Approximation Algorithms for Non-Single-minded Profit-Maximization Problems with Limited Supply

Khaled Elbassioni[1], Mahmoud Fouz[2], and Chaitanya Swamy[3]*

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany;
elbassio@mpi-inf.mpg.de
[2] FR Informatik, Universität des Saarlandes, Saarbrücken, Germany;
fmahmoud@mpi-inf.mpg.de
[3] Dept. of Combinatorics and Optimization, Univ. Waterloo, Waterloo, ON N2L 3G1;
cswamy@math.uwaterloo.ca

Abstract. We consider *profit-maximization* problems for *combinatorial auctions* with *non-single minded valuation functions* and *limited supply*. We obtain fairly general results that relate the approximability of the profit-maximization problem to that of the corresponding *social-welfare-maximization* (SWM) problem, which is the problem of finding an allocation $(S_1, \dots, S_n)$ satisfying the capacity constraints that has maximum total value $\sum_j v_j(S_j)$. Our results apply to both structured valuation classes, such as *subadditive valuations*, as well as *arbitrary valuations*. For subadditive valuations (and hence *submodular, XOS valuations*), we obtain a solution with profit $OPT_{SWM}/O(\log c_{\max})$, where OPT_{SWM} is the optimum social welfare and $c_{\max}$ is the maximum item-supply; thus, this yields an $O(\log c_{\max})$-approximation for the profit-maximization problem. Furthermore, given *any* class of valuation functions, if the SWM problem for this valuation class has an LP-relaxation (of a certain form) and an algorithm "verifying" an *integrality gap* of α for this LP, then we obtain a solution with profit $OPT_{SWM}/O(\alpha \log c_{\max})$, thus obtaining an $O(\alpha \log c_{\max})$-approximation. The latter result implies an $O(\sqrt{m} \log c_{\max})$-approximation for the profit maximization problem for combinatorial auctions with *arbitrary valuations*, and an $O(\log c_{\max})$-approximation for the non-single-minded *tollbooth problem* on trees. For the special case, when the tree is a path, we also obtain an incomparable $O(\log m)$-approximation (via a different approach) for subadditive valuations, and arbitrary valuations with unlimited supply. [4]

1 Introduction

Profit (or revenue) maximization is a classic and fundamental economic goal, and the design of computationally-efficient item-pricing schemes for various profit-maximization problems has received much recent attention [1, 11, 2, 4, 3]. We study the algorithmic problem of *item-pricing for profit-maximization* for *general* (multi unit) *combinatorial auctions* (CAs) with *limited supply*. There are n customers and m items. Each item is available in some limited supply or capacity, and each customer j has a value

* Supported in part by NSERC grant 327620-09 and an Ontario Early Researcher Award

[4] Omitted proofs can be found in the full version of the paper.

$v_j(S)$ for each subset S of items specifying the maximum amount she is willing to pay for that set (with $v_j(\emptyset) = 0$). Given a pricing of the items, a *feasible allocation* is an assignment of a (possibly empty) subset S_j to each customer j satisfying (i) the *budget constraints*, which require that the price of S_j (i.e., the total price of the items in S_j) is at most $v_j(S_j)$, and (ii) the *capacity constraints*, which stipulate that the number of customers who are allocated an item be at most the supply of that item. The objective is to determine item prices that maximize the total profit or revenue earned by selling items to the customers. Guruswami et al. [11] introduced the *envy-free* version of the problem, where there is the additional constraint that the set assigned to a customer must maximize her utility (defined as value$-$price). Item pricing has an appealing simplicity and enforces a basic notion of fairness wherein the seller does not discriminate between customers who get the same item(s). Our focus on item pricing is in keeping with the vast majority of work on algorithms for profit-maximization (for example, the above references; in fact, with unlimited supply and unit-demand valuations, our problem essentially reduces to the *Max-Buy* model in [1]). Various current trading practices are described by item pricing, and thus it becomes pertinent to understand what guarantees are obtainable via such schemes. Profit-maximization problems are typically *NP*-hard, even in various specialized settings, so we will be interested in designing approximation algorithms for these problems.

The framework of combinatorial auctions is an extremely rich framework that encapsulates a variety of applications. In fact, recognizing the generality of the envy-free profit-maximization problem for CAs, Guruswami et al. [11] proceeded to study various more-tractable special cases of the problem. In particular, they introduced the following two structured problems in the *single-minded* (SM) setting, where each customer desires a single fixed set: (a) the *tollbooth problem* where the items are edges of a graph and the customer-sets correspond to paths in this graph, which can be interpreted as the problem of pricing transportation links or network connections; (b) a further special case called the *highway problem* where the graph is a path, which can also be motivated from a scheduling perspective. The non-SM versions of such structured problems can also be used to capture various interesting scenarios.

