On the complexity of a bundle pricing problem

Alexander Grigoriev[a,1], Joyce van Loon[a], Marc Uetz[b]

[a] *Maastricht University SBE, Quantitative Economics*
P.O.Box 616, 6200MD Maastricht, The Netherlands
[b] *University of Twente, Applied Mathematics,*
P.O. Box 217, 7500AE Enschede, The Netherlands

Abstract

We consider the problem of pricing items in order to maximize the revenue obtainable from a set of single minded customers. We relate the tractability of the problem to structural properties of customers' valuations: the problem admits an efficient approximation algorithm, parameterized along the inhomogeneity of the valuations.

Keywords: Pricing problems, bundle pricing problem, computational complexity, approximation algorithm

---

1. Introduction

Problem definition. Let $I = \{1, \ldots, m\}$ represent a set of items for sale and let $J = \{1, \ldots, n\}$ represent a set of potential customers. Every customer $j \in J$ requests a subset of items, denoted $I_j \subseteq I$. We refer to these subsets as *bundles*. Customers are single minded, which refers to the fact that they are interested in their particular bundle only. The *valuation* v_j for each bundle $I_j, j \in J$, is publicly known. This is reasonable when assuming customers' rationality and a competitive market environment: any customer can observe the publicly known prices for her bundle at all companies in the market, and then, behaving rationally, the customer defines her valuation being the cheapest market price for her bundle. We assume $v_j > 0,\ j \in J$,

Email addresses: a.grigoriev@maastrichtuniversity.nl (Alexander Grigoriev), j.vanloon@maastrichtuniversity.nl (Joyce van Loon), m.uetz@utwente.nl (Marc Uetz)

[1]Corresponding author: tel. +31-43-3883853, fax +31-43-3884874

for otherwise the customers having non-positive valuations can be deleted from the instance. We assume the items are available in unlimited supply, that is to say, we deal with digital goods or services. Let p_i be the price for item $i \in I$. We refer to the set $W = W(p) = \{j \in J \mid \sum_{i \in I_j} p_i \leq v_j\}$ as the set of *winners*. The bundle pricing problem asks for a vector of item prices $p = (p_1, \ldots, p_m)$ such that the total revenue $\Pi(p) = \sum_{j \in W(p)} \sum_{i \in I_j} p_i$ is maximal. Let us denote by Π the maximal revenue that can be extracted from the given set of customers.

Related work. The bundle pricing problem was introduced in combinatorial optimization literature by Guruswami et al. [5]. They show that the problem is APX-hard. Later, Demaine et al. [1] prove that the problem is inapproximable by a semi-logarithmic factor in the number of customers n. On the positive side, Guruswami et al. [5] present a polynomial time $O(\log n + \log m)$-approximation algorithm. Hartline and Koltun [6] design near-linear and near-cubic time approximation schemes under the assumption that the number of distinct items m is constant. Under the monotonicity condition that the total price of any bundle does not exceed the total price of any bigger bundle, Grigoriev et al. [3] show that the problem is still NP-hard but admits a polynomial time approximation scheme.

Our result. In this note we interpret customers' valuations in such a way that we come a step closer towards understanding the complexity of the problem. To start with, let us make the following definition.

Definition 1. *For any instance of the bundle pricing problem, define* $\bar{b}_j = v_j / |I_j|$ *as the average (per item) valuation of customer* j*, and define the inhomogeneity of valuations as*

$$\alpha = \max_{j,k \in J} \left\{ \bar{b}_j / \bar{b}_k \right\} .$$

Notice that $\alpha \geq 1$, and that the problem becomes trivial as soon as the valuations are homogeneous (that is, $\alpha = 1$ and $\bar{b}_j =: \bar{b}$ for all j). In this case, setting the price for each item $i \in I$ uniformly at $p_i = \bar{b}$, we obtain the optimal solution.

