



# On the Complexity of Stackelberg Matroid Pricing Problems

Toni Böhnlein<sup>1</sup>(✉) and Oliver Schaudt<sup>2</sup>

<sup>1</sup> Faculty of Engineering, Bar Ilan University, 52900 Ramat-Gan, Israel  
toni.bohnlein@biu.ac.il

<sup>2</sup> Institut für Informatik, Universität zu Köln, Weyertal 80, 50321 Cologne, Germany  
schaudto@uni-koeln.de

**Abstract.** In a Stackelberg pricing problem a distinguished player, the *leader*, chooses prices for a set of items, and one or several other players, the *followers*, seeks to buy a feasible subset of the items with minimal costs. The leader's goal is to maximize her revenue, which is determined by the sold items and their prices.

We are interested in cases where the followers' feasible subsets are given by a combinatorial optimization problem. For example, a pricing problem based on the shortest path problem was used by Labbé et al. [15] to model road-toll setting scenarios.

In this paper, we consider Stackelberg pricing problems that are based on matroids. The followers seek to buy a subset that is a basis. More specifically, we consider uniform, partition and laminar matroids.

We study the complexity of computing leader-optimal prices for a single and multiple followers. We show that optimal prices can be computed in polynomial time for all three matroids if there is one follower. In general, such pricing problems based on matroids are APX-hard (see [11]).

For multiple followers, we show that computing optimal prices for uniform matroids can be done in polynomial time. However, for partition and laminar matroids the pricing problem becomes NP-hard.

**Keywords:** Algorithmic pricing · Stackelberg games · Revenue maximization · Matroids

## 1 Introduction

We study pricing problems in a game-theoretic model known as Stackelberg games or *Stackelberg pricing problems*. In this model, one player, the *leader*, chooses prices for a number of items and one or several other players, the *followers*, are interested in buying subsets of the items. The followers buy subsets that minimize their expenses subject to some constraints while the leader's goal is to maximize her revenue, which is determined by the sold items and their prices. We are interested in the complexity of computing leader-optimal prices depending on the constraints of the followers.

One line of research studies Stackelberg pricing problems where the followers' constraints are given by a combinatorial optimization problem. A well-motivated example of such a pricing problem was introduced by Labbé et al. [15] to compute optimal road-tolls: A road network is modeled by a graph where the edges have costs that have to be paid when traveling along an edge. A subset of the roads or edges belongs to the leader and she can charge a toll which increases their costs. Each follower is given by two nodes  $s$  and  $t$  in the graph and chooses a minimal-cost path connecting  $s$  and  $t$ . The leader gains revenue in the amount of paid tolls. When deciding on the tolls, the leader has to make the following consideration: On the one hand, low tolls might fail to produce maximum revenue. On the other hand, a large toll might cause the followers to avoid a road entirely resulting in zero revenue. Since the followers "buy" a shortest path, this variant is called *Stackelberg shortest path*.

*Stackelberg minimum spanning tree* was analyzed by Cardinal et al. and Bilò et al. [3, 11, 12]. The followers are again interested in subsets of a graph's edges, but the subsets have to form a spanning tree. This setting has applications, for example, when an internet service provider wants to connect hubs in a network. The leader charges additional costs for some of the edges and collects revenue if they are used by a follower (internet service provider). Moreover, *Stackelberg interval scheduling* models situations where the leader pays the follower to execute a set of jobs. The leader makes 'make or buy'-type decisions (see [6]).

Intuitively, the complexity of the pricing problem depends on the complexity of the followers' optimization problem. Stackelberg shortest path is hard to approximate within a factor of less than 2 (see Briest et al. [7]) and Stackelberg minimum spanning tree was shown to be APX-hard. However, Stackelberg interval scheduling is solvable in polynomial time.

**Our Results.** We study Stackelberg pricing problems that are based on matroids. A matroid is a family of subsets over a ground set that is subject to a set of constraints. The constraints are a bit technical and we spare them for the next section. As an example, for a matroid, we can think of the subsets of a graph's edges that are acyclic. This matroid is called the *graphic* matroid. The inclusion-wise maximal subsets of a matroid are its *bases*. If the elements of the ground set are associated with weights, then a minimal weight basis can be computed by the greedy algorithm. The minimal weight bases of a graphic matroid are the minimum weight spanning trees of the associated graph.

