

Online pricing for bundles of multiple items

Yong Zhang · Francis Y. L. Chin · Hing-Fung Ting

Received: 7 August 2012 / Accepted: 2 February 2013
© Springer Science+Business Media New York 2013

Abstract Given a seller with k types of items, m of each, a sequence of users $\{u_1, u_2, \dots\}$ arrive one by one. Each user is single-minded, i.e., each user is interested only in a particular bundle of items. The seller must set the price and assign some amount of bundles to each user upon his/her arrival. Bundles can be sold fractionally. Each u_i has his/her value function $v_i(\cdot)$ such that $v_i(x)$ is the highest unit price u_i is willing to pay for x bundles. The objective is to maximize the revenue of the seller by setting the price and amount of bundles for each user. In this paper, we first show that a lower bound of the competitive ratio for this problem is $\Omega(\log h + \log k)$, where h is the highest unit price to be paid among all users. We then give a deterministic online algorithm, Pricing, whose competitive ratio is $O(\sqrt{k} \cdot \log h \log k)$. When $k = 1$ the lower and upper bounds asymptotically match the optimal result $O(\log h)$.

Keywords Online algorithms · Pricing · Competitive ratio

1 Introduction

The problem of item pricing is one of the most important problems in computational economics. Item pricing contains two kinds of participators: the seller and the user. The seller has some items, which will be sold to the users at some designated prices; the user will buy the items at an acceptable price. The objective is to maximize the total revenue of the

Y. Zhang (✉) · F. Y. L. Chin · H.-F. Ting
Department of Computer Science, The University of Hong Kong, Pokfulam road, Hong Kong, Hong Kong
e-mail: yzhang@cs.hku.hk

F. Y. L. Chin
e-mail: chin@cs.hku.hk

H.-F. Ting
e-mail: hfting@cs.hku.hk

Y. Zhang
College of Mathematics and Computer Science, Hebei University, Hebei, China

seller by assigning items to the users. To achieve this target, the prices of the items must be sold dynamically, i.e., the prices of items are different for different users, at different times, with different amounts, . . . If the designated price is higher than the expected price of a user, this user will reject the assignment; otherwise, this user will accept the assignment.

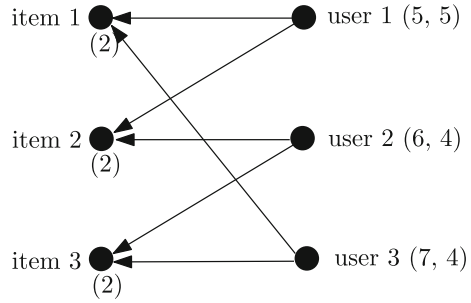
Pricing is a computationally interesting and realistic problem. Some researchers have studied more complex and realistic user preferences in the complete information setting, with the aim of devising fast algorithms to give the optimal, or a good approximate, pricing solution [5–9, 12–19]. Other researchers have studied the situation where the seller must set prices without knowledge, or complete knowledge, of the valuations of future users, with the aim of devising algorithms that will perform well even in the face of adversarial users trying to undermine the seller [1–4, 10, 11, 20].

The motivation of the problem studied in this paper is from some situations in economics. In investing in the stock market, suppose someone wants to sell all her shares to achieve the highest revenue. The price of the shares fluctuates with time, she only knows the prices of the history and has no idea of the future price. At each time, she must decide whether or not to sell some shares, and determines the number of shares to be sold. Another example, suppose a travel agent has some commodities, e.g., flights and hotels, to be sold to some users and the target is to maximize his revenue. Each user has her interested package of flights and hotels, and the valuations on different users are different. The travel agent must decide the price and amount of packages sold to each user on her arrival.

