Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Optimal Pricing for Submodular Valuations with Bounded Curvature

Takanori Maehara
Shizuoka University
maehara.takanori@shizuoka.ac.jp

Yasushi Kawase
Tokyo Institute of Technology
kawase.y.ab@m.titech.ac.jp

Hanna Sumita
National Institute of Informatics
sumita@nii.ac.jp

Katsuya Tono
University of Tokyo
katsuya_tono@mist.u-tokyo.ac.jp

Ken-ichi Kawarabayashi
National Institute of Informatics
k_keniti@nii.ac.jp

Abstract

The optimal pricing problem is a fundamental problem that arises in combinatorial auctions. Suppose that there is one seller who has indivisible items and multiple buyers who want to purchase a combination of the items. The seller wants to sell his items for the highest possible prices, and each buyer wants to maximize his utility (i.e., valuation minus payment) as long as his payment does not exceed his budget. The optimal pricing problem seeks a price of each item and an assignment of items to buyers such that every buyer achieves the maximum utility under the prices. The goal of the problem is to maximize the total payment from buyers. In this paper, we consider the case that the valuations are submodular. We show that the problem is computationally hard even if there exists only one buyer. Then we propose approximation algorithms for the unlimited budget case. We also extend the algorithm for the limited budget case when there exists one buyer and multiple buyers collaborate with each other.

1 Introduction

1.1 Background and motivation

In a *combinatorial auction* (Blumrosen and Nisan 2007; Cramton, Shoham, and Steinberg 2006), a seller has a set of indivisible items, and buyers purchase a combination of the items. The seller wants to sell his items to the buyers for the highest possible prices, and each buyer wants to purchase a set of items that is valuable for him and also has a reasonable price. More precisely, each buyer purchases a set of items that maximizes his *utility* within the limits of his budget; here, the utility for a set of items is the valuation for him (minus the payment for purchase). Thus, the seller seeks a price of each item (not bundle) and an assignment of items to buyers such that they are *stable*, i.e., no buyer can gain more utility by changing the set of items that he purchases. The goal is to maximize the total profit obtained from the buyers. The stability captures a fairness condition for the individual buyers (Goldberg and Hartline 2003; Guruswami et al. 2005; Cheung and Swamy 2008; Anshelevich, Kar, and Sekar 2015). In general, such a problem is called the *optimal pricing problem* and studied in many situations.

In this paper, we assume that the valuations of buyers are represented by *submodular* functions, which capture the notion of the diminishing returns property. Submodular functions appear in many situations (Bach 2010; Soma and Yoshida 2015), and have been studied extensively. For simplicity, we assume that every buyer truthfully tells the seller his valuation and his budget. The purpose of this paper is to analyze theoretical properties of the optimal pricing problems with submodular valuations, and to propose (approximation) algorithms for the problems. We analyze the performance of the algorithms in terms of the *curvatures* of the valuations, which capture the degree of nonlinearity. The curvature has been used to derive better approximation ratios for several submodular optimization problems (Iyer and Bilmes 2013; Iyer, Jegelka, and Bilmes 2013; Sviridenko, Vondrák, and Ward 2015; Vondrák 2010).

1.2 Our contributions

We summarize our results for the optimal pricing problem.

We first show that the optimal pricing problem with submodular valuations is NP-hard even for instances derived from our application (Theorem 5). Moreover, there exists an instance that requires exponentially many oracle evaluations in the oracle model (Theorem 6).

Our main result is to propose approximate pricing algorithms for the following three cases: single buyer case (Algorithm 1), multiple buyers case (Algorithm 2), and multiple collaborating buyers case (Algorithm 3). Then we show the approximation ratios for these algorithms in terms of the curvatures of the valuations (Theorems 7, 10, 12). Our algorithms output a nearly optimal solution if the curvatures are small. We will point out that a practical application of our problem has submodular valuations whose curvatures are typically small by using a general upper bound on curvatures (Theorem 2). This justifies our analysis of the optimal pricing problem using curvatures. The application is a similar problem to the budget allocation problem, which is widely studied both theoretically and practically in computational advertising. Due to the space constraint, we omit the proofs, which can be found in the full version (Maehara et al. 2016).

We also conduct computational experiments on some synthetic and realistic datasets to evaluate the proposed pricing algorithms (Section 6). Our algorithm performs better than baseline algorithms. To the best of our knowledge, no prior work proposed a suitable algorithm for the optimal pricing problem with submodular valuations.

1.3 Related Work

The optimal pricing problem is also referred to as the *profit maximizing pricing problem*. There are many works for the case when valuations of buyers are *unit-demand* or *single-minded* (Aggarwal et al. 2004; Goldberg and Hartline 2003; Goldberg, Hartline, and Wright 2001; Guruswami et al. 2005; Cheung and Swamy 2008; Anshelevich, Kar, and Sekar 2015). A valuation f is called unit-demand if $f(X) = \max_{x \in X} f(x)$ for any X with $|X| \geq 2$, and called single-minded if for some S^*, it holds that $f(X) = f(S^*) > 0$ for any $X \supseteq S^*$ and $f(X) = 0$ otherwise. Any unit-demand valuation is submodular, while single-minded valuations are not necessarily submodular. Guruswami et al. (2005) proved the APX-hardness of the optimal pricing problem where the valuations are all unit-demand or all single-minded. Thus, the general optimal pricing problem is computationally intractable. They also provided logarithmic approximation algorithms under the assumption that the valuations are all unit-demand or all single-minded. Since these algorithms fully rely on the assumption, they do not extend to our case.