For a Stackelberg pricing problem based on a matroid, the ground set is partitioned into two blocks. One block contains items<sup>1</sup> that have *fixed-costs*. Think of these fixed-cost items as being offered by the leader's competitors. The second block contains the *priceable* items for which the leader chooses prices. Each follower comes equipped with a matroid over the ground set and is interested in buying a minimum weight basis. The weight of an item is either its fixed-cost or its price. Given prices, the decision which subset a follower buys can be computed by the greedy algorithm. If a follower buys a basis, the leader gains

<sup>1</sup> We call the element of the ground set items.

revenue for each contained priceable item in the amount of its price. The leader's goal is to maximize her revenue.

Finding a minimum weight basis of a matroid can be regarded as a rather simple problem since it can be done with the greedy algorithm. Therefore, it is surprising that the pricing problem based on the graphic matroid, Stackelberg minimum spanning tree, is APX-hard. To find cases that can be solved in polynomial time, we have to resort to even simpler matroids.

We study three different classes of matroids in two scenarios. For the simpler scenario, we assume that there is a *single* follower. Note that this scenario was studied in most of the literature on Stackelberg pricing problems so far. For the second scenario, there are *multiple* followers which implies a few questions regarding the availability of items and coordination between followers. We assume that the items are available in unlimited supply which makes coordination between the followers unnecessary. The leader sets one price for each item that is valid for all the followers.

The first class are *uniform* matroids. Here, followers are interested in buying a set of items that has a given size. Different followers may come with different sizes. There is no additional structure on the items and a follower buys a subset of his given size with minimum total weight. We show that leader-optimal prices can be computed in polynomial time for a single and the multiple followers. While the single follower scenario is quite simple, the multiple followers scenario requires a dynamic programming approach.

Second, we consider *partition* matroids which generalize uniform matroids. A follower is associated with a partition of the ground set into blocks. For each block, the follower buys a subset of a given size. Different followers may be associated with different partitions and sizes. We show that computing leader-optimal prices for a single follower can be done in polynomial time. For the multiple followers, this computational task is NP-hard.

**Table 1.** Summary of our results on matroid based Stackelberg pricing problems. The results on Stackelberg minimum spanning tree (MST) appear in [11].

	Single follower	Multiple followers
Uniform	poly-time	poly-time
Partition	poly-time	NP-hard
Laminar	poly-time	NP-hard
MST	APX-hard	APX-hard

The third class are *laminar* matroids which generalize partition matroids. A laminar matroid is based on a hierarchical family of subsets of the ground set, i.e., two subsets of the family are either disjoint or one is contained in the other. Such a hierarchical family is also called a laminar family. For each of the subsets of the laminar family, a follower has an upper bound on the number of items

that he wants to buy from this subset. Finding leader-optimal prices has the same complexity for laminar matroids as it has for partition matroids.

Table 1 summarizes our results. The organization of the paper is as follows. In the next section, we give a more careful definition of matroids and Stackelberg pricing problems. In Sects. 3 and 4, we show how to solve Stackelberg uniform matroid with multiple followers and Stackelberg laminar matroid with one follower, respectively. Section 5 shows that Stackelberg partition matroid is hard with multiple followers. For the missing proofs, we refer to the full version of the paper. Finally, Sect. 6 discusses directions for future research.

**Related Work.** Additional literature includes surveys on Stackelberg shortest path by van Hoesel [20] and Labbé and Violin [16]. Roche et al. [18] present an algorithm with logarithmic approximation guarantee. The best lower bound is due to Briest et al. [7] showing approximation hardness within a factor of less than 2. This is an improvement over APX-hardness by Joret [13].

A Stackelberg shortest path tree game was studied by Bilo et al. [4] and Cabello [10]. Briest et al. [9] give a polynomial time algorithm for Stackelberg bipartite vertex cover game which was later improved by Baïou and Barahona [1].

Briest et al. [9] give a  $\log(k)$  approximation algorithm for Stackelberg pricing games where  $k$  is the number of items. Independently, a slightly more general result was obtained by Balcan et al. [2]. Their algorithms use a single price strategy which was studied in a more general setting by Böhnlein et al. [5].

