{LP}'(I)$ applies to optimal solutions to $\text{LP}(I)$ as well.

Suppose that $\mathcal{A}(\cdot, \cdot)$ satisfies both properties in Theorem I.4. By our construction of I and analysis of $\text{LP}'(I)$, we have

- 1) If $\text{Sum}(H) > t^*$, $\mathcal{A}(I, \mathbf{v}_H)$ assigns an item other than $n+2$ or no item to the buyer with a positive probability;
- 2) If $\text{Sum}(H) \leq t^*$, $\mathcal{A}(I, \mathbf{v}_H)$ always gives item $n+2$ with probability 1.

Given I and \mathbf{v}_H , deciding which case it belongs to NP as \mathcal{A} always terminates in polynomial time by assumption (this is the only place we need this assumption). The theorem follows from the #P-hardness of COMP. ■

ACKNOWLEDGMENT

Xi Chen, Anthi Orfanou, and Xiaorui Sun were partially supported by NSF grants CCF-1149257 and CCF-1423100. Ilias Diakonikolas was supported by EPSRC grant EP/L021749/1, a Marie Curie Career Integration Grant, and a SICSA grant. Dimitris Paparas and Mihalis Yannakakis were partially supported by NSF grant CCF-1320654.

REFERENCES

- [BB06] M.-F. Balcan and A. Blum, *Approximation algorithms and online mechanisms for item pricing*, Proceedings of the 7th ACM Conference on Electronic Commerce, 2006, pp. 29–35.
- [BBM08] M.-F. Balcan, A. Blum, and Y. Mansour, *Item pricing for revenue maximization*, Proceedings of the 7th ACM Conference on Electronic Commerce, 2008, pp. 50–59.
- [BCKW10] P. Briest, S. Chawla, R. Kleinberg, and S.M. Weinberg, *Pricing randomized allocations*, Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, 2010, pp. 585–597.
- [BILW14] M. Babaioff, N. Immorlica, B. Lucier, and S.M. Weinberg, *A simple and approximately optimal mechanism for an additive buyer*, SIGecom Exchange **13** (2014), no. 2, 31–35.
- [BK06] P. Briest and P. Krysta, *Single-minded unlimited supply pricing on sparse instances*, Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm, 2006, pp. 1093–1102.
- [Bri08] P. Briest, *Uniform budgets and the envy-free pricing problem*, Proceedings of the 35th International Colloquium on Automata, Languages and Programming, 2008, pp. 808–819.
- [CD11] Y. Cai and C. Daskalakis, *Extreme-value theorems for optimal multidimensional pricing*, Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science, 2011, pp. 522–531.
- [CDP⁺14] X. Chen, I. Diakonikolas, D. Paparas, X. Sun, and M. Yannakakis, *The complexity of optimal multidimensional pricing*, Proceedings of the 25th ACM-SIAM Symposium on Discrete Algorithms, 2014.
- [CHK07] S. Chawla, J.D. Hartline, and R.D. Kleinberg, *Algorithmic pricing via virtual valuations*, Proceedings of the 8th ACM Conference on Electronic Commerce, 2007, pp. 243–251.
- [CMS10] S. Chawla, D.L. Malec, and B. Sivan, *The power of randomness in bayesian optimal mechanism design*, Proceedings of the 11th ACM Conference on Electronic Commerce, 2010, pp. 149–158.

- [DDT13] C. Daskalakis, A. Deckelbaum, and C. Tzamos, *Mechanism design via optimal transport*, Proceedings of the 14th ACM conference on Electronic commerce, 2013, pp. 269–286.
- [DDT14a] ———, *The complexity of optimal mechanism design*, Proceedings of the 25th ACM-SIAM Symposium on Discrete Algorithms, 2014, pp. 1302–1318.
- [DDT14b] ———, *Strong duality for a multiple-good monopolist*, CoRR abs/1409.4150 (2014).
- [DHFS06] E.D. Demaine, M.T. Hajiaghayi, U. Feige, and M.R. Salavatipour, *Combination can be hard: Approximability of the unique coverage problem*, Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm, 2006, pp. 162–171.
- [DHN14] S. Dughmi, L. Han, and N. Nisan, *Sampling and representation complexity of revenue maximization*, Proceedings of the 10th Conference on Web and Internet Economics, 2014.
- [DW12] C. Daskalakis and S.M. Weinberg, *Symmetries and optimal multi-dimensional mechanism design*, Proceedings of the 13th ACM Conference on Electronic Commerce, 2012, pp. 370–387.
- [GHK⁺05] V. Guruswami, J.D. Hartline, A.R. Karlin, D. Kempe, C. Kenyon, and F. McSherry, *On profit-maximizing envy-free pricing*, Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms, 2005, pp. 1164–1173.
- [HK05] J.D. Hartline and V. Koltun, *Near-optimal pricing in near-linear time*, Proceedings of the 9th International Workshop on Algorithms and Data Structures, 2005, pp. 422–431.
- [HN12] S. Hart and N. Nisan, *Approximate revenue maximization with multiple items*, Proceedings of the 13th ACM Conference on Electronic Commerce, 2012, pp. 656–656.
- [HN13] S. Hart and N. Nisan, *The menu-size complexity of auctions*, Proceedings of the 14th ACM Conference on Electronic Commerce, 2013, pp. 565–566.
- [LY13] X. Li and A.C.-C. Yao, *On revenue maximization for selling multiple independently distributed items*, Proceedings of the National Academy of Sciences **110** (2013), no. 28, 11232–11237.
- [MV06] A.M. Manelli and D.R. Vincent, *Bundling as an optimal selling mechanism for a multiple-good monopolist*, Journal of Economic Theory **127** (2006), no. 1, 1–35.
- [Mye81] R.B. Myerson, *Optimal auction design*, Mathematics of Operations Research **6** (1981), no. 1, 58–73.
- [Pav10] G. Pavlov, *Optimal Mechanism for Selling Two Goods*, UWO Department of Economics Working Papers 20103, University of Western Ontario, Department of Economics, 2010.
- [Tha04] J. Thanassoulis, *Haggling over substitutes*, Journal of Economic Theory **117** (2004), no. 2, 217–245.
- [WT14] Z. Wang and P. Tang, *Optimal mechanisms with simple menus*, Proceedings of the 15th ACM Conference on Economics and Computation, 2014, pp. 227–240.
- [Yao15] A.C.-C. Yao, *An n-to-1 bidder reduction for multi-item auctions and its applications*, Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms, 2015, pp. 92–109.