Our results. We obtain fairly general *polytime approximation guarantees* for profit-maximization problems involving *combinatorial auctions* with *limited supply* and *non-single-minded* valuations. We obtain results for both (a) certain structured valuation classes, namely *subadditive valuations* (where $v(A) + v(B) \geq v(A \cup B)$) and hence, *submodular* valuations, which have been intensely studied recently (e.g. [14, 8, 9, 3]; and (b) *arbitrary valuations*. Our results relate the approximability of the profit-maximization problem to that of the corresponding *social-welfare-maximization* (SWM) problem, which is the problem of finding an allocation $(S_1, \dots, S_n)$ satisfying the capacity constraints that has maximum total value $\sum_j v_j(S_j)$. Our main theorem, stated informally below and proved in Section 3, shows that any LP-based approximation algorithm that provides an integrality-gap bound for the SWM problem with a given class of valuations, can be leveraged to obtain a corresponding approximation guarantee for the profit-maximization problem with that class of valuations. Let $c_{\max} \leq n$ denote the maximum item supply, and OPT_{SWM} denote the optimum value of the SWM problem, which is clearly an upper bound on the maximum profit achievable.

Theorem 1. *(i) For the class of subadditive (and hence submodular) valuations, one can obtain a solution with profit $\frac{OPT_{SWM}}{O(\log c_{\max})}$, thus achieving an $O(\log c_{\max})$-approximation; (ii) Given any class of valuations for which the corresponding SWM problem admits a packing-type LP relaxation with an integrality gap of α as "verified" by an α-approximation algorithm, one can obtain a solution with profit $\frac{OPT_{SWM}}{O(\alpha \log c_{\max})}$, thereby achieving an $O(\alpha \log c_{\max})$-approximation.*

(Part (ii) above does not imply part (i), because for part (ii) we require an integrality-gap guarantee which, roughly speaking, means that we require an algorithm that returns a "good" solution for *every* profile of n valuations; see Definition 1.)

A key notable aspect of our theorem is its versatility. One can simply "plug in" various known (or easily derivable) results about the SWM problem to obtain approximation algorithms for various limited-supply profit-maximization problems. For example, as *corollaries* of part (ii) of our theorem, we obtain an $O(\sqrt{m} \log c_{\max})$-approximation for profit-maximization for combinatorial auctions with *arbitrary valuations*, and an $O(\log c_{\max})$-approximation for the *non-single-minded tollbooth problem* on trees (see Section 3.1). The first result follows from the various known $O(\sqrt{m})$-approximation algorithms for the SWM problem for CAs with arbitrary valuations that also bound the integrality gap [15, 12]. For the second result, we devise a suitable $O(1)$-approximation for the SWM problem corresponding to non-SM tollbooth on trees, by adapting the randomized-rounding approach of Chakrabarty et al. [6].

Notice that with bundle-pricing, which is often used in the context of mechanism design for CAs, the profit-maximization problem becomes equivalent to the SWM problem. Thus, our results provide worst-case bounds on how item-pricing (which may be viewed as a fairness constraint on the seller) diminishes the revenue of the seller versus bundle-pricing. It is also worth remarking that our algorithms for an arbitrary valuation class (i.e., part (ii) above) can be modified in a simple way to return prices and an allocation $(S_1, \ldots, S_n)$ with the following ϵ-"one-sided envy-freeness" property while diminishing the profit by a $(1-\epsilon)$-factor (for any $\epsilon \in [0,1]$): for every non-empty S_j, the utility that j obtains from S_j is at least ϵ times the maximum utility j may obtain from any set (see Remark 2).

The only previous guarantees for limited-supply CAs with a general valuation-class are those obtained via a reduction in [2], showing that an α-approximation for the SWM problem and an algorithm for the unlimited-supply SM problem that returns profit at least OPT_{SWM}/β yield an $\alpha\beta$-approximation. A simple "grouping-by-density" approach gives $\beta = O(\log m + \log n)$; using the best known bound on β [4] yields an $O\big(\alpha(\log m + \log c_{\max})\big)$ guarantee, which is significantly weaker than our guarantees. (E.g., we obtain an $O(\alpha)$-approximation for constant $c_{\max}$.) The $O(\log c_{\max})$-factor we incur is unavoidable if one compares the profit against the optimal social welfare: a well-known example with one item, $n = c_{\max}$ customers shows a gap of $H_{c_{\max}} := 1 + \frac{1}{2} + \cdots + \frac{1}{c_{\max}}$ between the optima of the SWM- and profit-maximization problems. Almost all results for profit-maximization for CAs with non-SM valuations also compare against the optimum social welfare, so they also incur this factor. Also, it is easy to see that with $c_{\max} = 1$, the profit-maximization problem reduces to the SWM problem, so an inapproximability result for the SWM problem also yields an inapproximability result for our problem. Thus, we obtain an $m^{\frac{1}{2}}$-, or n-, inapproximability for CAs with even SM valuations (see, e.g., [10]), and *APX*-hardness for CAs with subadditive, submodular valuations, and the tollbooth problem on trees.