Briest et al. [8] study Stackelberg pricing games where the follower is based on a NP-hard optimization problem and runs a known approximation algorithm.

## 2 Preliminaries

**Stackelberg Pricing Problems.** Let  $E = E_f \dot{\cup} E_p$  be a finite set of items which consists of two blocks  $E_f$  and  $E_p$ .  $E_f$  contains the *fixed-cost* items and  $E_p$  contains the *priceable* items. Let

$$|E_f| = m \quad \text{and} \quad |E_p| = n.$$

The items in  $E_f$  have costs given by the function  $c : E_f \rightarrow \mathbb{R}$ .

We have one *leader* and  $\ell$  *followers*, for an integer  $\ell \geq 1$ . The leader seeks to sell the items in  $E_p$  to the followers. But the followers can also buy items in  $E_f$  paying their costs  $c$ . The leader choose prices by specifying a price function  $p : E_p \rightarrow \mathbb{R}$ . From the followers' perspective, we do not distinguish between priceable and fixed-cost items. Hence, given a price function  $p$ , we compose a weight function  $w : E \rightarrow \mathbb{R}$ :

$$w(e) = \begin{cases} c(e), & e \in E_f, \\ p(e), & e \in E_p. \end{cases}$$

Each follower  $i$  is determined by a family of *feasible* subsets  $\mathcal{S}_i \subseteq 2^E$  which contains the subsets that he is interested in buying. Given a price function  $p$ , the weight of a subset  $S \in \mathcal{S}_i$  is defined as

$$w(S, p) = \sum_{e \in S} w(e).$$

The objective of the follower is to buy a feasible subset with minimum total weight which is  $w_i^*(p) = \min_{S \in \mathcal{S}_i} w(S, p)$ . A subset  $S \subseteq E$  (if bought by a follower) yields *revenue* for the leader:

$$\text{rev}(S) = \sum_{e \in S \cap E_p} p(e).$$

In case there are several feasible subsets of weight  $w_i^*(p)$ , we assume that followers are optimistic and buy a subset that yields maximum revenue for the leader. Hence, follower  $i$  buys the following feasible subset:

$$S_i^*(p) = \arg \max_{S \in \mathcal{S}_i} \{\text{rev}(S) : w(S, p) = w_i^*(p)\}. \quad (1)$$

The revenue from follower  $i$  is  $\text{rev}(S_i^*(p))$  and the leader's total revenue is

$$\text{rev}(p) = \sum_{i=1}^{\ell} \text{rev}(S_i^*(p)).$$

The leader's objective is to determine a price function  $p$  that maximizes  $\text{rev}(p)$ .

A follower's decision is the solution to an optimization problem (given a price function). When the leader decides on the prices, she is aware of the fixed-cost items and their costs as well as the followers' objective functions and feasible subsets, i.e., we are in a full information setting. Moreover, we assume that each follower has a feasible subset that does not contain any priceable items; otherwise, the leader's revenue is unbounded. If there are multiple followers, we assume that items are available in unlimited supply.

#### STACKELBERG PRICING

**Input:** A ground set  $E = E_f \cup E_p$ , a cost function  $c : E_f \rightarrow \mathbb{R}$ , and  $\ell$  followers given by families  $\mathcal{S}_i \subseteq 2^E$ , for  $i \in [\ell]$ .

**Objective:** Find prices  $p : E_p \rightarrow \mathbb{R}$  maximizing  $\text{rev}(p)$ .

Note that STACKELBERG PRICING captures the problems mentioned in the introduction.

**Matroids.** Given a ground set  $E$ , a family of subsets  $\mathcal{S} \subset 2^E$  is a *matroid* if it satisfies the following conditions:

(M1)  $\emptyset \in \mathcal{S}$ .

(M2) If  $X \subseteq Y \in \mathcal{S}$ , then  $X \in \mathcal{S}$ .

(M3) If  $X, Y \in \mathcal{S}$  and  $|X| > |Y|$ , there exists  $x \in X \setminus Y$  such that  $Y \cup x \in \mathcal{S}$ .