We remark that the optimal pricing problem is different from the problem of finding a Walrasian equilibrium. A pair of a pricing and an assignment is called a *Walrasian equilibrium* (or *competitive equilibrium*) if it is stable and all positive-price items are allocated to some buyer (Blumrosen and Nisan 2007). In our model, the seller can decide a subset of items that are not allocated. This difference may improve the seller's profit. We give such an example in Remark 8.

The *winner determination problem* is also similar to the optimal pricing problem. This is the problem of finding an allocation of items to buyers that maximizes the sum of buyers' valuations in combinatorial auctions. Rothkopf et al. (1998) proved the NP-hardness of the winner determination problem. Sandholm (2002) provided an inapproximability result on the problem and some approximation algorithms for special cases. For more details of this problem, see, e.g., (Blumrosen and Nisan 2007; Cramton, Shoham, and Steinberg 2006). The winner determination problem maximizes the total valuation of buyers whereas our problem maximizes the profit of the seller. In general, these problems have different optimal solutions (see Remark 8 for an example). So our problem setting is different from the winner determination problem.

2 Preliminaries

In this section, we review submodular functions and the optimal pricing problem, and describe our motivation to study the optimal pricing problem.

2.1 Submodular function and curvature

Let V be a finite set. A function $f : 2^V \to \mathbb{R}$ is *submodular* if it satisfies

$$f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y) \tag{1}$$

for all $X, Y \subseteq V$ (Fujishige 2005). This condition is equivalent to the *diminishing returns property*: $f(X) - f(X \setminus x) \geq f(Y) - f(Y \setminus x)$ for all $X \subseteq Y \subseteq V$; here we denote "$X \setminus \{x\}$" by "$X \setminus x$" for notational simplicity. We say that f is *monotone nondecreasing* if $f(X) \leq f(Y)$ for all $X \subseteq Y$. In this paper, we assume that $f(\emptyset) = 0$.

The diminishing returns property is a fundamental principle of economics (Samuelson and Nordhaus 2004). Thus, submodular functions are often used to model user utilities and preferences. They also appear in combinatorial optimization (Fisher, Nemhauser, and Wolsey 1978; Fujishige 2005), social network analysis (Kempe, Kleinberg, and Tardos 2015), and machine learning (Bach 2010; Pan et al. 2014; Soma and Yoshida 2015).

For a monotone nondecreasing function $f : 2^V \to \mathbb{R}$ and an integer $s \in \{0, \ldots, |V|\}$, the *curvature* $\kappa(s)$ is defined by the largest nonnegative number that satisfies

$$(1 - \kappa(s)) f(x) \leq f(X) - f(X \setminus x) \tag{2}$$

for all $|X| = s$ and $x \in X$ (Conforti and Cornuéjols 1984).[1] If f is submodular, then its curvature $\kappa(s)$ is a monotone nondecreasing sequence by the diminishing returns property.

We remark that computing $\kappa(s)$ is difficult since it requires exponentially many function evaluations; therefore, we cannot use explicitly the value in an algorithm.

2.2 Optimal pricing problem

Here we define the optimal pricing problem. Suppose that a seller wants to sell indivisible items V to buyers $N = \{1, \ldots, n\}$ simultaneously. Each buyer $i \in N$ has a budget B_i and a valuation function $f_i : 2^V \to \mathbb{R}$, where f_i is a monotone nondecreasing submodular function. We denote by κ_i the curvature of f_i for $i \in N$. In this problem, we find a price vector $p \in \mathbb{R}^V$ and an assignment $(X_1, \ldots, X_n)$ which is a subdivision of V. For a price vector p and an item set $X \subseteq V$, let $p(X) = \sum_{x \in X} p(x)$. Buyers are assumed to have quasi-linear utility, i.e., the utility of $i \in N$ is given by $f_i(X_i) - p(X_i)$.

The seller wants to maximize the total profit $p(X_1) + \cdots + p(X_n)$. On the other hand, each buyer also wants to maximize his utility, as long as his payment to the seller does not exceed his budget. Therefore, the assignment must satisfy some "agreement" condition. We say that a price vector p and assignment $(X_1, \ldots, X_n)$ pair is *stable* if it satisfies

$$f_i(X_i) - p(X_i) \geq f_i(X) - p(X) \tag{3}$$

for any $i \in N$ and all $X \subseteq V$. The stability condition means that each buyer i has no incentive to change his allocation X_i under the pricing p. For a price vector p, we define the *demand set* of buyer i as a family of sets X satisfying (3), denoted by

$$D_i(p) = \operatorname{argmax}\{f_i(X) - p(X) \mid X \subseteq V,\ p(X) \leq B_i\}. \tag{4}$$

The stability condition is necessary to avoid a grudge or an antipathy of buyers even when each buyer i knows his own allocated items X_i and every price of the items V.