In Section 4, we consider an alternate approach for the non-SM highway problem that does not use OPT_{SWM} as an upper bound and achieves an (incomparable) $O(\log m)$-approximation factor. We decompose the instance via an exponential-size *configuration LP*, which is solved approximately using the ellipsoid method and rounded via randomized rounding. Here, we use LP duality to handle dependencies arising from the non-SM setting.

Theorem 2. *There is an $O(\log m)$-approximation algorithm for the non-single-minded highway problem with (i) subadditive valuations with limited supply; and (ii) arbitrary valuations with unlimited supply.*

2 Problem definition and preliminaries

The general setup of *profit-maximization* problems for (multi unit) *combinatorial auctions* (CAs) is as follows. There are n customers and m items. Let $[n] := \{1, \ldots, n\}$ and $[m] := \{1, \ldots, m\}$. Each item e is available in some limited supply or capacity c_e. Each customer j has a *valuation function* $v_j : 2^{[m]} \mapsto \mathbb{R}_+$, where $v_j(S)$ specifies the maximum amount that customer j is willing to pay for the set S; equivalently this is j's value for receiving the set S of items. We assume that $v_j(\emptyset) = 0$; we often assume for convenience that $v_j(S) \leq v_j(T)$ for $S \subseteq T$, but this monotonicity requirement is not crucial for our results. The objective is to find non-negative prices $p_e \geq 0$ for the items, and an allocation $(S_1, \ldots, S_n)$ of items to customers (where S_j could be empty) so as to maximize the total profit $\sum_{j\in[n]} \sum_{e\in S_j} p_e = \sum_{e\in[m]} p_e |\{j : e \in S_j\}|$ while satisfying the following two constraints: (i) *Budget constraints:* $p\big(S_j\big) := \sum_{e\in S_j} p_e \leq v_j\big(S_j\big)$; and (ii) *Capacity constraints:* Each element e is assigned to at most c_e customers: $|\{j \in [m] : e \in S_j\}| \leq c_e$.

As is standard in the literature on combinatorial auctions and profit-maximization problems (see, e.g., [13, 9, 3]), we assume that a valuation v is specified by a *demand oracle*, which means that given item prices $\{p_e\}$, the oracle returns a set S that maximizes the *utility* $v(S) - p(S)$. We write $c_{\max} := \max_e c_e$.

An LP relaxation. We consider a natural linear programming (LP) relaxation (P) of the SWM problem for combinatorial auctions, and its dual (D). Throughout, we use j to index customers, e to index items, and S to index sets of items.

$$\begin{aligned} \max \quad & \sum_{j,S} v_j(S) x_{j,S} & & \text{(P)} \\ \text{s.t.} \quad & \sum_{S} x_{j,S} \leq 1 & \forall j & \quad (1) \\ & \sum_{j} \sum_{S: e\in S} x_{j,S} \leq c_e & \forall e & \quad (2) \\ & x_{j,S} \geq 0 & \forall j, S & \end{aligned}$$

$$\begin{aligned} \min \quad & \sum_{e} c_e y_e + \sum_{j} z_j & & \text{(D)} \\ \text{s.t.} \quad & \sum_{e\in S} y_e + z_j \geq v_j(S) & \forall j, S & \\ & y_e, z_j \geq 0 & \forall e, j. & \end{aligned}$$

In the primal LP, we have a variable $x_{j,S}$ for each customer j and set S that indicates if j receives set S, and we relax the integrality constraints on these variables to obtain (P).

The dual (D) has variables z_j and y_e for each customer j and element e respectively, which correspond to the primal constraints (1) and (2) respectively. Although (D) has an exponential number of constraints, it can be solved efficiently given demand oracles for the valuations as these oracles yield the desired separation oracle for (D). This in turn implies that (P) can be solved efficiently. We say that an algorithm $\mathcal{A}$ for the SWM problem is an *LP-based α-approximation algorithm* for a class $\mathcal{V}$ of valuations if for every instance involving valuation functions $(v_1, \ldots, v_n)$, where each $v_j \in \mathcal{V}$, $\mathcal{A}$ returns an integer solution of value at least OPT/α. For example, the algorithm in [9] is an LP-based 2-approximation algorithm for the class of subadditive valuations.

Definition 1 *We say that an algorithm $\mathcal{A}$ for the SWM problem* "verifies" an integrality gap of (at most) α *for an LP-relaxation of the SWM problem (e.g.,*(P)*), if for* every *profile of (monotonic) valuation functions $(v_1, \ldots, v_n)$, $\mathcal{A}$ returns an integer solution of value at least $(LP\text{-}optimum)/\alpha$.*

As emphasized above, an integrality-gap-verifying algorithm must "work" for every valuation-profile. In particular, an LP-based α-approx. algorithm for a given *structured class* of valuations (e.g., submodular or subadditive valuations) *does not* verify the integrality gap for the LP-relaxation. This is precisely why our guarantee for subadditive valuations (part (i) of Theorem 1) does not follow from part (ii) of Theorem 1.