Matroids are a well-studied combinatorial structure (cf. [17]). The *bases* of a matroid  $\mathcal{S}$  are its inclusion-wise maximal elements. For example, the acyclic subsets of the edges of a graph  $G$  form a matroid. It is called the graphic matroid and its bases are the spanning forests of  $G$ .

Given weights  $w : E \rightarrow \mathbb{R}$  on the ground sets, a minimum weight basis  $B$  can be computed using a greedy algorithm: To compute  $B$ , we (starting with  $B = \emptyset$ ) consider the elements of  $E$  sorted by their weights in non-decreasing order and add an element  $e$  to  $B$  if  $B \cup e \in \mathcal{S}$ .

**Stackelberg Matroid** is an instance of STACKELBERG PRICING where followers are given by a matroid and buy a minimum weight basis. Cardinal et al. [11] show that STACKELBERG PRICING based on the graphic matroid is APX-hard.

**Theorem 2.1 (Cardinal et al. [11]).** *STACKELBERG MATROID with one follower is APX-hard.*

Cardinal et al. observe that an optimal price function uses only values that appear as fixed-costs in  $c$ . Given an instance of STACKELBERG MATROID and a price function  $p$ , then  $E = \{e_1, \dots, e_{m+n}\}$  are the elements of  $E$  sorted non-decreasingly by their weights  $w$ . If  $w(e_j) = w(e_i)$  where  $e_j \in E_f$  and  $e_i \in E_p$ , then  $i < j$ . Hence, the optimistic follower computes his solution greedily based on this order. If  $p$  assigns a price that is not a fixed-cost, increasing this price to the next larger fixed-cost does not change the ordering but increases the leader's revenue. This observation also holds if there are multiple followers.

**Lemma 2.1 (Cardinal et al. [11]).** *There is an optimal price function that uses only values of the cost function  $c$ .*

We close this section with some more notation. Let  $E_f = \{g_1, \dots, g_m\}$  be the elements of  $E_f$  sorted non-decreasingly by their costs. Shorthand, we write  $c(g_i) = c_i$  for the costs of  $g_i \in E_f$ . Similarly, given a price function  $p$ , let  $E_p = \{h_1, \dots, h_n\}$  be the elements of  $E_p$  sorted non-decreasingly by their price. Usually, the price function is clear from the context and we write  $p(h_i) = p_i$ .

### 3 Uniform Matroid

In light of Theorem 2.1, we consider an arguably simple class of matroids, namely *uniform* matroids. Given a ground set  $E$  and an integer  $s \geq 1$ , the uniform matroid  $\mathcal{S}$  (of rank  $s$ ) contains all subsets of  $E$  that have size at most  $s$ . Formally,

$$\mathcal{S} = \{U \subseteq E : |U| \leq s\}.$$

The bases of the uniform matroid are the subsets of size exactly  $s$ .

**Stackelberg Uniform Matroid with One Follower.** Let the follower be based on a uniform matroid of rank  $s$ . Since  $E$  has no structure, the follower only cares about the weight of an element. Given a price function  $p$ , the follower buys the set  $S^*(p) = \{e_1, \dots, e_s\}$ .

With the next lemma we analyze the conditions when the follower buys a fixed number  $a \in \mathbb{N}$  of the priceable items, for  $a \leq \min\{s, n\}$ .

**Lemma 3.1.** *Let  $a \leq \min\{s, n\}$ . Given a price function  $p$ ,  $|S^*(p) \cap E_p| = a$  if and only if  $p_a \leq c_{s-a+1}$  and  $c_{s-a} < p_{a+1}$ .*

*Proof.* First, assume that  $|S^*(p) \cap E_p| = a$ . It follows that  $|S^*(p) \cap E_f| = s - a$ ,  $g_{s-a+1} \notin S^*(p)$ ,  $g_{s-a} \in S^*(p)$ , and that  $h_a \in S^*(p)$ . Hence, we must have that  $p_a \leq c_{s-a+1}$  and  $c_{s-a} < p_{a+1}$ .