In contrast to the trivially solvable homogeneous case, the problem with inhomogeneity of valuations is NP-hard. While this does not sound very surprising, the main point is that the NP-hardness holds even if the inhomogeneity α is bounded from above by any constant $1 + \varepsilon$. In some sense, we thereby delineate the borderline between triviality and NP-hardness for the bundle pricing problem.

For the fact that the bundle pricing problem is NP-hard even for inhomogeneity arbitrarily close to 1, consider the NP-hardness reduction from Independent Set to the bundle pricing problem presented in Grigoriev et al. [3]. In this reduction, all average valuations of the bundles are at least M and at most $M+1$, where M is a chosen large number. The NP-hardness result for $\alpha \leq 1+\varepsilon, \varepsilon > 0$, follows straightforwardly. Moreover, the reduction works even under stronger restrictions on customers' valuations, namely *monotonicity*: $v_j \leq v_k$ for any $j,k \in J$ such that $I_j \subset I_k$. Thus, we proved the following theorem.

Theorem 1. *The bundle pricing problem is strongly NP-hard even if inhomogeneity $\alpha \leq 1+\varepsilon$ for any $\varepsilon > 0$, and if the valuations are monotone.*

In the next section we present a parametric approximation algorithm for the bundle pricing problem that complements the NP-hardness result. The proposed $\mathrm{O}(n(\log n + m))$-time algorithm has performance guarantee $1+\ln\alpha+\varepsilon$, for any $\varepsilon > 0$. Notice that this is a constant-factor approximation algorithm as soon as the inhomogeneity α of valuations is bounded by some constant, and the semi-logarithmic inapproximability result of Demaine et al. [1] is not longer valid. We believe that a constant bound on α is not unreasonable in practical applications.

2. $O(\ln\alpha)$-approximation algorithm

The idea of the approximation algorithm is as follows. We partition the set of customers J into $O(\ln\alpha)$ subsets $S_1,\ldots,S_K$, such that in each subset any two customers have average valuations different from each other by at most a constant factor $\delta > 1$. Denote by Π_k the maximum revenue for the bundle pricing problem restricted to the set of customers S_k (referred to as *S_k-restricted problem*). Then $\sum_{k=1}^{K}\Pi_k$ is clearly an upper bound for the optimum Π of the original problem. Therefore, the highest maximum revenue $\max_{k=1,\ldots,K}\Pi_k$ over all restricted problems is at least Π/K. Next, from the fact that the inhomogeneity of the average valuations in S_k is bounded by at most factor of δ, we derive that for the S_k-restricted problem there exists a price vector generating revenue at least Π_k/δ. Thus, taking the price vector yielding the highest revenue over all restricted problems, we generate a revenue at least $\Pi/\delta K$. Finally, we optimize the performance guarantee over parameters K and δ.

To partition the set of customers J into subsets $S_1, \ldots, S_K$, we use the following recursive procedure running in K steps. At step $k = 1, \ldots, K$, we construct subset S_k. Consider the set of customers J_k not yet assigned to any of the subsets $S_1, \ldots, S_{k-1}$, assuming $J_1 = J$. Add all customers $j \in J_k$ to S_k for which $\bar{b}_j \leq \delta^k \bar{b}_{\min}$, where $\bar{b}_{\min} = \min_{j \in J}\{\bar{b}_j\}$ and $\delta > 1$ to be defined later. Set $J_{k+1} = J_k \setminus S_k$ and recurse on this set.

By definition of the inhomogeneity α, we have $\bar{b}_k \leq \alpha \bar{b}_j$ for every pair of customers $k, j \in J$. Then, by straightforward induction on k, one can prove that the ratio between the highest and the lowest average valuations in J_k is at most α/δ^{k-1}, yielding $K \leq 1 + \log_\delta \alpha = 1 + \ln \alpha / \ln \delta$. Thus, we derived the first ingredient of the approximation algorithm, formulated in the following lemma.