In certain cases however, one may be able to encapsulate the combinatorial structure of the SWM problem with a structured valuation class by formulating a stronger LP-relaxation for the SWM problem, and thereby prove that an approximation algorithm for the structured valuation class is in fact an integrality-gap-verifying approximation algorithm with respect to this *stronger LP-relaxation*. For example, in Section 3.1 we consider the setting where items are edges of a tree and customers desire paths of the tree. This leads to the structured valuation where $v(T) = \max\{v(P) : P \text{ is a path in } T\}$ (with $v(P) \geq 0$ being the value for path P). We design an $O(1)$-approximation algorithm for such valuations, and formulate a stronger LP for the corresponding SWM problem for which our algorithm verifies a constant integrality gap.

For a given instance $\mathcal{I} = \big(m, n, \{v_j\}_{j\in[n]}, \{c(e)\}_{e\in[m]}\big)$, our algorithms will consider different capacity vectors $k \leq c$. Let (P_k) and (D_k) denote respectively (P) and (D) with capacity-vector $k = (k_e)$, and $OPT(k)$ denote their common optimal value. Let $OPT := OPT(c)$ denote the optimum value of (P) with the original capacities. We utilize the following facts, which follow from complementary slackness, and a rounding result that follows from the work of Carr and Vempala [5], and are made explicit in [13].

Claim 1 *Let $k = (k_e)$ be any capacity-vector, and let x^* and (y^*, z^*) be optimal solutions to (P_k) and (D_k) respectively: (i) If $x^*_{j,S} > 0$, then $\sum_{e\in S} y^*_e \leq v_j(S)$; (ii) If $x^*_{j,S} > 0$, and v_j is subadditive, then $\sum_{e\in T} y^*_e \leq v_j(T)$ for any $T \subseteq S$; (iii) If $y^*_e > 0$, then $\sum_{j,S:e\in S} x^*_{j,S} = k_e$.*

Remark 1. As mentioned above, we will sometimes consider a different LP-relaxation when considering the SWM problem with a structured class of valuations. Roughly speaking, the only properties we require of this LP are that it should: (a) include a constraint similar to (2) that encodes the supply constraints; and (b) be a *packing LP*, i.e., have the form $Ax \leq b,\ x \geq 0$ where A is a nonnegative matrix. Given this, parts

(i) and (iii) of Claim 1 continue to hold with y_e denoting (as before) the dual variable corresponding to the supply constraint for item e, since the dual is then a covering LP.

Lemma 1 ([5, 13]). *Given a fractional solution x to the LP-relaxation of an SWM problem that is a packing LP (e.g., (P_k)), and a polytime integrality-gap-verifying α-approx. algorithm $\mathcal{A}$ for this LP, one can express $\frac{x}{\alpha}$ as a convex combination of integer solutions to the LP in polytime. Thus, one can round x to a random integer solution $\hat{x}$ satisfying the following "rounding property": $\frac{x_{j,S}}{\alpha} \leq \Pr[\hat{x}_{j,S} = 1] \leq x_{j,S}$ for all j, S.*

3 The main algorithm and its applications

Claim 1 leads to the simple, but important observation that if $k \leq c$ and the optimal primal solution x^* is integral, then by using $\{y^*_e\}$ as the prices, one obtains a feasible solution to the profit-maximization problem with profit $\sum_e k_e y^*_e$. There are two main obstacles encountered in leveraging this observation and turning it into an approximation algorithm. First, (P_k) will not in general have an integral optimal solution. Second, it is not clear what capacity-vector $k \leq c$ to use, e.g., $\sum_e c_e y^*_e$ could be much smaller than OPT, and in general, $\sum_e k_e y^*_e$ could be quite small for a given capacity-vector $k \leq c$. We overcome these difficulties by taking an approach similar to the one in [7].