In this paper, we study the online pricing for bundles of multiple items. Formally speaking, given a seller with k types of items, i_1, i_2, \dots, i_k , the amount of each type is m . A sequence of users $\{u_1, u_2, \dots\}$ come one by one, each user is *single-minded*, i.e., each user is interested only in a particular bundle of items. For example, user u 's bundle of interest is $I_u = \{i_1, i_2\}$ (or $I_u = \{1, 2\}$). The bundles can be sold fractionally, but the amount of each item in the sold bundle must be the same. Still considering the above example, the seller may sell a half bundle to u , i.e., half i_1 and half i_2 . The seller must set the unit price and sell a certain number of bundles to each user on his/her arrival. In this paper, for ease of computation and comparison, the unit price is defined on items, not on bundles, even though one can convert to the other easily. For example, if the seller sells 1.5 bundles $I_u = \{1, 2\}$ to u at price 3, then the unit price is 1. Actually, if we define the unit price on a bundle, the results in this paper still hold, because the unit price on bundle I can be regarded as $|I|$ times the unit price on an item. However, defining the unit price on an item is more convenient since different bundles can be compared easily. Each u_i has his/her value function $v_i(\cdot)$ such that $v_i(x)$ is the highest unit price u_i is willing to pay for x bundles. Generally, the more bundles a user buys, the lower unit price he expects. Thus, in this paper, we assume that $v_i(x)$ is non-increasing. Let h be the highest value among all $v_i(x)$, i.e., $v_i(x) \leq h$ for all i and x . When user u comes with his bundle of interest I_u , suppose that the seller sets the item unit price p and assigns ℓ bundles to u . If $p > v_i(\ell)$, then user u cannot accept this price, and no bundle is bought by u . Otherwise, $p \leq v_i(\ell)$, and u accepts this price and pays $p \cdot \ell \cdot |I_u|$ to the seller.

To understand this model clearly, consider the example as shown in Fig. 1. The seller has $k = 3$ types of items, and each type contains $m = 2$ items. There are three single-minded users who want to buy these items. User 1's bundle of interest is $I_1 = \{1, 2\}$; the unit prices at which user 1 is willing to buy his bundle are 5 and 5 for buying 1 and 2 bundles respectively, i.e., $v_1(1) = v_1(2) = 5$. User 2's bundle of interest is $I_2 = \{2, 3\}$; the unit prices at which user 2 is willing to buy his bundle are 6 and 4 for buying one and two bundles respectively, i.e., $v_2(1) = 6$ and $v_2(2) = 4$. User 3's bundle of interest is $I_3 = \{1, 3\}$; the unit prices at which user 3 is willing to buy his bundle are 7 and 4 for buying one and two bundles respectively, i.e., $v_3(1) = 7$ and $v_3(2) = 4$. When user 1 comes, to maximize the seller's

Fig. 1 An example of online pricing for multiple items



revenue on this user, the seller will assign 2 bundles of $I_1 = \{1, 2\}$ at unit price 5 to him. When user 2 and user 3 come, there is no item 1 and item 2 left. In this case, user 2 and user 3 cannot buy anything and the total revenue of the seller is 20. However, the optimal strategy can achieve a total revenue of 36 by assigning one bundle of I_1 at unit price 5 to user 1, one bundle of $I_2 = \{2, 3\}$ at unit price 6 to user 2, and one bundle of $I_3 = \{1, 3\}$ at unit price 7 to user 3.

We consider the online version of this problem, i.e., before the i th user comes, the seller has no information of the j th user for $j \geq i$. To measure the performance of online algorithms, competitive analysis is generally used, i.e., to compare the outputs between the online algorithm and the optimal offline algorithm (which assumes all information is known in advance). Given the seller with item set B and a user sequence σ , let $A(B, \sigma)$ and $O(B, \sigma)$ denote the total revenue received by the seller according to the online algorithm A and the optimal offline algorithm O , respectively. The competitive ratio of the algorithm A is

$$R_A = \sup_{B, \sigma} \frac{O(B, \sigma)}{A(B, \sigma)}.$$

The pricing problem for items has been well studied in the past few years. Both multiple types of items and single type of items have been considered. Previous work has mainly focused on two supply models: the unlimited supply model [2, 5, 9, 15, 17] where the number of each type of item is unbounded and the limited supply model [2, 6–8, 12, 16, 18, 19] where the number of each type of item is bounded by some value. As for the users, there are several users’ behaviors studied, including single-minded [12–15, 17, 19] (each user is interested only in a particular set of items), unit-demand [5–9, 17, 19] (each user will buy at most one item in total) and envy free [2, 8, 12, 15, 17] (after the assignment, no user would prefer to be assigned a different set of items with the designated prices, loosely speaking, each user is happy with his/her purchase). Most of the previous studies have considered a combination of the above scenarios (e.g. envy-free pricing for single-minded users when there is unlimited supply). In Zhang et al. [20], considered a more practical and realistic model in the sense that the seller has a finite number of items (one type of item with limited supply) and users can demand more than one items and arrive online. They proved that the lower bound of the competitive ratio is $\Omega(\log h)$, moreover, they gave a deterministic online algorithm with competitive ratio $O(\log h)$, where h is the highest unit price.