Lemma 2. *For any $\delta > 1$, the number of subsets K is at most $1 + \ln \alpha / \ln \delta$.*

Second, we show that there is a solution to the S_k-restricted problem such that (i) the set of winners $W = S_k$; and (ii) the revenue generated in this solution is at least Π_k/δ. Consider the price vector $p^k = (p^k_1, \ldots, p^k_m)$ where price p^k_i of item $i \in I$ is determined as follows. Let $S_{ik} \subseteq S_k$ be the set of customers requesting item i. If $S_{ik} = \emptyset$, then price p^k_i can be chosen arbitrarily. If $S_{ik} \neq \emptyset$, define $p^k_i = \min\{\bar{b}_j |\ j \in S_{ik}\}$. Now, consider a customer $j \in S_k$. By definition of price vector p^k, the price of bundle I_j is $\sum_{i \in I_j} p^k_i \leq \sum_{i \in I_j} \bar{b}_j = v_j$, and therefore $j \in W$. By definition of set S_k, $\max_{j \in S_k} \bar{b}_j / \min_{j \in S_k} \bar{b}_j \leq \delta$, that yields a revenue at least Π_k/δ. Thus, we proved the following lemma.

Lemma 3. *In the S_k-restricted problem, price vector p^k yields a revenue at least Π_k/δ.*

Now, we are ready to present our first approximation result.

Theorem 4. *The bundle pricing problem admits an $e(1+\ln \alpha)$-approximation algorithm with computation time $O(n(\log n + m))$.*

Proof. The combination of Lemma 2 and Lemma 3 immediately implies that the revenue generated by the best price vector from $\{p^k |\ k = 1, \ldots, K\}$

is at least $\Pi/\delta(1+\frac{\ln\alpha}{\ln\delta})$, which is maximized for $\delta = e^{\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\ln\alpha}}\right)^{-1}}$. The claim follows from the fact that for big α the value of δ is close to e.

We arrive at the computation time as follows. First, in $O(n\log n)$ time we order the customers according to their average valuation (increasingly). Then, for all $k=1,\dots,K$, we use binary search to create set S_k in $O(\log n)$ time. For all items $i=1,\dots,m$, we determine the set of customers that request the item. This requires $O(nm)$ total time. So, the total computation time is $O(n\log n + K(\log n + nm))$, which is in $O(n(\log n + m))$, as K is a constant. □

3. Improved analysis

There are several directions for improvement of the obtained approximate solution to the bundle pricing problem. First, instead of the constructed price vectors $p^k, k=1,\dots,K$, we can use price vectors maximizing the revenue in the S_k-restricted problems, with given set of winners $W=S_k$. Notice that, for any set of winners $W\subseteq J$, the price vector maximizing the revenue obtained from W can be found in polynomial time by solving a simple linear program; see [2, 5]. Unfortunately, this approach does not necessarily lead to any provable improvement of the performance guarantee.

The following approach allows us to slightly improve the performance guarantee; it is simply based on a more careful analysis. By construction of the partition of J, for any two subsets S_k and $S_{k'}$, $k\le k'$, the average valuation of any customer from S_k is at most the average valuation of a customer from $S_{k'}$. Therefore, for any $k=1,\dots,K$, and for all $k'\ge k$, if $S_k\subseteq W$, then $S_{k'}\subseteq W$ as well. By definition of the subsets, the maximum average valuation in set S_{k+1} is at most δ times the maximum average valuation in set S_k. Thus, we have that the revenue generated by price vector p^k applied to the set of customers J is at least

$$R_k = \frac{1}{\delta}\Pi_k + \frac{1}{\delta^2}\Pi_{k+1} + \ldots + \frac{1}{\delta^{K-k+1}}\Pi_K, \qquad \forall k=1,\dots,K.$$