[1] Originally, Conforti and Cornuéjols (1984) introduced *total curvature* and *greedy curvature* for monotone nondecreasing submodular functions.

The *optimal pricing problem* seeks a price vector $p \in \mathbb{R}^V$ and an assignment $(X_1, \ldots, X_n)$ that maximizes the total profit $p(X_1) + \cdots + p(X_n)$ under the stability condition. It is formulated as

$$\begin{array}{ll} \text{maximize} & \sum_{i \in N} p(X_i) \\ \text{subject to} & X_i \in D_i(p)\ (i \in N), \\ & X_i \cap X_j = \emptyset\ (i \neq j). \end{array} \tag{5}$$

We propose algorithms for the problem (5) where all buyers have unlimited budgets, i.e., $B_i = +\infty$ for all $i \in N$. We extend our results to the limited budget case (see (Maehara et al. 2016)).

2.3 Application

We present an application that motivates us to study the optimal pricing problem with submodular valuations. We will claim that the curvatures of valuations are small in practice.

Consider that the publisher (= seller) has a set V of marketing channels, and that there is a set N of advertisers (= buyers) that have budgets B_i $(i \in N)$. Each advertiser $i \in N$ purchases a subset X of channels for advertising under the budget constraint, i.e., $p(X) \leq B_i$. The valuation $f_i(X)$ of $X \subseteq V$ for advertiser i is the expected value of the total revenue from loyal customers influenced by marketing channels in X. Let $p \in \mathbb{R}^V$ be the price vector, i.e., $p(v)$ is the price to publish an advertisement through marketing channel v. Each advertiser i wants to buy X_i that maximizes the total revenue minus the cost, i.e., $f_i(X_i) - p(X_i)$, under the budget constraint $p(X_i) \leq B_i$. In the following, we explicitly formulate the valuation function f_i of each advertiser i.

We adopt the bipartite influence model of advertising proposed by Alon et al. (Alon, Gamzu, and Tennenholtz 2012). Let W be set of customers. We consider a bipartite graph $G = (V \cup W, E)$. Each edge $(v, w) \in E$ indicates that marketing channel v affects customer w. Each edge (v, w) is assigned probabilities $q_i(v, w)$ $(i \in N)$, called *activation probability*. If advertiser i puts an advertisement on marketing channel v, then customer w will become a loyal customer of buyer i with probability $q_i(v, w)$.

The probability $Q_i(X, w)$ that customer w becomes a loyal customer when a advertiser i runs advertisements on $X \subseteq V$ is given by $Q_i(X, w) = 1 - \prod_{x \in X, (x,w) \in E}(1 - q_i(x, w))$. Thus, the expected number of his loyal customers is $\sum_{w \in W} Q_i(X, w)$. Let γ_i be the expected revenue from one loyal customer. The expected total revenue is given by

$$f_i(X) = \gamma_i \cdot \textstyle\sum_{w \in W} Q_i(X, w). \tag{6}$$

Since $Q_i(X, w)$ is a monotone nondecreasing submodular function in X, $f_i(X)$ is also a monotone nondecreasing submodular function.

Here, we can observe that each curvature κ_i of f_i is small $(i \in N)$. This implies that our analysis based on the curvature is particularly effective for this application.

Lemma 1. For each $i \in N$ and $s \in \{1, \ldots, |V|\}$, $\kappa_i(s) \leq \max_{|X|=s,\ x \in X,\ (x,w) \in E} Q(X \setminus x, w)$.

Theorem 2. For each $i \in N$ and $s \in \{1, \ldots, |V|\}$, if $q_i(e) \leq q$ for any $e \in E$, then it holds that $\kappa_i(s) \leq 1 - (1 - q)^{\min\{s,d\}-1}$, where d is the maximum degree of the right vertices W.

In practice, d is relatively small (e.g., $d \leq 100$) since it is the number of incoming information channels of a customer. Moreover, q is very small (e.g., $q \leq 0.001$) since it is the probability of gaining a customer through a single advertisement.

3 Single buyer

In this section, we analyze the optimal pricing problem with a single buyer (i.e., $n = 1$). We prove the NP-hardness of the problem and present a nearly optimal approximate algorithm for the buyer with an unlimited budget.

Consider that there is only one buyer with an unlimited budget. For notational convenience, let $f : 2^V \to \mathbb{R}$ be his valuation, which is a monotone nondecreasing submodular function. For a price vector p, we denote by $D(p)$ the demand set for p. When we fix an assignment, we can easily obtain the maximum profit for the assignment.

Lemma 3. Let X be an assignment. An optimal price vector for (5) with fixed X is given by $p(x) = f(X) - f(X \setminus x)$ if $x \in X$, and $p(x) = +\infty$ otherwise.

From this lemma, we obtain the following characterization of optimal solutions to (5).