Now, assume that  $p_a \leq c_{s-a+1}$  and  $c_{s-a} < p_{a+1}$ . It follows that  $|S^*(p) \cap E_p| \geq a$  and that  $|S^*(p) \cap E_f| \geq s - a$ . Consequently,  $|S^*(p) \cap E_p| = a$ .  $\square$

To gain maximum revenue when selling  $a$  items, the leader chooses the largest prices that satisfy the conditions of Lemma 3.1. It follows that an optimal price function assigns prices  $c_{s-a+1}$  for  $a$  many items while the prices of the remaining items must be larger. The revenue of such a price function is  $a \cdot c_{s-a+1}$ , and the maximum revenue  $\text{rev}^*$  can be computed as follows:

$$\text{rev}^* = \max_{a \in [\min\{s, n\}]} a \cdot c_{s-a+1}.$$

Böhnlein et al. [5] showed that constant functions are optimal for STACKELBERG UNIFORM MATROID. The target values are the values of  $c$ .

**Stackelberg Uniform Matroid with  $\ell \geq 2$  Followers.** Each follower  $i$  is determined by its ranks  $s_i \in \mathbb{N}$ . Without loss of generality,  $s_\ell \leq \dots \leq s_1$ . For a given price function  $p$ , follower  $i$  buys the set  $S_i^*(p) = \{e_1, \dots, e_{s_i}\}$ . It follows that  $S_\ell^*(p) \subseteq \dots \subseteq S_1^*(p)$  and that an item  $e \in S_i^*(p)$  is bought by  $i$  many followers.

**Observation 1.** *If  $|S_i^*(p) \cap E_f| = a$ , then the leader sells  $a$  items at least  $i$  times. Moreover,  $|S_i^*(p) \cap E_f| = a$  if and only if  $p_a \leq c_{s_i-a+1}$  and  $c_{s_i-a} < p_{a+1}$ .*

The second part of Observation 1 follows from Lemma 3.1.

To convince ourselves that an optimal price function does not have to be constant when there are more than one followers, we consider a small example. Assume that there are 4 items with fixed-costs  $c_1 = 3$  and  $c_{2/3/4} = 5$  as well as 4 priceable items. We have two followers of rank 1 and 4, respectively. Verify that the constant price functions with values 3 and 5 each yield a revenue of 15. But the price function that assigns prices  $(3, 5, 5, 5)$  yields a revenue of 16.

From the small example, we get the intuition that an optimal price function  $p$  can have several steps (assuming a non-decreasing ordering of the function's values). To determine these steps or equivalently the step-lengths, we use dynamic programming. We construct an algorithm based on solving elementary cases, in

which the leader sells  $a$  many priceable items  $i+1$  times and  $a+b$  many priceable items  $i$  times. The set of all price functions that satisfy these conditions is

$$\mathcal{P}_{a,b}^i = \{p : E_p \rightarrow \mathbb{R} : |S_{i+1}^* \cap E_p| = a \text{ and } |S_i^* \cap E_p| = a + b\},$$

for  $a, b \in \mathbb{N}$ . With the next lemma we characterize optimal price functions under these conditions.

**Lemma 3.2.** *Let  $i \in [\ell]$  and  $a, b \in \mathbb{N}$  such that  $a + b \leq n$ ,  $a \leq s_{i+1}$  and  $b \leq s_i - s_{i+1}$ . A price function  $p \in \mathcal{P}_{a,b}^i$  is optimal, if*

$$p_{a+1} = \dots = p_{a+b} = c_{s_i - (a+b) + 1}.$$

*Proof.* Let  $p \in \mathcal{P}_{a,b}^i$  be an optimal price function. With Observation 1 we have that  $p_{a+1} \leq \dots \leq p_{a+b} \leq c_{s_i - (a+b) + 1}$ . Suppose towards a contradiction that  $p_{a+1} < c_{s_i - (a+b) + 1}$ . But this implies that  $p_{a+1}$  can be increased to  $c_{s_i - (a+b) + 1}$  without changing the followers' decision and increasing the leader's revenue, contradicting that  $p$  was optimal.  $\square$

For  $i \leq \ell$  and  $a \leq s_{i+1}$ , we compute recursively the maximum revenue that the items  $h_{a+1}, \dots, h_n$  can yield under the conditions of  $\mathcal{P}_{a,b}^i$ .