In this paper, besides generalizing the problem to more than one type of items and bundles of items required by users, the idea used in our proposed online pricing algorithm is rooted by considering the amount of remaining items in addition to the user’s value function in determining the price and amount of items to be sold. The proof in establishing the upper bound is more complicated by employing a fine price partition. The result in this paper can

also match the optimal result $O(\log h)$ [20] asymptotically when there is only one type of item ($k = 1$) in the model.

This paper is organized as follows: Sect. 2 proves the lower bound of the competitive ratio for this variant to be $\Omega(\log h + \log k)$; in Sect. 3, a deterministic online algorithm whose competitive ratio is $O(\sqrt{k} \log h \log k)$ is given.

2 Lower bound of the competitive ratio

In this section, our target is to show that the lower bound of the competitive ratio of the online pricing problem for multiple items is $\Omega(\log h + \log k)$, where h is the highest unit price and k is the number of types.

For easy analysis of the lower bound, we assume that $h = 2^\ell$ and $k = 2^j$, i.e., $\log h$ and $\log k$ are both integers. The lower bound is proved step by step. In each step, the adversary sends a user with his value function to the seller. In this proof, all value functions are flat, particularly, the value function $v(x)$ is some power of 2 for all x .

In the first $\log k$ steps, the value function $v(x) = 1$.

Step 1:

The adversary sends user u_1 to the seller, with bundle of interest $\{1\}$.

If the seller assigns x_1 bundles to u_1 such that $x_1 \leq m/(\log h + \log k)$, the adversary stops. In this case, the revenue of the seller is at most $m/(\log h + \log k)$, while the maximal revenue is m by assigning all m bundles to u_1 . Thus, the ratio in this case is at least $\Omega(\log h + \log k)$.

Otherwise, the seller assigns more than $m/(\log h + \log k)$ bundles to u_1 . The adversary will send the next user to the seller.

Step 2:

The adversary sends user u_2 to the seller, with bundle of interest $\{1, 2\}$.

If the seller assigns x_2 bundles to u_2 such that $x_1 + x_2 \leq 2m/(\log h + \log k)$, the adversary stops. In this case, the revenue of the seller is at most $x_1 + 2x_2 \leq 3m/(\log h + \log k)$, while the optimal revenue is $2m$ by assigning all i_1 and i_2 to user u_2 . Thus, the ratio in this case is at least $\Omega(\log h + \log k)$.

Otherwise, the seller assigns x_2 bundles to u_2 such that $x_1 + x_2 > 2m/(\log h + \log k)$.

Step ℓ : ($1 < \ell \leq \log k$)

The adversary sends user u_ℓ to the seller, with bundle of interest $\{1, 2^{\ell-2} + 1, \dots, 2^{\ell-1}\}$.

If the seller assigns x_ℓ bundles to u_ℓ such that $\sum_{t=1}^\ell x_t \leq \ell \cdot m/(\log h + \log k)$, the adversary stops. In this case, the revenue achieved by the seller is

$$x_1 + 2x_2 + \dots + (2^{\ell-2} + 1)x_\ell \tag{1}$$

Lemma 1 *If the adversary stops at step ℓ , the total revenue is at most*

$$\frac{(2^{\ell-1} + \ell - 1)m}{\log h + \log k}.$$

Proof From previous steps, we have

$$\sum_{p=1}^t x_p \geq \frac{tm}{\log h + \log k} \quad (1 \leq t < \ell),$$

thus,

$$\sum_{p=t+1}^{\ell} x_p \leq \frac{(\ell - t)m}{\log h + \log k} \quad (1 \leq t < \ell).$$

Therefore, Eq. (1) achieves the maximal value $\frac{(2^{\ell-1} + \ell - 1)m}{\log h + \log k}$ when each x_p equals to $m/(\log h + \log k)$. □

The optimal revenue is $(2^{\ell-2} + 1)m$ by assigning all m bundles to u_{ℓ} . Therefore, in this case, the competitive ratio is still bounded by $\Omega(\log h + \log k)$.

Otherwise, the adversary sends the next user to the seller.

In the following $\log h$ steps ($\text{step } \log k + 1 \leq \ell \leq \log k + \log h$), the bundles of interest are $\{1, k/2 + 1 \dots k\}$, and the value functions are $v(x) = 2^{\ell - \log k}$ at step ℓ .