Lemma 4. Let X be an assignment. There exists a price vector p such that (p, X) is optimal to (5) if and only if X achieves

$$\max_{X' \subseteq V} \textstyle\sum_{x \in X'}(f(X') - f(X' \setminus x)). \tag{7}$$

This implies that problem (5) is equivalent to (7). For any $X \subseteq V$, we denote $h(X) = \sum_{x \in X}(f(X) - f(X \setminus x))$.

We show the NP-hardness of (5) by reducing the *one-in-three positive 3-SAT problem*. Given a boolean formula in conjunctive normal form with three positive literals per clause, the one-in-three positive 3-SAT problem determines whether there exists a truth assignment to the variables so that each clause has exactly one true variable. This problem is known to be NP-complete (Schaefer 1978).

Theorem 5. Problem (5) is NP-hard even when the function f is of the form (6).

Furthermore, we obtain the result below.

Theorem 6. If f is given by an oracle, problem (5) requires exponentially many oracle evaluations.

Since (5) is NP-hard, we propose an algorithm to find an approximate pricing. Once we determine an assignment, an optimal price vector for the assignment is easily obtained from Lemma 3. Thus, we only need to find an assignment X maximizing $h(X)$. However, an overly large assignment X may have small $h(X)$ value. In our algorithm, we assign the top s elements in order of their value, for each $s = 1, \ldots, |V|$. The formal description is given in Algorithm 1.

This algorithm can be implemented to run in $O(|V| \log |V| + A|V|^2)$ time, where A is the computational cost of evaluating $f(X)$. For a variant of budget allocation problem with the bipartite graph model, if we implement the algorithm carefully, it runs in $O(|V||E|)$ time.

We analyze the approximation ratio of our algorithm.

Algorithm 1 Pricing algorithm for a single buyer.

For $s = 1, 2, \dots, |V|$
  Let X^s be the largest s elements of $f(x)$
  Price $p^s(x) = f(X^s) - f(X^s \setminus x)$ $(x \in X^s)$ and $p^s(y) = +\infty$ $(y \in V \setminus X^s)$
Return p and X that attains maximum of $p^s(X^s)$

Theorem 7. Let (p^*, X^*) be an optimal solution to (5), and let (p, X) be the output of Algorithm 1. Then, it holds that $X \in D(p)$ and $(1 - \kappa(|X^*|))p^*(X^*) \leq p(X)$.

Remark 8. Selling all items (i.e., $s = |V|$) is not always optimal, even when the function f is of the form (6). To demonstrate this, let us consider the instance of the application in Section (2.3) where there are two channels u, v and one user w with $\gamma = 1$. The activation probabilities from u and v to w are 0.9. When we use the both channels, the activation probability is $1 - (1 - 0.9)^2 = 0.99$. Thus $f(u) = f(v) = 0.9$ and $f(u, v) = 0.99$. The optimal pricing sells only a single channel $X^* = \{u\}$ at price $p^*(u) = 0.9$, and $p^*(X^*) = 0.9$. On the other hand, to sell all items $X' = \{u, v\}$, the price should be $p'(u) = p'(v) = 0.99 - 0.9 = 0.09$, and hence $p'(X') = 0.18$.

We also remark that this example shows the difference between our problem and related problems, namely, the problem of finding Walrasian equilibrium and the winner determination problem. There are two Walrasian equilibria (p', X') and (p'', X') where $p''(u) = p''(v) = 0$. Thus (p', X') achieves the maximum profit $p'(X') = 0.18$ among Walrasian eqiulibria whereas the optimal value for our problem is 0.9. When we regard this example as an instance of the winner determination problem, the optimal solution is X' and its valuation of the *buyer* is 0.99. However, the optimal solution for our problem sells only X^*, and the profit of the seller is 0.9.

We also show that if the curvature of f is small, then the optimal values of (5) and the one without the stability condition is almost the same; see the full version (Maehara, Yabe, and Kawarabayashi 2015).

4 Multiple buyers

In this section, we deal with the general optimal pricing problem (5) that admits more than one buyer. Recall that if $n = 1$, then for any assignment X, there always exists a price vector p satisfying $X \in D(p)$ (see Lemma 3). However, in general, there may not exist a price vector p such that $X_i \in D_i(p)$ for some assignment $(X_1, \dots, X_n)$. Moreover, it is difficult to determine whether or not such a price vector exists for a given assignment.

We first approach the coNP-hardness of deciding the existence of a stable price vector for a given assignment by reducing the *exact cover by 3-sets problem (X3C)*, which is NP-complete (Garey and Johnson 1979). In this problem, we are given a set E with $|E| = 3l$ and a collection $\mathcal{C} = \{C_1, \dots, C_m\}$ of 3-element subsets of E. The task is to decide whether or not $\mathcal{C}$ contains an exact cover for E, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ such that every element of E occurs in exactly one member of $\mathcal{C}'$.