These equalities can be equivalently represented by the following recurrent formulas

$$R_k = \frac{1}{\delta}\Pi_k + \frac{1}{\delta}R_{k+1}, \qquad \forall k=1,\dots,K-1; \tag{1}$$

$$R_K = \frac{1}{\delta}\Pi_K. \tag{2}$$

Summing up all Equations (1) and (2) and dividing both sides by K, we derive

$$\bar{R} = \frac{1}{K}\sum_{k=1}^{K} R_k = \frac{1}{K\delta}\sum_{k=1}^{K} \Pi_k + \frac{1}{K\delta}\sum_{k=1}^{K} R_k - \frac{1}{K\delta}R_1.$$

Let $R_1 = \phi\bar{R}$. Since $\sum_{k=1}^{K} \Pi_k \geq \Pi$, we derive

$$\bar{R} \geq \frac{\Pi}{K(\delta-1)+\phi}.$$

Taking the maximum revenue over all price vectors $p^k, k = 1, \ldots, K$, we obtain

$$\max_{k=1,\ldots,K} R_k \geq \max\{R_1, \bar{R}\} \geq \max\left\{\frac{\phi\Pi}{K(\delta-1)+\phi}, \frac{\Pi}{K(\delta-1)+\phi}\right\},$$

that is minimized with $\phi = 1$, yielding

$$\max_{k=1,\ldots,K} R_k \geq \frac{\Pi}{\delta(1+\frac{\ln\alpha}{\ln\delta}) - \frac{\ln\alpha}{\ln\delta}}.$$

Note that $\delta(1+\frac{\ln\alpha}{\ln\delta}) - \frac{\ln\alpha}{\ln\delta} < \delta\ln\alpha + \delta$. Given $\varepsilon > 0$, let $\delta = 1+\varepsilon/(\ln\alpha+1)$. Then,

$$\delta\ln\alpha + \delta = \left(1+\frac{\varepsilon}{\ln\alpha+1}\right)\ln\alpha + \left(1+\frac{\varepsilon}{\ln\alpha+1}\right) = 1+\ln\alpha+\varepsilon,$$

and we arrive at the following theorem.

Theorem 5. *For any $\varepsilon > 0$, the bundle pricing problem admits an $(1+\ln\alpha+\varepsilon)$-approximation algorithm with computation time $O(n(\log n + m))$.*

Acknowledgements

Joyce van Loon acknowledges support by METEOR, the Maastricht Research School of Economics of Technology and Organizations. A preliminary version with parts of this work appeared in the conference proceedings [4].

References

[1] E.D. Demaine, U. Feige, M.T. Hajiaghayi, M.R. Salavatipour, *Combination Can Be Hard: Approximability of the Unique Coverage Problem*, SIAM J. Comput. 38 (2008) 1464-1483.

[2] A. Grigoriev, J. van Loon, R. Sitters, M. Uetz, *Optimal pricing of capacitated networks*, Networks 53 (2009) 79-87.

[3] A. Grigoriev, J. van Loon, M. Sviridenko, M. Uetz, T. Vredeveld, *Optimal bundle pricing with monotonicity constraint*, Operations Research Letters 36 (2008) 609-614.

[4] A. Grigoriev, J. van Loon, M. Uetz, *On the Complexity of the Highway Pricing Problem*, in: J. van Leeuwen, A. Muscholl, D. Peleg, J. Pokorný, B. Rumpe (Eds.), SOFSEM 2010: Theory and Practice of Computer Science, Lecture Notes in Computer Science 5901, Springer, Berlin/Heidelberg, 2010, pp. 465-476

[5] V. Guruswami, J.D. Hartline, A.R. Karlin, D. Kempe, C. Kenyon, F. McSherry, *On profit-maximizing envy-free pricing*, in: Proc. 16th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2005, ACM Press, USA, 2005, pp. 1164-1173.

[6] J.D. Hartline, V. Koltun, *Near-optimal pricing in near-linear time*, in: F.K.H.A. Dehne, A. Lpez-Ortiz, J.-R. Sack (Eds.), Algorithms and Data Structures - WADS 2005, Lecture Notes in Computer Sciences 3608, Springer, Berlin/Heidelberg, 2005, pp. 422-431.