Step $\log k + 1$:

The adversary sends user $u_{\log k+1}$ to the seller.

If the seller assigns $x_{\log k+1}$ bundles to $u_{\log k+1}$ such that $\sum_{\ell=1}^{\log k+1} x_{\ell} \leq (\log k + 1)m/(\log h + \log k)$, the adversary stops. In this case, the revenue achieved by the seller is

$$x_1 + 2x_2 + \dots + (k/2 + 1)x_{\log k} + 2 \cdot (k/2 + 1)x_{\log k+1} \tag{2}$$

Similar to the proof in Lemma 1, we can find that the revenue achieved is at most

$$\frac{(3k/2 + \log k + 1)m}{\log k + \log h}.$$

The optimal revenue is $(k + 2)m$ by assigning all m bundles to $u_{\log k+1}$. Therefore, in this case, the competitive ratio is still bounded by $\Omega(\log h + \log k)$.

Otherwise, the adversary sends the next user to the seller.

The analysis from Step $\log k + 2$ until Step $\log k + \log h - 1$ are similar to the above one.

Step $\log k + \log h$:

The adversary sends user $u_{\log k + \log h}$ to the seller and the seller assigns $x_{\log k + \log h}$ bundles to the user. Since all bundles of interest include i_1 , thus, the total number of all assigned bundles is no more than m . Thus, the adversary must stop at this step. Similar to the previous analysis, the ratio between the optimal solution and the revenue achieved by the online algorithm is at least $\Omega(\log k + \log h)$.

Therefore, we have the following conclusion.

Theorem 2 *For the online pricing for multiple items, the lower bound of the competitive ratio is $\Omega(\log h + \log k)$.*

3 Online algorithm

To maximize the revenue of the seller on a particular user u with bundle of interest I , a straightforward idea is finding unit price p and amount of bundles b such that p is acceptable when buying b bundles, and $b \cdot p$ is maximized. If we assign b bundles with unit price p to u , the revenue is $b \cdot p \cdot |I|$.

In our algorithm, we assign unit price $2^\ell (\ell \geq 0)$ to each user. In this way, the performance will not be affected too much. Let (b, p) be the assignment such that $b \cdot p \cdot |I|$ is maximal. W.l.o.g., suppose $2^i \leq p < 2^{i+1}$, note that $v(x)$ is non-increasing, we have $v^{-1}(2^{i+1}) \leq b \leq v^{-1}(2^i)$, where $v^{-1}(\cdot)$ is the reverse function of $v(\cdot)$. Thus,

$$b \cdot p \leq v^{-1}(2^i) \cdot p \leq v^{-1}(2^i) \cdot 2^{i+1} = 2 \cdot v^{-1}(2^i) \cdot 2^i.$$

If we choose the unit price equals to some power of 2, $(v^{-1}(2^i), 2^i)$ is a candidate of the assignment, which is at least half of the maximal revenue on the user.

In our algorithm, we partition the amount of each type of item into $\lfloor \log h \rfloor$ stages, from stage 1 to stage $\lfloor \log h \rfloor$. The amount of items in stage i can be only assigned with unit price 2^{i-1} . Furthermore, partition the items in each stage into $\lfloor \log k \rfloor + 1$ levels, from level 0 to level $\lfloor \log k \rfloor$. For type i , items in level ℓ can be assigned only to users such that type i is in the user's bundle of interest and the size of the bundle is within $[2^{\ell-1} + 1, 2^\ell]$. For example, a user u 's bundle of interest is $\{1, 2, 3, 4\}$, thus, in our algorithm, we choose items from level 2 in some stage to satisfy this user.

Let $\delta_i^{s,t}$ denote the available amount of items of type i in stage s level t . Initially, $\delta_i^{s,t} = m / (\lfloor \log h \rfloor (\lfloor \log k \rfloor + 1))$.

Next, we will formally describe the pricing algorithm for multiple items.