We tackle the second difficulty by utilizing a key lemma proved by Cheung and Swamy [7], which is stated in a slightly more general form in Lemma 3 so that it can be readily applied to various profit-maximization problems. This lemma implies that one can efficiently compute a capacity-vector $k \leq c$ and an optimal dual solution (y^*, z^*) to (D_k) such that $\sum_e k_e y^*_e$ is $\big(OPT - OPT(\mathbf{1})\big)/O(\log c_{\max})$, where $\mathbf{1}$ denotes the all-ones vector (Corollary 1). To handle the first difficulty, notice that part (i) of Claim 1 implies that one can still use $\{y^*_e\}$ as the prices, provided we obtain an allocation (i.e., integer solution) $\hat{x}$ that only assigns a set S to customer j (i.e., $\hat{x}_{j,S} = 1$) if $x^*_{j,S} > 0$. (In contrast, in the envy-free setting, if we use $\{y^*_e\}$ as the prices then *every* customer j with $z^*_j > 0$, and hence $\sum_S x^*_{j,S} = 1$, must be assigned a set S with $x^*_{j,S} > 0$; this may be impossible with non-single-minded valuations, whereas this is easy to accomplish with single-minded valuations (as there is only one set per customer).) Furthermore, for subadditive valuations, part (ii) of Claim 1 shows that it suffices to obtain an allocation where $\hat{x}_{j,T} = 1$ implies that there is some set $S \supseteq T$ with $x^*_{j,S} > 0$. This is precisely what our algorithms do. We show that one can round x^* into an integer solution $\hat{x}$ satisfying the above *structural* properties, and in addition ensure that the profit obtained, $\sum_{j,T} \hat{x}_{j,T}\big(\sum_{e \in T} y^*_e\big)$, is "close" to $\sum_e k_e y^*_e$ (Lemma 4).

So if $\sum_e k_e y^*_e$ is $OPT/O(\log c_{\max})$ then applying this rounding procedure to the optimal primal solution to (P_k) yields a "good" solution. On the other hand, Corollary 1 implies that if this is not the case, then $OPT(\mathbf{1})$ must be large compared to OPT, and then we observe that an α-approximation to the SWM problem trivially yields a solution with profit $OPT(\mathbf{1})/\alpha$ (Lemma 2). In either case we obtain the desired approximation.

The algorithm is described precisely in Algorithm 1. If we use an LP-relaxation different from (P) for the SWM problem with a given valuation class that satisfies the properties stated in Remark 1, then the only change to Algorithm 1 is that we now use this LP and its dual (with the appropriate capacity-vector) instead of (P) and (D) above.

Algorithm 1 Non-single-minded profit-maximization

Input: a profit-maximization instance $\mathcal{I} = \big(m, n, \{v_j\}, \{c_e\}\big)$ with demand oracle for each v_j

1. Define $k^1, k^2, \ldots, k^\ell$ as the following capacity-vectors. Let $k^1_e = 1\ \forall e$. For $j > 1$, let $k^j_e = \min\big\{\lceil(1+\epsilon)k^{j-1}_e\rceil, c_e\big\}$; let ℓ be the smallest index such that $k^\ell = c$.
2. For each vector $k = k^j,\ j \in [\ell]$, compute an optimal solution $(y^{(k)}, z^{(k)})$ to (D_k) maximizing $\sum_e k_e y^{(k)}_e$ among all such solutions. Select $u \in \{k^1, \ldots, k^\ell\}$ maximizing $\sum_e u_e y^{(u)}_e$.
3. Compute an optimal solution $x^{(u)}$ to (P_u). Use $\mathsf{Round}(u, x^{(u)})$ to get a feasible allocation.
4. Use an LP-based α-approx. algorithm for the SWM problem (with the given valuation class) to compute an α-approx. solution to the SWM problem with unit capacities, and a pricing scheme for this allocation that yields profit equal to the social-welfare value of the allocation.
5. Return the better of the following two solutions: (1) allocation computed in step 3 with $\{y^{(u)}_e\}$ as the prices; (2) allocation and pricing scheme computed in step 4.

$\mathsf{Round}(\boldsymbol{\mu = (\mu_e), x^*})$ (x^* is an optimal solution to the SWM-LP with capacity-vector μ)

Subadditive valuations: Independently for each customer j, assign j at most one set S by choosing set S with probability $x^*_{j,S}$. If an item e gets allotted to more than μ_e customers this way, then arbitrarily select μ_e customers from among these customers and assign e to these customers. This algorithm can be derandomized via the method of conditional expectations.

General valuation class: Given an integrality-gap-verifying α-approximation algorithm (for (P_μ)), use Lemma 1 to decompose $\frac{x^*}{\alpha}$ into a convex combination $\sum_{r=1}^{\ell} \lambda_r \hat{x}^r$ of integer solutions to (P_μ). (Here $\sum_r \lambda_r = 1$ and $\lambda_r \geq 0$ for each r.) Return $\hat{x}^{(r)}$ with probability λ_r. Given item prices, this algorithm can be derandomized by choosing the solution in $\{\hat{x}^{(1)}, \ldots, \hat{x}^{(r)}\}$ achieving maximum profit.

Analysis. The analysis for both subadditive valuations and a general valuation class proceeds very similarly with the only point of difference being in the analysis of the rounding procedure (Lemma 4). First, observe that if we have an allocation $(S_1, \ldots, S_n)$ that is feasible with unit capacities, then since the sets S_j are disjoint we can charge each customer her valuation for the assigned set by pricing one of her items at this value, and hence, obtain profit equal to the social-welfare value $\sum_j v_j(S_j)$ of the allocation.