**Definition 3.1.** *Let  $i \leq \ell$  and  $a, b \in \mathbb{N}$  such that  $a \leq \min\{s_{i+1}, n\}$  and  $b \leq \min\{s_i - s_{i+1}, n - i\}$ . The maximum revenue that items  $h_{a+1}, \dots, h_n$  can yield under a price function  $p \in \mathcal{P}_{a,b}^i$  is*

$$\lambda_{a,b}^i = \max_{p \in \mathcal{P}_{a,b}^i} \sum_{k=1}^i \text{rev}(S_k^*(p) \setminus S_{i+1}^*(p)).$$

Note that  $\mathcal{P}_{a,b}^i$  can be empty, and for a sound definition max is replaced by sup. For a practical algorithm, we define  $\lambda_{a,b}^i = -\infty$  in these cases. Moreover, we set  $S_{\ell+1}^* = \emptyset$  and  $s_{\ell+1} = 0$ . The main technical feat of this section is the next lemma which derives a recursive formula for  $\lambda_{a,b}^i$ .

**Lemma 3.3.**  $\lambda_{a,b}^i = \max_{\theta \in [\min\{n - (a+b), s_{i-1} - s_i\}]} \lambda_{a+b, \theta}^{i-1} + b \cdot i \cdot c_{s_i - (a+b) + 1}$

Following Lemma 3.2, the base cases for the dynamic program ( $i = 1$ ) can be determined as follows:  $\lambda_{a,b}^1 = b \cdot c_{s_1 - (a+b) + 1}$ . Lemma 3.3 allows us to compute the values of  $\lambda_{a,b}^i$  (for  $i \geq 2$ ).

**Lemma 3.4.** *The leader's maximum revenue can be computed as follows:*

$$\max_p \text{rev}(p) = \max_{b \leq \min\{s_\ell, n\}} \lambda_{0,b}^\ell.$$

The lemmas above imply an algorithm to compute the leader's optimal revenue for STACKELBERG UNIFORM MATROID when there are multiple followers. The running time of the algorithm is of order  $\mathcal{O}(\ell \cdot n^3)$ . Hence, the main result of this section is as follows.

**Theorem 3.1.** STACKELBERG UNIFORM MATROID *with multiple followers can be solved in polynomial time.*



## 4 Laminar Matroid

We continue with positive results showing that STACKELBERG MATROID based on laminar matroids can be solved in polynomial time if there is only one follower.

Let  $E$  be a ground set. Then,  $\mathcal{F} \subseteq 2^E$  is a *laminar family* if for all  $U, V \in \mathcal{F}$ , either  $U \subseteq V$ ,  $V \subseteq U$  or  $U \cap V = \emptyset$ . Each element of  $\mathcal{F}$  has a *capacity*  $\varphi : \mathcal{F} \rightarrow \mathbb{N}$ . The *laminar matroid*  $\mathcal{S}$  based on  $\mathcal{F}$  and  $\varphi$  is defined as follows

$$\mathcal{S} = \{W \subseteq E : |W \cap U| \leq \varphi(U) \text{ for all } U \in \mathcal{F}\}.$$

Without loss of generality, we assume that  $E \in \mathcal{F}$ . In case  $E \notin \mathcal{F}$ , we add  $E$  to  $\mathcal{F}$  and set  $\varphi(E) = |E|$  without changing the matroid.

Let  $N, M \in \mathcal{F}$  such that  $N \subset M$ . If there is no subset  $T \in \mathcal{F}$  such that  $N \subset T \subset M$ ,  $N$  is a *direct* subset of  $M$ . Let  $D(M)$  be the set of  $M$ 's direct subsets. If  $D(M) = \emptyset$ ,  $M$  is *minimal*. Observe that laminar families have a hierarchical structure. We can associate a laminar family with a rooted tree where we identify the nodes of the tree with the elements of the laminar family. Set  $E$  is the root. The children of a node are its direct subsets and the minimal subsets are leaves.