Algorithm 1 Pricing

- 1: Let I be the bundle of interest of the coming user u .
 - 2: Let $\ell = \lceil \log |I| \rceil$ $\triangleright \ell$ denotes the level which may assign items to user i .
 - 3: Let x_j be the largest amount of bundles that user i is willing to buy given unit price 2^j and satisfying $x_j \leq m$.
 - 4: Let $y_j = \min\{x_j, \min_{i \in I} \{\delta_i^{j+1, \ell}\}\}$.
 - 5: Let $s = \arg \max_j y_j \cdot 2^j$ such that $y_j > 0$.
 - 6: **if** no such s exists **then**
 - 7: Assign 0 bundles to user u .
 - 8: **else**
 - 9: Set the unit price $p = 2^s$.
 - 10: Assign y_s bundles to user u .
 - 11: $\delta_i^{s+1, \ell} = \delta_i^{s+1, \ell} - y_s$ for all $i \in I$.
 - 12: **end if**
-

According to the algorithm Pricing, if a user u with bundle of interest I cannot be satisfied, that means in each stage s such that 2^s is an acceptable price for u , at least one type of item in I at level $\lceil \log |I| \rceil$ are all assigned to other users.

For a user sequence $\{u_1, u_2, \dots\}$, let ALG denote the total revenue received from the algorithm Pricing, let OPT be the revenue achieved by the optimal algorithm. Next, we give the competitive ratio of the algorithm Pricing, i.e., prove the upper bound of the ratio between OPT and ALG .

After the assignment on the whole user sequence $\{u_1, u_2, \dots\}$ by the algorithm Pricing, for each type of item, some levels in some stages are full, the others still contain some available items. Classify all levels into two classes: L_i^f denotes the levels of type i which are full, i.e., level t in stage s of type i belongs to L_i^f if $\delta_i^{s,t} = 0$; L_i^n denotes the levels of type i which contain available items, i.e., $\delta_i^{s,t} > 0$ for the corresponding levels.

Compare the assignments from the optimal algorithm and Pricing, we also partition the assignments from the optimal algorithm into two classes according to L_i^f and L_i^n . In the

optimal solution, consider the assignment to a user with bundle of interest I , suppose the unit price $p \in [2^\ell, 2^{\ell+1})$. In the assignment from Pricing, if there exist $i \in I$ such that $\delta_i^{\ell+1, \lceil \log |I| \rceil} = 0$, i.e., this level belongs to L_i^f , we say the revenue of this assignment from the optimal algorithm belongs to O_f . Otherwise, if for any $i \in I$, $\delta_i^{\ell+1, \lceil \log |I| \rceil} > 0$ in the assignment w.r.t. Pricing, the revenue of this assignment belongs to O_n .

Moreover, we partition the assignment according to the size of the bundle. If $|I| \leq \sqrt{k}$, we say the size of the bundle is *small*, otherwise, the size is *large*. The revenue of the assignment from the optimal algorithm is further partitioned into four classes: O_f^s, O_f^l, O_n^s , and O_n^l , where O_f^s and O_n^s denote the revenue from small bundles, O_f^l and O_n^l denote the revenue from large bundles. Note that $O_f^s + O_f^l = O_f$ and $O_n^s + O_n^l = O_n$. Next, we compare these four classes with *ALG* respectively.

Lemma 3 $\frac{O_f^s}{ALG} \leq O(\sqrt{k} \cdot \log h \log k)$.

Proof By the optimal algorithm, consider an assignment A on a bundle I with unit price $p \in [2^\ell, 2^{\ell+1})$. Assume that $|I| \leq \sqrt{k}$ and in the assignment from Pricing, at least one type $i \in I$ at level $\lceil \log |I| \rceil$ in stage $\ell + 1$ is full. The optimal revenue of the assignment A is

$$p \cdot m \cdot |I| \tag{3}$$

From the algorithm Pricing, the revenue at level $\lceil \log |I| \rceil$ in stage $\ell + 1$ of type i is

$$2^\ell \cdot m / (\lceil \log h \rceil (\lceil \log k \rceil + 1)) \tag{4}$$

The ratio between (3) and (4) is

$$O(\sqrt{k} \cdot \log h \log k).$$

Suppose by the optimal algorithm, there are more than one assigned small bundles, I_1, I_2, \dots, I_j with unit prices p_1, p_2, \dots, p_j satisfy $p_{j'} \in [2^\ell, 2^{\ell+1})$ for $1 \leq j' \leq j$ and all $\lceil \log |I_{j'}| \rceil$ ($1 \leq j' \leq j$) are equal. If these bundles share an item i such that in the assignment from Pricing, level $\lceil \log |I_1| \rceil$ in stage $\ell + 1$ of item i is full, the total revenue on these bundles in the optimal solution is at most

$$\max\{p_{j'}\} \cdot m \cdot \max\{|I_{j'}|\} \quad 1 \leq j' \leq j \tag{5}$$

Compare the value in Eq. (5) with the revenue from Pricing at level $\lceil \log |I| \rceil$ in stage $\ell + 1$ of type i (Eq. (4)), the ratio is also

$$O(\sqrt{k} \cdot \log h \log k).$$

From the optimal pricing, all assignments in O_f^s can be partitioned into parts, each part contains the assignments which satisfy the property mentioned above. Since the total revenues on such full levels is a lower bound of *ALG*, we have

$$\frac{O_f^s}{ALG} \leq O(\sqrt{k} \cdot \log h \log k).$$

□

Lemma 4 $\frac{O_f^l}{ALG} \leq O(\sqrt{k} \cdot \log h \log k)$.