Lemma 2. *Given an LP-based α-approximation algorithm for the SWM problem with a given valuation class, one can compute a solution that achieves profit at least $\frac{OPT(\mathbf{1})}{\alpha}$.*

Lemma 3 ([7] paraphrased). *Let (C_k): $\min\ k^T y + b^T z$ s.t. $(y, z) \in \mathcal{P} \subseteq \mathbb{R}^{m+n}_+$, where $k, y \in \mathbb{R}^m_+,\ b, z \in \mathbb{R}^n_+,\ \mathcal{P} \neq \emptyset$. Let $(y^{(k)}, z^{(k)})$ be an optimal solution to (C_k) that maximizes $k^T y$ among all optimal solutions, and $\mathsf{opt}(k)$ denote the optimal value. Let $k^1, \ldots, k^\ell$, and u be as defined in steps 1 and 2 respectively of Algorithm 1. Then, $\sum_e u_e y^{(u)}_e \geq \big(\mathsf{opt}(c) - \mathsf{opt}(\mathbf{1})\big)/\big(2(1+\epsilon)H_{c_{\max}}\big)$.*

Corollary 1. *The capacity-vector u computed in step 2 of Algorithm 1 satisfies the inequality $\sum_e u_e y^{(u)}_e \geq \big(OPT(c) - OPT(\mathbf{1})\big)/\big(2(1+\epsilon)H_{c_{\max}}\big)$.*

Lemma 4. *Let $\hat{x}$ be the integer solution returned by* Round *in step 3 of Algorithm 1. Then $\hat{x}$ combined with the pricing $y^{(u)}$ is a feasible solution to the profit-maximization problem with probability 1, which achieves expected profit at least (i) $\big(1-\frac{1}{e}\big)\sum_e u_e y^{(u)}_e$ for subadditive valuations; and (ii) $\sum_e u_e y^{(u)}_e/\alpha$ for a general valuation class.*

Theorem 3. *Algorithm 1 runs in time* $\text{poly}\big(\textit{input size}, \frac{1}{\epsilon}\big)$ *and achieves an*

(i) $O(\log c_{\max})$*-approximation for subadditive valuations, using the 2-approximation algorithm for the SWM problem with subadditive valuations in [9];*
(ii) $O(\alpha \log c_{\max})$*-approximation for a general valuation class given an integrality-gap-verifying* α*-approximation algorithm for the SWM problem.*

Remark 2. Note that if the allocation $(S_1, \ldots, S_n)$ returned by Algorithm 1 is obtained via Round, then S_j is always a subset of a *utility-maximizing set* of j, and with a general valuation class, if $S_j \neq \emptyset$, it is a utility-maximizing set (under the computed prices). Also, if $(S_1, \ldots, S_n)$ is obtained in step 4, then we may assume that $v_j(S_j) > v_j(S_j \setminus \{e\})$ for all $e \in S_j$; moreover, with a general valuation class, this solution can be modified to yield an approximate "one-sided envy-freeness" property. We compute $(S_1, \ldots, S_n)$ by rounding $x^{(1)}$ as described in Lemma 1. Now choose prices $\{p'_e\}$ (arbitrarily) such that $p' \geq y^{(1)}$ and $p'(S_j) = \max\{y^{(1)}(S_j), (1-\epsilon)v_j(S_j)\}$ for every j. Since any non-empty S_j is a utility-maximizing set under $y^{(1)}$, it follows that (a) p' is a valid item-pricing yielding profit at least $(1-\epsilon)\sum_j v_j(S_j)$; (b) if $S_j \neq \emptyset$, then the utility j derives from S_j under p' is at least ϵ(max utility of j under p').

3.1 Applications

Arbitrary valuation functions. The integrality gap of (P) is known to be $\Theta(\sqrt{m})$, and there are efficient (deterministic) algorithms that verify this integrality gap [15, 12]. So Theorem 3 immediately yields an $O(\sqrt{m}\log c_{\max})$-approximation algorithm for the profit-maximization problem for combinatorial auctions with arbitrary valuations.

Non-single-minded tollbooth problem on trees. In this profit-maximization problem, items are *edges* of a tree and customers desire paths of the tree. More precisely, let $\mathcal{P}$ denote the set of all paths in the tree (including $\emptyset$). Each customer j has a value $v_j(S) \geq 0$ for path $S \in \mathcal{P}$, and may be assigned any (one) path of the tree. This leads to the *structured* valuation function $v_j : 2^{[m]} \mapsto \mathbb{R}_+$ where $v_j(T) = \max\{v_j(S) : S \text{ is a path in } T\}$. We use Algorithm 1 to obtain an $O(\log c_{\max})$-approximation guarantee by formulating an LP-relaxation of the SWM problem that is tailored to this setting and designing an $O(1)$-integrality-gap-verifying algorithm for this LP.