Algorithm 2 Pricing algorithm for multiple buyers

For $s = 1, 2, \dots, |V|$
  Let X^s be s largest elements of $\max_{i \in N} f_i(x)$
  Price $p^s(x) = \max_{i \in N}(f_i(X^s) - f_i(X^s \setminus x))$ $(x \in X^s)$ and $p^s(y) = +\infty$ $(y \in V \setminus X^s)$
  Let $(X_1^s, \dots, X_n^s)$ be a partition of X^s such that $p^s(x) = f_i(X^s) - f_i(X^s \setminus x)$ $(\forall x \in X^s)$
Return p and $(X_1, \dots, X_n)$ that attains maximum of $\sum_{i \in N} p^s(X_i^s)$

Theorem 9. It is coNP-hard to determine, for a given assignment $(X_1, \dots, X_n)$, the existence of price vector p such that $X_i \in D_i(p)$ for all $i \in N$.

We also show that, given a price vector p, it is NP-hard to decide the existence of an assignment $\mathbf{X} = (X_1, \dots, X_n)$ such that $(p, \mathbf{X})$ is stable (see the full version (Maehara et al. 2016)).

By above results, it is difficult to find a stable pair $(p, \mathbf{X})$ for given p (or $\mathbf{X}$). Therefore in order to obtain efficiently an approximate solution, we take a natural approach that we slightly relax the stability condition.

For any positive number $\alpha \leq 1$ and each buyer i, we define the α-demand set of buyer i as $D_i^\alpha(p) = \{X \subseteq V \mid f_i(X) - p(X) \geq \alpha f_i(Y) - p(Y),\ \forall Y \subseteq V\}$. For a price vector p and an assignment $\mathbf{X} = (X_1, \dots, X_n)$, we say that $(p, \mathbf{X})$ is *α-stable* if $X_i \in D_i^\alpha$ for all $i \in N$.

We propose a pricing algorithm in Algorithm 2. The algorithm can be implemented to run in $O(An|V|^2 + |V| \log |V|)$ time, where A is the computational cost of evaluating $f_i(X)$ $(i \in N)$. It has the following theoretical guarantee. Here we denote $\kappa(s) = \max \kappa_i(s)$.

Theorem 10. Let p^* and $\mathbf{X}^* = (X_1^*, \dots, X_n^*)$ be the optimal solution to (5) and let p and $\mathbf{X} = (X_1, \dots, X_n)$ be the solution obtained by Algorithm 2. Then, for $s = |X_1 \cup \cdots \cup X_n|$ and $s^* = |X_1^* \cup \cdots \cup X_n^*|$, $(p, \mathbf{X})$ is $(1 - \kappa(s))$-stable and $(1 - \kappa(s^*)) \sum_{i \in N} p^*(X_i^*) \leq \sum_{i \in N} p(X_i)$.

5 Multiple collaborating buyers

In this section, we analyze the optimal pricing problem with *collaborating buyers*, i.e., the case where buyers cooperate to maximize the total utilities. This occurs when buyers are employed by the same organization. We present an approximation algorithm for this problem.

We first describe the model. Assume that there are buyers $N = \{1, \dots, n\}$, whose valuation functions are given by $f_1, \dots, f_n$. Let $(X_1, \dots, X_n)$ be an assignment. Since the goal of buyers is to maximize the sum of their utilities, the stability condition is written as $\sum_{i \in N}(f_i(X_i) - p(X_i)) \geq \sum_{i \in N}(f_i(Y_i) - p(Y_i))$ for any assignment $(Y_1, \dots, Y_n)$.

Because only the total amount of the utilities matters, the publisher only needs to find a set X and a price vector p that satisfies the above stability condition for some partition of X. Thus, in the following, we assume that there exists one buyer who represents the set of original buyers. Let $f : 2^V \to \mathbb{R}$ be an aggregated valuation function defined by $f(X) = \max_{(X_1, \dots, X_n) \text{: partition of } X} \sum_{i \in N} f_i(X_i)$ for $X \subseteq V$. Note

Algorithm 3 Pricing algorithm for collaborating buyers

For $s = 1, 2, \ldots, |V|$
  Let X^s be s largest elements of $f(x)$ $(= \max_i f_i(x))$
  Price $p^s(x) = \min_{i \in N} \frac{f_i(X^s) - f_i(X^s \setminus x)}{f_i(x)} f(x)$ $(x \in X^s)$
and $p(y) = +\infty$ $(y \in V \setminus X^s)$
Return p and X that attains maximum of $p^s(X^s)$

that $f(X)$ is monotone nondecreasing but not necessarily submodular (See the full version (Maehara et al. 2016)).

By using the aggregated valuation function, the stability condition is equivalent to the condition that $f(X) - p(X) \geq f(Y) - p(Y)$ for all $Y \subseteq V$. Thus, the demand set is defined as (4), and the optimal pricing problem for collaborating buyers is formulated as

$$\text{maximize} \quad p(X) \quad \text{subject to} \quad X \in D(p). \qquad (8)$$

Although the aggregated valuation function is not necessarily submodular, problem (8) has a similar formulation to (5) with a single buyer. We obtain a similar result to Lemma 3.

Lemma 11. Let X be an assignment. An optimal price vector for (8) with fixed X is given by $p(x) = \min_{Y \subseteq X : x \in Y} (f(Y \cup x) - f(Y))$ if $x \in X$, and $p(x) = +\infty$ otherwise.

