Games and Economic Behavior 110 (2018) 139–164

Contents lists available at ScienceDirect

Games and Economic Behavior

www.elsevier.com/locate/geb

The complexity of optimal multidimensional pricing for a unit-demand buyer [☆]

Xi Chen [a,1], Ilias Diakonikolas [b,2], Dimitris Paparas [c,*,3], Xiaorui Sun [d,4], Mihalis Yannakakis [a,5]

[a] Department of Computer Science, Columbia University, New York, NY 10027, United States of America
[b] Department of Computer Science, University of Southern California, Los Angeles, CA 90089, United States of America
[c] Department of Computer Science, University of Wisconsin-Madison, Madison, WI 53706, United States of America
[d] Simons Institute for the Theory of Computing, University of California at Berkeley, Berkeley, CA 94720, United States of America

ARTICLE INFO

Article history:
Received 4 April 2014
Available online 10 April 2018

JEL classification:
00-01
99-00

Keywords:
Deterministic mechanism design
NP-completeness
Complexity theory
Algorithms

ABSTRACT

We resolve the complexity of revenue-optimal deterministic auctions in the unit-demand single-buyer Bayesian setting, i.e., the optimal item pricing problem, when the buyer's values for the items are independent. We show that the problem of computing a revenue-optimal pricing can be solved in polynomial time for distributions of support size 2, and its decision version is NP-complete for distributions of support size 3. We also show that the problem remains NP-complete for the case of identical distributions.

1. Introduction

Consider the following natural pricing scenario: We have a set of n items for sale and a single *unit-demand* buyer, i.e., a consumer interested in obtaining at most one of the items. The goal of the seller is to set prices for the items in order to maximize her revenue by exploiting stochastic information about the buyer's preferences. More specifically, the seller is given access to a distribution $\mathcal{F}$ from which the buyer's valuations $\mathbf{v} = (v_1, \ldots, v_n)$ for the items are drawn, i.e., $\mathbf{v} \sim \mathcal{F}$, and wants to assign a price p_i to each item in order to maximize her expected revenue. The buyer's utility for item $i \in [n]$ is given by $v_i - p_i$ and she will select an item with the maximum nonnegative utility or nothing if no such item exists.

☆ An extended abstract of this work appeared in the Proceedings of the Twenty-Fifth Annual ACM–SIAM Symposium on Discrete Algorithms (SODA 2014), SIAM, Philadelphia, pp. 1319–1328.

* Corresponding author.
E-mail address: paparas@cs.wisc.edu (D. Paparas).

1 Research supported by NSF grants CCF-1149257, CCF-1423100, CCF-1703925, and a Alfred P. Sloan Foundation fellowship.
2 Work done while the author was at the University of Edinburgh, supported in part by a startup from the University of Edinburgh and a SICSA PECE grant.
3 Work done while the author was at Columbia University, supported by NSF grant CCF-1017955 and by an Alexander S. Onassis Foundation Fellowship.
4 Work done while the author was at Columbia University, supported by NSF grant CCF-1149257 and a Alfred P. Sloan Foundation fellowship.
5 Research supported by NSF grants CCF-1017955, CCF-1423100 and CCF-1703925.

https://doi.org/10.1016/j.geb.2018.03.016
0899-8256/

This problem is known as the *Bayesian Unit-demand Item-Pricing Problem* (Chawla et al., 2007) which we refer to as the *item-pricing problem* below, and has received considerable attention during the past decade (more discussion on previous work can be found in Section 1.2).

The item-pricing problem is known to have tight connections with the *optimal mechanism design problem*, a central question in mathematical economics (see Manelli and Vincent, 2007 and references therein). Finding an optimal item-pricing in our setting is equivalent to finding a revenue-optimal *deterministic* mechanism. A *randomized* mechanism, on the other hand, is more general and would allow the seller to offer *lotteries*[6] over items (Briest et al., 2010; Chawla et al., 2015). Even though randomized mechanisms in general can derive strictly more revenue (as observed in Manelli and Vincent, 2007; Thanassoulis, 2004), deterministic mechanisms (such as the item-pricings we study in this paper) are more natural and simple, and indeed they are more commonly used in practice. Optimal mechanism design is well-understood in single-parameter settings (such as the case of selling a single item to multiple buyers, including the special case of $n = 1$ in the model we study in this paper) for which Myerson (1981) obtained a closed-form characterization for the optimal mechanism; in particular, Myerson showed that in the single-parameter setting the optimal deterministic mechanism can achieve as much revenue as any randomized mechanism. The multi-parameter mechanism design problem (such as the case of selling multiple items to a single buyer studied here), however, turns out to be much more challenging.