The "new" LP is almost identical to (P), except that we now *only have variables* $x_{j,S}$ *for* $S \in \mathcal{P}$. Correspondingly, in the dual (D), we only have a constraint for (j, S) when $S \in \mathcal{P}$. Clearly, this new LP satisfies the properties stated in Remark 1, so parts (i) and (iii) of Claim 1 hold for this new LP, and so does Lemma 1. Thus, we only need to design an $O(1)$-integrality-gap-verifying algorithm for this new LP to apply Theorem 3. Let $\{v_j : \mathcal{P} \mapsto \mathbb{R}_+\}_{j\in[n]}$ be any instance and x^* be an optimal solution to this new LP for this instance. We design a randomized algorithm that returns a (random) integer solution $\hat{x}$ of expected objective value $\Omega(\sum_{j,S\in\mathcal{P}} v_j(S)x^*_{j,S})$. This algorithm can be derandomized using the work of [16]; this yields an $O(1)$-integrality-gap-verifying algorithm for the new LP. Our algorithm is a generalization of the one proposed by [6] for unsplittable flow on a line. Root the tree at an arbitrary node. Define the *depth* of an edge (a, b) to be the minimum of the distances of a and b to the root. Define the depth of an edge-set T to be the minimum depth of any edge in T. Let $\alpha = 0.01$.

1. Independently, for every customer j, choose at most one path S, by picking S with probability $\alpha x^*_{j,S}$. Let S_j be the set assigned to j. (If j is unassigned, then $S_j = \emptyset$.)
2. Let $W = \emptyset$. Consider the sets $\{S_j\}$ in non-decreasing order of their depth (breaking ties arbitrarily). For each set $T = S_j$, if T can be added to $\{S_i : i \in W\}$ without violating any capacities, add j to W, otherwise discard T.

Let $\hat{x}$ be the (random) integer solution computed. Using a similar argument as in [6], we can prove that $\Pr[\hat{x}_{j,S} = 1] \geq 0.004 x^*_{j,S}$, so $\mathrm{E}\big[\sum_{j,S\in\mathcal{P}} v_j(S)\hat{x}_{j,S}\big] \geq 0.004 \cdot \sum_{j,S\in\mathcal{P}} v_j(S) x^*_{j,S}$. We thus obtain the following theorem as a corollary of Theorem 3.

Theorem 4. *There is an $O(1)$-integrality-gap-verifying algorithm for the above LP, and thus an $O(\log c_{\max})$-approx. algorithm for the non-SM tollbooth problem on trees.*

Since the above algorithm satisfies the rounding property in Lemma 1, we can use it to round $x^{(u)}$ (more efficiently) to a feasible allocation in step 3 of Algorithm 1, instead of using the Carr-Vempala procedure (which relies on the ellipsoid method).

4 Refinement for the non-single-minded highway problem

In this section, we describe a different approach that does not use OPT_{SWM} as an upper bound on the optimum profit. Instead our approach is based on using an exponential-size *configuration LP* to decompose the original instance into various smaller (and easier) instances. We use this to obtain an $O(\log m)$-approx. for the non-SM highway problem with subadditive valuations, and arbitrary valuations but unlimited supply (Theorem 2).

Let $\mathcal{P}$ be the set of all intervals on the line (with m edges). As with the non-SM tollbooth problem on trees, each customer j has a value for each subpath (which is now an interval). So we view v_j as a function $v_j : \mathcal{P} \mapsto \mathbb{R}_+$, and *subadditivity* means that $v_j(A \cup B) \leq v_j(A) + v_j(B)$ for any two intervals A, B, where $A \cup B$ is also an interval.

We sketch the proof of Theorem 2. First, we use a standard decomposition to partition the intervals in $\mathcal{P}$ into $O(\log m)$ disjoint sets, where each set is a union of item-disjoint "pyramids". A pyramid is a set of paths that share a common edge; two pyramids $\mathcal{P}_1$ and $\mathcal{P}_2$ are item-disjoint, if $A \cap B = \emptyset$ for all $A \in \mathcal{P}_1,\ B \in \mathcal{P}_2$. Thus, to get an $O(\log m)$-approximation algorithm, it suffices to give an $O(1)$-approximation algorithm when the intervals form a union of item-disjoint pyramids. It is unclear how to achieve a near-optimal solution even in this structured setting, as there are various *dependencies between the pyramids* in a set: a customer can only be assigned an interval in *one* of the pyramids. We solve this "union-of-pyramids" pricing problem as follows. We first trim each pyramid $\mathcal{P}_i$ in our set randomly to a one-sided half-pyramid $\mathcal{H}_i$ by (essentially) ignoring the items to the left or right of the common edge of $\mathcal{P}_i$. The details of this truncation are slightly different depending on whether we have subadditive or arbitrary valuations, but a key observation is that, in expectation, we only lose a factor of 2 by this truncation. We formulate an LP-relaxation for the pricing problem involving these half-pyramids. Let $\mathcal{R}_i$ denote the set of all possible solutions for $\mathcal{H}_i$, where a solution specifies a pricing of the intervals in $\mathcal{H}_i$ (rounded to the nearest power of 2) and an allocation of intervals to customers satisfying the budget and capacity constraints. We introduce a variable $y_{jp} \geq 0$ for each customer j and price p denoting if customer