We say that  $\mathcal{F} = \{M_1, \dots, M_s\}$  is a *topological* order of  $\mathcal{F}$  if for  $M_i \subseteq M_j$  it follows that  $i > j$ . For  $i \in [s]$ , let  $\mathcal{F}_i = \{M_j \in \mathcal{F} : j \geq i\}$ . Observe that  $\mathcal{F}_i$  is again a laminar family on  $E$ . Moreover, we define  $\mathcal{S}_i$  to be the laminar matroid based on  $\mathcal{F}_i$  and  $\varphi|_{\mathcal{F}_i}$ . It holds that  $\mathcal{S} \subseteq \mathcal{S}_i$ .

**Stackelberg Laminar Matroid with One Follower.** We are given a ground set  $E = E_f \cup E_p$ . The follower is determined by a laminar matroid  $\mathcal{S}$  based on a laminar family  $\mathcal{F} = \{M_1, \dots, M_s\}$  with a topological order.

To solve the pricing problem, we use a dynamic program whose structure is based on the tree structure of  $\mathcal{F}$ . First, we compute the optimal revenue for the minimal elements for several *configurations*. For an inner node of the tree, the optimal revenue is computed based on the configurations of its direct subsets.

Let  $\mathcal{P} = \{c(e) : e \in E_f\}$  be the set of the fixed-costs and  $\mathcal{P}_\infty = \mathcal{P} \cup \{-\infty, \infty\}$ . For a set  $M$  that contains priceable items, we define  $\mathcal{P}[M] = \{p : M \cap E_p \rightarrow \mathcal{P}\}$  to be the set of all functions that map a priceable item of  $M$  to a value of  $\mathcal{P}$ . According to Lemma 2.1 only the values in  $\mathcal{P}$  are relevant prices.

**Definition 4.1.** Let  $M_i \in \mathcal{F}$ ,  $x \leq \varphi(M_i)$  and  $Q^-, Q^+ \in \mathcal{P}_\infty$ . Then  $\Theta_{x, Q^-, Q^+}^{M_i}$  is the set of all pairs  $(p, S) \in \mathcal{P}[M_i] \times 2^{M_i}$  such that

- T1  $S \subseteq M_i$  and  $|S| = x$ .
- T2  $\max_{e \in S} w(e) = Q^-$ .
- T3  $\min\{w(e) : e \in M_i \setminus S \text{ and } S \cup e \in \mathcal{S}_i\} = Q^+$ .
- T4  $\nexists e \in S, e' \in M_i \setminus S$  such that  $(S \setminus e) \cup e' \in \mathcal{S}_i$  and  $w(e) < w(e')$ .

For a pair  $(p, S) \in \Theta_{x, Q^-, Q^+}^{M_i}$ ,  $S$  is a minimal weight subset of  $M_i$  under price function  $p$ . Set  $S$  has size  $x$  and satisfies the capacity constraint of  $\varphi(M_i)$ .  $Q^-$

is the weight of maximum weight element of  $S$ . Adding an element of  $M_i \setminus S$  to  $S$  such that the capacity constraints are still met increases the weight of  $S$  by at least  $Q^+$ . Based on  $\Theta_{x,Q^-,Q^+}^{M_i}$  we define  $\lambda_{x,Q^-,Q^+}^{M_i}$  as follows:

**Definition 4.2.** Let  $M_i \in \mathcal{F}$ ,  $x \leq \varphi(M_i)$  and  $Q^-, Q^+ \in \mathcal{P}_\infty$ . Then

$$\lambda_{x,Q^-,Q^+}^{M_i} = \max \left\{ \sum_{e \in S \cap E_p} p(e) : (p, S) \in \Theta_{x,Q^-,Q^+}^{M_i} \right\}.$$

Note that if  $Q^+ < Q^-$ ,  $\Theta_{x,Q^-,Q^+}^{M_i} = \emptyset$ , and for a sound definition max is replaced by sup. For a practical algorithm, we set  $\lambda_{x,Q^-,Q^+}^{M_i} = -\infty$  for these cases.

First, we show that if we indeed know all the values of  $\lambda_{x,Q^-,Q^+}^{M_i}$ , we can compute the optimal revenue. Note that each basis of a matroid has the same size and that we can compute this size by computing a basis for any price function.