From this lemma, we see that problem (8) is equivalent to the problem of finding $X \subseteq V$ maximizing $\sum_{x \in X} \min_{Y \subseteq X : x \in Y} (f(Y) - f(Y \setminus x))$. Thus, we can apply the same principles as the ones of Algorithm 1. In fact, setting prices $p^s(x) = \min_{Y \subseteq X^s : x \in Y} (f(Y) - f(Y \setminus x))$ implies a similar result. However, computing $f(X)$ is intractable (this problem is called submodular welfare problem) and hence it is hard to evaluate the value $\min_{Y \subseteq X : x \in Y} (f(Y) - f(Y \setminus x))$. Thus, we need a further modification.

Our algorithm, summarized in Algorithm 3, finds an approximate solution to (8) in $O(An|V|^2 + |V| \log |V|)$ time, where A is the computational cost of evaluating $f_i(X)$ $(i \in N)$. We analyze the approximation ratio of our algorithm. Let $\kappa_1, \ldots, \kappa_n$ be curvatures of $f_1, \ldots, f_n$ and $\kappa(s) = \max_j \kappa_j(s)$ for $s = 1, \ldots, |V|$.

Theorem 12. Let (p^*, X^*) be an optimal solution to (8), and let (p, X) be the output of Algorithm 3. It then holds that $X \in D(p)$ and $(1 - \kappa(|X^*|)) p^*(X^*) \leq p(X)$.

To prove this theorem, we show the following two lemmas.

Lemma 13. For a set X and $x \in X$, it holds that $f(X) - f(X \setminus x) \leq f(x)$.

Lemma 14. For a set X and $x \in X$, it holds that $f(X) - f(X \setminus x) \geq \min_{i \in N} \frac{f_i(X) - f_i(X \setminus x)}{f_i(x)} f(x) \geq (1 - \kappa(|X|)) f(x)$.

6 Experiments

In this section, we present experimental results on our pricing algorithms for a variant of budget allocation problem, which are described in Section 2.3. All experiments were conducted on an Intel Xeon E5-2690 2.90GHz CPU (32 cores) with 256GB memory running Ubuntu 12.04. All codes were implemented in Python 2.7.3.

Table 1: Ranking by #plays.

rank	artist – music	#play	UU
1	The Postal Service – Such Great Heights	3992	321
2	Boy Division – Love Will Tear Us Apart	3663	318
3	Radiohead – Karma Police	3534	346
4	Muse – Supermassive Black Hole	3483	263
5	Death Cab For Cutie – Soul Meets Body	3479	233
6	The Knife – Heartbeats	3156	177
7	Muse – Starlight	3060	260
8	Arcade Fire – Rebellion (Lies)	3048	292
9	Britney Spears – Gimme More	3004	59
10	The Killers – When You Were Young	2998	235

Table 2: Ranking by prices.

rank	original	artist – music	price
1	1	The Postal Service – Such Great Heights	3.330
2	8	Arcade Fire – Rebellion (Lies)	2.101
3	4	Muse – Supermassive Black Hole	2.029
4	11	Interpol – Evil	2.026
5	3	Radiohead – Karma Police	2.003
6	6	The Knife – Heartbeats	1.992
7	12	Kanye West – Love Lockdown	1.893
8	17	Arcade Fire – Neighborhood #1 (Tunnels)	1.868
9	23	Kanye West – Heartless	1.788
10	24	Radiohead – Nude	1.770

We performed the following five experiments: For the single advertiser case, (1) we computed prices of each channel for a realistic dataset; (2) we compared the proposed algorithm with other baseline algorithms; (3) we evaluated the scalability of the proposed algorithm; and (4) we observed the relationship between the activation probabilities and the number of allocated channels. For the multiple advertisers case and the multiple collaborating advertisers case, (5) we observed the relationship between the obtained profit and the number of advertisers.

For these experiments, we used two random synthetic networks (`Uniform`, `PowerLaw`) and three networks constructed from real-world datasets (`Last.fm`, `MovieLens`, `BookCrossing`). Throughout the experiments, we assume that the expected revenue from one loyal customer is 1, i.e., $\gamma_i = 1$. Due to the space constraint, the description of the datasets is given in the full version (Maehara et al. 2016).

(1) Typical result First, we ran Algorithm 1 to `Last.fm` dataset to compute prices for the musics played in Last.fm. Top 10 frequently played musics and top 10 high price musics are displayed in Table 1 and Table 2, respectively. We can observe that some musics with a large number of plays (or unique users) are not assigned high prices. This occurs because of the stability condition.

(2) Comparison with other pricing algorithms Next, we compared Algorithm 1 with the following four baseline algorithms:

Selling all items. Assign $X = V$ and price $p(v) = f(V) -$

Table 3: Comparison of pricing algorithms on several datasets. Each value is the ratio of the profit obtained by the algorithm and the proposed algorithm.