j buys a path at price p, and a variable $x_{i,R}$ for each $R \in \mathcal{R}_i$ denoting whether solution R has been chosen for $\mathcal{H}_i$. Let $p_j(R)$ be the price that j pays under the solution R, and $\mathcal{R}_{i,j,p} = \{R \in \mathcal{H}_i : p_j(R) = p\}$ be the set of solutions for $\mathcal{H}_i$ where j pays price p. We consider the following LP: $\max \sum_{j,p} p \cdot y_{jp}$ s.t. $\sum_{R \in \mathcal{R}_i} x_{i,R} = 1 \ \forall i, \ \ \sum_p y_{jp} \leq 1 \ \forall j$, $\sum_{i,R:R \in \mathcal{R}_{i,j,p}} x_{i,R} \geq y_{jp} \ \ \forall j, p$, and $x_{i,R}, y_{jp} \geq 0 \ \ \forall i, R, j, p$. We solve this LP using the ellipsoid method on the dual problem; the separation oracle is provided by the solution to an easier pricing problem, where the *half-pyramids are now decoupled*. We devise an algorithm based on dynamic programming to compute a near-optimal solution to this pricing problem, which then yields a near-optimal solution to the LP. Finally, we argue that this solution can be rounded to an integer solution losing only an $O(1)$-factor. This gives us the desired $O(1)$-approx. for the "union-of-pyramids" pricing problem, which in turn yields an $O(\log m)$-approx. for our original non-SM highway problem.

Lemma 5. *There is a $16(1 + \frac{1}{m})$-approx. algorithm for the non-SM highway problem when intervals form a union of item-disjoint pyramids for (i) subadditive valuations with limited supply; (ii) arbitrary valuations with unlimited supply.*

References

1. Aggarwal, G., Feder, T., Motwani, R., Zhu, A.: Algorithms for multi-product pricing. In: ICALP. pp. 72–83 (2004)
2. Aggarwal, G., Hartline, J.D.: Knapsack auctions. In: SODA. pp. 1083–1092 (2006)
3. Balcan, M.F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: EC. pp. 50–59 (2008)
4. Briest, P., Krysta, P.: Single-minded unlimited supply pricing on sparse instances. In: SODA. pp. 1093–1102 (2006)
5. Carr, R., Vempala, S.: Randomized metarounding. Random Structures and Algorithms 20(3), 343–352 (2002)
6. Chakrabarti, A., Chekuri, C., Gupta, A., Kumar, A.: Approximation algorithms for the unsplittable flow problem. Algorithmica 47 (2007)
7. Cheung, M., Swamy, C.: Approximation algorithms for single-minded envy-free profit-maximization problems with limited supply. In: FOCS. pp. 35–44 (2008)
8. Dobzinski, S., Schapira, M.: An improved approximation algorithm for combinatorial auctions with submodular bidders. In: SODA. pp. 1064–1073 (2006)
9. Feige, U.: On maximizing welfare when utility functions are subadditive. In: STOC. pp. 41–50 (2006)
10. Grigoriev, A., van Loon, J., Sitters, R., Uetz, M.: How to sell a graph: Guidelines for graph retailers. In: WG. pp. 125–136 (2006)
11. Guruswami, V., Hartline, J.D., Karlin, A.R., Kempe, D., Kenyon, C., McSherry, F.: On profit-maximizing envy-free pricing. In: SODA. pp. 1164–1173 (2005)
12. Kolliopoulos, S.G., Stein, C.: Approximating disjoint-path problems using packing integer programs. Math. Program. 99(1), 63–87 (2004)
13. Lavi, R., Swamy, C.: Truthful and near-optimal mechanism design via linear programming. In: FOCS (2005)
14. Lehmann, B., Lehmann, D.J., Nisan, N.: Combinatorial auctions with decreasing marginal utilities. In: EC. pp. 18–28 (2001)
15. Raghavan, P.: Probabilistic construction of deterministic algorithms: approximating packing integer programs. J. Comput. Syst. Sci. 37(2), 130–143 (1988)
16. Sivakumar, D.: Algorithmic derandomization via complexity theory. In: STOC. pp. 619–626 (2002)