Proof By the optimal algorithm, consider some amount of bundle I with unit price $p \in [2^\ell, 2^{\ell+1})$ is assigned to a user. Assume that $|I| > \sqrt{k}$ and there exist an item $i \in I$ such that in the assignment from Pricing, level $\lceil \log |I| \rceil$ in stage $\ell + 1$ of item i is full. Note that from the algorithm Pricing, items in this level can be only assigned to bundles with size in between $(2^{\lceil \log |I| \rceil - 1}, 2^{\lceil \log |I| \rceil}]$, thus, the total revenue on level $\lceil \log |I| \rceil$ in stage $\ell + 1$ is at least

$$\frac{2^\ell \cdot m \cdot 2^{\lceil \log |I| \rceil - 1}}{\lceil \log h \rceil (\lceil \log k \rceil + 1)} \tag{6}$$

Note that the optimal revenue for bundles with unit price $p \in [2^\ell, 2^{\ell+1})$ is at most

$$2^{\ell+1} \cdot m \cdot k \tag{7}$$

The ratio between the above two equations is $O(\sqrt{k} \cdot \log h \log k)$. Combine all revenues in O_f^l , this lemma is true. \square

Lemma 5 $\frac{O_n^s}{ALG} \leq O(\sqrt{k} \cdot \log h \log k)$.

Proof Again, consider an assigned bundle I from the optimal algorithm, such that the revenue on I belongs to O_n^s and the unit price is $p \in [2^\ell, 2^{\ell+1})$. The algorithm Pricing chooses the unit price 2^j such that $2^j \cdot y_j$ is maximized. Note that $(2^\ell, y_\ell)$ is also a candidate for satisfying bundle I .

- If Pricing assigns $(2^\ell, y_\ell)$, since after the assignment, $\delta_i^{\ell+1, \lceil \log I \rceil} > 0$ for all $i \in I$, the revenue achieved on I by Pricing is at least half of the maximal revenue on this bundle.
- Otherwise, Pricing assigns $(2^j, y_j)$ such that $j \neq \ell$. From the choosing criteria, $2^j \cdot y_j \geq 2^\ell \cdot y_\ell$.
- If $\delta_i^{\ell+1, \lceil \log I \rceil} > y_\ell$ for all $i \in I$, from the above analysis, the revenue on I by Pricing is at least half of the maximal revenue on this bundle.
- Otherwise, $\delta_i^{\ell+1, \lceil \log I \rceil} = y_\ell$ for some $i \in I$. Since $2^j \cdot y_j \geq 2^\ell \cdot y_\ell$, the revenue achieved on I by Pricing plus the current revenue on level $\lceil \log I \rceil$ in stage $\ell + 1$ of type i is at least

$$\frac{2^\ell \cdot m}{\lceil \log h \rceil (\lceil \log k \rceil + 1)} \tag{8}$$

This is because if we assign $(2^\ell, y_\ell)$ for this bundle, level $\lceil \log I \rceil$ in stage $\ell + 1$ of type i is full.

Similar to the analysis in Lemma 3, suppose more than one assigned small bundles I_1, I_2, \dots with unit price p_1, p_2, \dots within $[2^\ell, 2^{\ell+1})$ and the sizes of these bundles are all within $(2^{\lceil \log |I_1| \rceil - 1}, 2^{\lceil \log |I_1| \rceil}]$, if all these bundles share type i , the total revenue on such bundles by the optimal scheme is at most

$$2^{\ell+1} \cdot m \cdot 2^{\lceil \log |I_1| \rceil} \tag{9}$$

The ratio between the (9) and (8) is $O(\sqrt{k} \cdot \log h \log k)$.