	Proposed	SellAll	Random	Scaled	Ascend
`Uniform`	**1.00**	0.89	0.55	0.98	0.96
`PowerLaw`	**1.00**	0.89	0.65	0.98	0.51
`Last.fm`	**1.00**	0.71	0.46	0.99	0.78
`MovieLens`	**1.00**	0.58	0.48	0.96	0.67
`BookCrossing`	**1.00**	**1.00**	0.39	0.78	0.43

$f(V \setminus v)$ for each $v \in V$. This algorithm gives a stable assignment.

Random pricing. Price $p(v) \in [0, f(v)]$ uniformly at random for each $v \in V$ and find an assignment X by the greedy algorithm.

Scaled pricing. Price $p(v) = \alpha f(v)$ for each v and find an assignment X by the greedy algorithm. α is chosen optimally from $\{0.1, \ldots, 1.0\}$.

Ascending pricing. Start from $X = V$ and $p(v) = 0$ $(v \in V)$, repeat the following process: Price $p(v) = \min_{X : x \in X}(f(X) - f(X \setminus x))$ for each $v \in X$, remove $\tilde{x}$ that attains the minimum from X, and then price $p(\tilde{x}) = +\infty$. This algorithm is motivated by the ascending auction (Krishna 2009).

We remark that there are no existing algorithms that are directly applicable to the optimal pricing problem with submodular valuations (see also Section 1.3).

We used all the networks described above; we set $|V| = 100$, $|W| = 10000$, $d = 10$ and $q_{\max} = 0.3$ for `Uniform` and `PowerLaw`. The result is summarized in Table 3. The proposed algorithm outperforms all compared algorithms for all datasets

(3) Scalability We evaluated the scalability of the proposed algorithm. We used `Uniform` with $|V| \in \{16, 32, \ldots, 1024\}$, $|W| \in \{100, 1000, 10000, 100000\}$, $d = 10$, and $q_{\max} = 0.3$. We also conducted the same experiment on `PowerLaw` but we omit it since it yields very similar results.

The result is shown in Figure 1. The elapsed times were (roughly) proportional to both $|V|$ and $|W|$. This is consistent with our analysis that the proposed algorithm runs in $O(|V||E|)$ time, and the number of edges is proportional to $|W|$ for these networks. Therefore, the proposed algorithm scales to moderately large networks.

(4) Number of allocated channels and activation probabilities We observe the relationship between activation probability and the obtained allocation. We used `Uniform` and `PowerLaw` with the parameters $|V| = 100$, $|W| = 10000$, and $d = 10$. We controlled the maximum activation probability $q_{\max} \in \{0.05, 0.10, \ldots, 0.95\}$ and observe the number of assigned marketing channels.

The result is shown in Figure 2. For both networks, when $q_{\max}$ was small the proposed algorithm assigned all channels, and when $q_{\max}$ was large it assigned a few channels. The number of assigned channels decreased much faster in `PowerLaw` than in `Uniform`, since there were highly correlated marketing channels in `PowerLaw`.

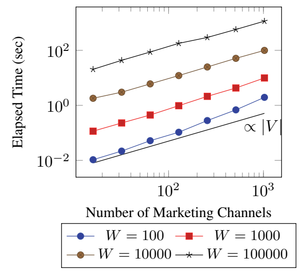

Figure 1: Scalability of the proposed algorithm.

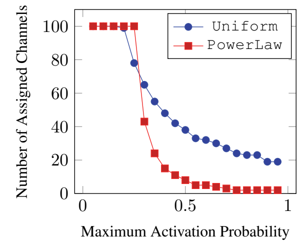

Figure 2: The number of assigned channels versus edge probability.

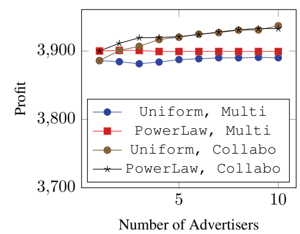

Figure 3: Profit versus the number of non-collaborating or collaborating advertisers.

(5) Profit and the number of advertisers Next, we conducted experiments on the multiple advertisers case and the multiple collaborating advertisers case. Here we observe the relationship between the profit and the number of advertisers in these settings. For these experiments, we modified `Uniform` and `PowerLaw` to assign multiple probabilities $q_1(e), \ldots, q_n(e)$ for each edge, each of which follows the uniform distribution on $[0, q_{\max}]$.

The result is shown in Figure 3. By comparing two results obtained by the multiple (non-collaborating) advertisers case, the number of advertisers had little influence on the profit. On the other hand, by comparing the results obtained by collaborating advertisers, the profit increased when the number of advertisers increased. Moreover, the profits obtained from the collaborating advertisers consistently outperformed those obtained from non-collaborating advertisers. This means that collaboration of advertisers yields a better profit to the publisher. Note that we could not observe the difference between `Uniform` and `PowerLaw`.

7 Conclusion

We propose some future works. One is to develop an approximate pricing algorithm for the case that multiple buyers have limited budgets. Another one is to analyze the optimal pricing problem with multiple sellers. In this study, we assumed there is one seller; who can be regarded as a monopolist. The seller can select both the assignment and the price as long as they satisfy the stability condition. This situation is highly advantageous for the seller. Finally, in this study, we do not consider nonlinear or non-anonymous pricing. It would be interesting in future work to analyze the effect of such generalizations of pricing.