In this paper we study the item-pricing problem with a single unit-demand buyer when $\mathcal{F} = \times_{i=1}^{n} \mathcal{F}_i$ is a *product distribution* (Chawla et al., 2007, 2010; Cai and Daskalakis, 2011), i.e., the valuations of the buyer for the n items are independent random variables. We further assume that the distributions $\mathcal{F}_i$, as the input of the problem, are *discrete* (i.e., the support of each $\mathcal{F}_i$ is a finite set) and *rational* (i.e., both values in the support of each $\mathcal{F}_i$ and their corresponding probabilities are all *rational* numbers encoded in binary). Thus the input size is the number of bits needed to represent $\mathcal{F}_i$'s. We use Item-Pricing-Opt to denote the optimization problem:

Given a product distribution, find a price vector that achieves the optimal expected revenue,

and use Item-Pricing-Decision to denote its decision version[7]:

Given a product distribution and a rational $t \geq 0$, decide if the optimal revenue is at least t.

See Section 2 for formal definitions. As is the case for most optimization problems, Item-Pricing-Opt is at least as hard as its decision version since, as we show, given any price vector, one can compute the expected revenue it achieves efficiently (see Lemma 3.1).

These computational problems exhibit very rich structures. Prior to our work, even the special case when the distributions $\mathcal{F}_i$ have support size 2 was not well understood: First note that the search space is apparently at least exponential, since the support size of $\mathcal{F}$ is 2^n. What makes things more challenging is that the optimal prices are not necessarily in the support of $\mathcal{F}$ (see Cai and Daskalakis, 2011 for a simple example with two items and distributions of support size 2). So, a priori, it was not even clear whether the optimal prices can be described with polynomially many[8] bits in the input size, whether the decision problem is in NP,[9] and whether the problems can even be solved in exponential time.

1.1. Our results

We take a principled complexity-theoretic look at the item-pricing problem with independent discrete distributions. We start by showing (Theorem 1) that the decision problem Item-Pricing-Decision is in NP (and as a corollary, the optimal prices can be described with polynomially many bits). As mentioned above, the membership proof is non-trivial because the optimal prices may not lie in the support of $\mathcal{F}$. Our proof proceeds by partitioning the space of price vectors into a set of (exponentially many) cells (defined using the input distributions $\mathcal{F}_i$), so that the optimal revenue within each cell can be computed efficiently by a shortest path computation. One consequence of the analysis is that Item-Pricing-Opt has the integrality property: if all values in the supports are integer then the optimal prices are also integer (though they may not belong to the support). Another consequence of the analysis is a simple algorithm which computes an optimal pricing by generating and evaluating a sufficient set of candidate price vectors which is guaranteed to contain an optimal price vector. The algorithm runs in polynomial space and exponential time; for a constant number of items, it runs in polynomial time. These results apply also to correlated distributions.

[6] A lottery in the setting of a single unit-demand buyer consists of a vector $(x_1, \ldots, x_n)$ and a price p, with $x_i \geq 0$ for all $i = 1, \ldots, n$, and $\sum_i x_i \leq 1$. If it is bought, the buyer pays p and receives an item i with probability x_i and nothing with probability $1 - \sum_i x_i$. The seller can offer a set (sometimes called a menu) of lotteries and the buyer buys one that maximizes her expected utility or nothing if every lottery in the menu has a negative utility.

[7] A decision problem is a problem that poses a "yes" or "no" question. Decision problems play a central role in computational complexity theory.

[8] This means that the number of bits is bounded from above by m^c for some constant c, where m is the input size.

[9] Informally, P is the set of all decision problems that can be solved in polynomial time and NP is the set of all decision problems for which solutions can be verified in polynomial time. A problem is NP-hard if it is at least as hard as every problem in NP (shown via polynomial-time reductions) and is NP-complete if it is both in NP and NP-hard. Many natural problems from a variety of scientific fields were shown to be NP-complete (Garey and Johnson, 1990). With NP vs P being the major open problem in theoretical computer science, NP-hard problems are generally perceived as computationally intractable. We refer interested readers to Papadimitriou (1994) for a more formal treatment of complexity theory.