Mapping the assignments in O_n^s to the corresponding assignments of Pricing described above, each assignment by Pricing is counted at most TWICE. Combining with the above analysis,

$$\frac{O_n^s}{ALG} \leq O(\sqrt{k} \cdot \log h \log k).$$

\square

Lemma 6 $\frac{O_n^l}{ALG} \leq O(\sqrt{k} \cdot \log h \log k)$.

Proof The proof of this lemma is similar to the proofs in Lemmas 3 and 4. Consider a bundle I such that the revenue on I from the optimal algorithm belongs to O_n^l and the unit price is $p \in [2^\ell, 2^{\ell+1})$. For the same bundle I , the algorithm Pricing chooses the unit price 2^j such that $2^j \cdot y_j$ is maximized. Since any level $\lceil \log |I| \rceil$ in stage $\ell + 1$ of item $i \in I$ is not full, $(2^\ell, y_\ell)$ is also a candidate of assignment of the algorithm Pricing.

- If Pricing assigns $(2^\ell, y_\ell)$, since $\delta_i^{\ell+1, \lceil \log |I| \rceil} > 0$ for all $i \in I$ after the assignment of the whole user sequence, the revenue achieved on I is at least half of the maximal revenue on this bundle.
 - Otherwise, Pricing assigns $(2^j, y_j)$ such that $j \neq \ell$. From the choosing criteria, $2^j \cdot y_j \geq 2^\ell \cdot y_\ell$.
- If $\delta_i^{\ell+1, \lceil \log |I| \rceil} > y_\ell$ for all $i \in I$, from the above analysis, the revenue on I by Pricing is at least half of the maximal revenue on this bundle.
- Otherwise, $\delta_i^{\ell+1, \lceil \log |I| \rceil} = y_\ell$ for some $i \in I$. Since $2^j \cdot y_j \geq 2^\ell \cdot y_\ell$, the revenue achieved on I by Pricing plus the revenue on level $\lceil \log |I| \rceil$ in stage $\ell + 1$ is at least

$$\frac{2^\ell \cdot m \cdot 2^{\lceil \log |I| \rceil - 1}}{\lfloor \log h \rfloor (\lfloor \log k \rfloor + 1)} \tag{10}$$

This is because if we assign $(2^\ell, y_\ell)$ for this bundle, level $\lceil \log |I| \rceil$ in stage $\ell + 1$ of type i is full, and the size of each bundle on this level is at least $2^{\lceil \log |I| \rceil - 1}$. From the optimal scheme, the total revenue on assignments from unit price in between $[2^\ell, 2^{\ell+1})$ and bundle size in between $(2^{\lceil \log |I| \rceil - 1}, 2^{\lceil \log |I| \rceil}]$ is at most

$$2^{\ell+1} \cdot m \cdot k \tag{11}$$

The ratio between (11) and (10) is $O(\sqrt{k} \cdot \log h \log k)$

Mapping the revenue of assignments in O_n^l to the assignments by Pricing, from the above analysis, each assignment by Pricing is counted at most TWICE. Thus,

$$\frac{O_n^l}{ALG} \leq O(\sqrt{k} \cdot \log h \log k).$$

□

Now we give the main conclusion of this paper.

Theorem 7 *The competitive ratio of the algorithm Pricing is at most*

$$O(\sqrt{k} \cdot \log h \log k).$$

Proof From the definitions on $O_f^s, O_f^l, O_n^s,$ and O_n^l , these four classes are disjoint. Note that if a requested bundle cannot be satisfied, it must belongs to O_f^s or O_f^l . Combining Lemmas 2, 3, 4 and 5, the competitive ratio of the algorithm Pricing is $O(\sqrt{k} \cdot \log h \log k)$. □

4 Concluding remark

In this paper, we study the online pricing problem for bundles of multiple items. To maximize the revenue, the seller must have a good strategy to decide the price and amount of bundles

sold to each user on her arrival. This model has several realistic motivating applications. We give an online pricing strategy for this problem, and the upper bound of the competitive ratio is proved. Moreover, we also show the lower bound of the performance ratio. In this model, we assume that the amount of each item in any assigned bundle must be the same and each type of items has the same amount m . Based on this assumption, we have the competitive algorithm by partitioning the amount of items into levels and stages. In future research, it would be interesting to examine a generalization of this model that removes this assumption.

Acknowledgments We thank Jack Snoeyink and the anonymous referees for their very helpful insights and comments on this paper. Y. Zhang was supported by NSFC 11171086. F.Y.L. Chin was supported by Hong Kong GRF grant. H.-F. Ting was supported by Hong Kong GRF grant HKU716412E.