Acknowledgments This work was supported by JSPS

KAKENHI Grant Number 16K16011, and JST, ERATO, Kawarabayashi Large Graph Project.

References

Aggarwal, G.; Feder, T.; Motwani, R.; and Zhu, A. 2004. Algorithms for multi-product pricing. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming*, 72–83.

Alon, N.; Gamzu, I.; and Tennenholtz, M. 2012. Optimizing budget allocation among channels and influencers. In *Proceedings of the 21st International Conference on World Wide Web*, 381–388.

Anshelevich, E.; Kar, K.; and Sekar, S. 2015. Envy-free pricing in large markets: Approximating revenue and welfare. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming*, 52–64.

Bach, F. R. 2010. Structured sparsity-inducing norms through submodular functions. In *Advances in Neural Information Processing Systems (NIPS 2010)*, 118–126.

Blumrosen, L., and Nisan, N. 2007. Combinatorial auctions. In Nisan, N.; Roughgarden, T.; Tardos, É.; and Vazirani, V. V., eds., *Algorithmic Game Theory*. Cambridge University Press. chapter 11, 267–300.

Cheung, M., and Swamy, C. 2008. Approximation algorithms for single-minded envy-free profit-maximization problems with limited supply. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 35–44.

Conforti, M., and Cornuéjols, G. 1984. Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the rado-edmonds theorem. *Discrete applied mathematics* 7(3):251–274.

Cramton, P. C.; Shoham, Y.; and Steinberg, R. 2006. *Combinatorial auctions*, volume 475.

Fisher, M. L.; Nemhauser, G. L.; and Wolsey, L. A. 1978. An analysis of approximations for maximizing submodular set functions—II. *Mathematical Programming Study* 8:73–87.

Fujishige, S. 2005. *Submodular functions and optimization*, volume 58. Elsevier.

Garey, M. R., and Johnson, D. S. 1979. *Computers and intractability: a guide to NP-completeness*. WH Freeman New York.

Goldberg, A. V., and Hartline, J. D. 2003. Envy-free auctions for digital goods. In *Proceedings of the 4th ACM conference on Electronic commerce*, 29–35. ACM.

Goldberg, A. V.; Hartline, J. D.; and Wright, A. 2001. Competitive auctions and digital goods. In *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, 735–744. Society for Industrial and Applied Mathematics.

Guruswami, V.; Hartline, J. D.; Karlin, A. R.; Kempe, D.; Kenyon, C.; and McSherry, F. 2005. On profit-maximizing envy-free pricing. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1164–1173. Society for Industrial and Applied Mathematics.

Iyer, R. K., and Bilmes, J. A. 2013. Submodular optimization with submodular cover and submodular knapsack constraints. In *Advances in Neural Information Processing Systems (NIPS 2013)*, 2436–2444.

Iyer, R. K.; Jegelka, S.; and Bilmes, J. A. 2013. Curvature and optimal algorithms for learning and minimizing submodular functions. In *Advances in Neural Information Processing Systems (NIPS 2013)*, 2742–2750.

Kempe, D.; Kleinberg, J.; and Tardos, É. 2015. Maximizing the spread of influence through a social network. *Theory of Computing* 11(4):105–147.

Krishna, V. 2009. *Auction theory*. Academic press.

Maehara, T.; Kawase, Y.; Sumita, H.; Tono, K.; and Kawarabayashi, K. 2016. Optimal Pricing for Submodular Valuations with Bounded Curvature. *ArXiv preprints cs.GT/1611.07605*.

Maehara, T.; Yabe, A.; and Kawarabayashi, K.-i. 2015. Budget allocation problem with multiple advertisers: A game theoretic view. In *Proceedings of the 32nd International Conference on Machine Learning*, 428–437.

Pan, X.; Jegelka, S.; Gonzalez, J. E.; Bradley, J. K.; and Jordan, M. I. 2014. Parallel double greedy submodular maximization. In *Advances in Neural Information Processing Systems (NIPS 2014)*, 118–126.

Rothkopf, M. H.; Pekeč, A.; and Harstad, R. M. 1998. Computationally manageable combinational auctions. *Management science* 44(8):1131–1147.

Samuelson, P., and Nordhaus, W. 2004. *Microeconomics, 18th eds.* McGraw-Hill.

Sandholm, T. 2002. Algorithm for optimal winner determination in combinatorial auctions. *Artificial Inteligence* 135:1–54.

Schaefer, T. J. 1978. The complexity of satisfiability problems. In *Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, 216–226. ACM.

Soma, T., and Yoshida, Y. 2015. A generalization of submodular cover via the diminishing return property on the integer lattice. In *Advances in Neural Information Processing Systems (NIPS 2015)*, 847–855.

Sviridenko, M.; Vondrák, J.; and Ward, J. 2015. Optimal approximation for submodular and supermodular optimization with bounded curvature. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1134–1148. SIAM.

Vondrák, J. 2010. Submodularity and curvature: the optimal algorithm. *RIMS Kokyuroku Bessatsu B* 23:253–266.