References

1. Babaioff, M., Dughmi, S., Kleinberg, R., Slivkins A.: Dynamic pricing with limited supply. In: Proceedings of the 13th ACM Conference on Electronic Commerce (EC12), pp. 74–91. Valencia, Spain, June 4–8 (2012)
2. Balcan, M.-F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: Proceedings of the 9th ACM Conference on Electronic Commerce (EC'08), pp. 50–59 (2008)
3. Blum, A., Gupta, A., Mansour Y., Sharma, A.: Welfare and profit maximization with production costs. In: Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS 2011), pp. 77–86, Palm Springs (2011)
4. Blum, A., Hartline J.D.: Near-optimal online auctions. In: Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2005), pp. 1156–1163. Vancouver, British Columbia, January 23–25, 2005 (2005)
5. Bansal, N., Chen, N., Cherniavsky, N., Rurda, A., Schieber, B., Sviridenko, M.: Dynamic pricing for impatient bidders. *ACM Trans. Algorithm.* 6(2), 35 (2010)
6. Briest, P.: Uniform budgets and the envy-free pricing problem. In: Proceedings of the 35th International Colloquium on Automata, Languages and Programming, Part I, Reykjavik, Iceland, July 07–11, 2008 (2008)
7. Briest, P., Krysta, P.: Buying cheap is expensive: hardness of non-parametric multi-product pricing. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 716–725. New Orleans, Louisiana, January 07–09, 2007 (2007)
8. Chen, N., Deng, X.: Envy-free pricing in multi-item markets In: Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP 2010), pp. 418–429 (2012)
9. Chen, N., Ghosh, A., Vassilvitskii, S.: Optimal envy-free pricing with metric substitutability. In: Proceedings of the 9th ACM Conference on Electronic Commerce (EC'08), pp. 60–69 (2008)
10. Chawla, S., Hartline, J.D., Kleinberg, R.: Algorithmic pricing via virtual valuations. In: Proceedings of the 8th ACM Conference on Electronic Commerce (EC'07), pp. 243–251. San Diego, California, USA, June, 13–16 (2007)
11. Chakraborty, T., Huang, Z., Khanna, S.: Dynamic and non-uniform pricing strategies for revenue maximization. In: Proceedings of the 2009 IEEE 50th annual symposium on foundations of computer science (FOCS 2009), 495 C504, Atlanta (2009)
12. Cheung, M., Swamy, C.: Approximation algorithms for single-minded envy-free profit-maximization problems with limited supply. In: Proceedings of 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2008), pp. 35–44 (2008)
13. Elbassioni, K., Raman, R., Ray, S., Sitters, R.: On profit-maximizing pricing for the highway and tollbooth problems. In: Proceedings of the 2nd International Symposium on Algorithmic Game Theory, p. 275–286. Paphos, Cyprus, October 18–20, 2009 (2009)
14. Elbassioni, K., Sitters, R., Zhang, Y.: A quasi-PTAS for profit-maximizing pricing on line graphs. In: Proceedings of the 15th Annual European Conference on Algorithms (ESA'07), pp. 451–462 (2007)
15. Fiat, A., Wingarten A.: Envy, multi envy and revenue maximization. In: Proceedings of the 5th International Workshop on Internet and Network Economics (WINE 2009), pp. 498–504 (2009)
16. Grigoriev, A., van Loon, J., Sitters, R., Uetz, M.: How to sell a graph: guidelines for graph retailers. In: Proceedings of the 32nd International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2006), pp. 125–136 (2006)

17. Guruswami, V., Hartline, J., Karlin, A., Kempe, D., Kenyon, C., McSherry F.: On profit-maximizing envy-free pricing. In: Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 05), pp. 1164–1173 (2005)
18. Im S., Lu , P., Wang Y.: Envy-free pricing with general supply constraints. In: Proceedings of the 6th International Workshop on Internet and Network Economics (WINE 2010), pp. 483–491 (2010)
19. Krauthgamer, Robert, Mehta, Aranyak, Rudra, Atri: Pricing commodities. *Theor. Comput. Sci.* **412**(7), 602–613 (2011)
20. Zhang, Y., Chin, F., Ting, H.-F.: Competitive algorithms for online pricing. In: Proceedings of the 17th Annual International Computing and Combinatorics Conference (COCOON 2011), LNCS 6842, pp. 391–401 (2011)