**Lemma 4.1.** Let  $b$  be the size of a basis of  $\mathcal{S}$ . The maximum revenue  $rev^*$  of the leader can be computed as follows:

$$rev^* = \max_{Q^-, Q^+ \in \mathcal{P}_\infty} \lambda_{b, Q^-, Q^+}^E.$$

Our dynamic program computes all values  $\lambda_{x,Q^-,Q^+}^{M_i}$  in reverse order of the topological order. First, we derive a direct formula for the minimal elements of  $\mathcal{F}$ . Second, we derive a recursive formula for the non-minimal elements of  $\mathcal{F}$ .

*Minimal elements.* Let  $M_i \in \mathcal{F}$  be minimal,  $x \leq \varphi(M_i)$  and  $p \in \mathcal{P}[M_i]$ . We define the set  $S_{x,p}^{M_i}$  to contain the  $x$  items of  $M_i$  with the smallest weight under price function  $p$ . The priceable items are preferred in this selection.

**Definition 4.3.** Let  $M_i \in \mathcal{F}$  be minimal,  $x \leq \varphi(M_i)$  and  $p \in \mathcal{P}[M_i]$ . Moreover, let  $Q^-, Q^+ \in \mathcal{P}_\infty$  such that  $Q^- < Q^+$ . Then,  $J_{x,Q^-,Q^+}^{M_i}$  is the set of indices  $j \in [\min\{x, |M_i \cap E_p|\}]$  such that there is a  $p \in \mathcal{P}[M_i]$  where

$$J1 \quad |S_{x,p}^{M_i} \cap E_p| = j.$$

$$J2 \quad \max_{e \in S_{x,p}^{M_i}} w(e) = Q^-.$$

$$J3 \quad \min\{w(e) : e \in M_i \setminus S_{x,p}^{M_i} \text{ and } S_{x,p}^{M_i} \cup e \in \mathcal{S}_i\} = Q^+.$$

The set  $J_{x,Q^-,Q^+}^{M_i}$  contains the numbers of possible priceable items among the  $x$  minimum weight items in  $M_i$  if their maximum weight item has weight  $Q^-$  and the item in  $M_i$  with the  $x + 1$  largest weight has weight  $Q^+$ . The set  $J_{x,Q^-,Q^+}^{M_i}$  is essential in determining  $\lambda_{x,Q^-,Q^+}^{M_i}$  for the minimal sets  $M_i$ .

**Lemma 4.2.** Let  $M_i \in \mathcal{F}$  be minimal,  $x \leq \varphi(M_i)$  and  $p \in \mathcal{P}[M_i]$ . Moreover, let  $Q^-, Q^+ \in \mathcal{P}_\infty$  such that  $Q^- < Q^+$ . Then,

$$\lambda_{x,Q^-,Q^+}^{M_i} = \begin{cases} \max J_{x,Q^-,Q^+}^{M_i} \cdot Q^-, & J_{x,Q^-,Q^+}^{M_i} \neq \emptyset, \\ -\infty, & \text{otherwise.} \end{cases}$$

We show how to compute  $J_{x,Q^-,Q^+}^{M_i}$  only in the full version of the paper. Basically, this can be done by simple routines that inspect the fixed-cost items in  $M_i$ .

*Non-minimal elements.* We show how to compute  $\lambda_{x, Q^-, Q^+}^{M_i}$  if  $M_i$  is not minimal.

**Definition 4.4.** Let  $M_i \in \mathcal{F}$  such that  $D(M) = \{N_1, \dots, N_s\}$ ,  $x \leq \varphi(M_i)$  and  $Q^-, Q^+ \in \mathcal{P}_\infty$ . Then,  $\Lambda_{x, Q^-, Q^+}^{M_i}$  is the set of all tuples  $(\bar{x}, \bar{Q}^-, \bar{Q}^+) \in [\varphi(M_i)]^s \times \mathcal{P}_\infty^s \times \mathcal{P}_\infty^s$  such that

- L1  $\sum_{j=1}^s \bar{x}_j = x$
- L2  $Q^- = \max_{j \in [s]} \bar{Q}_j^-$
- L3  $Q^+ = \begin{cases} \min_{j \in [s]} \bar{Q}_j^+, & x < \varphi(M_i), \\ \infty, & x = \varphi(M_i). \end{cases}$
- L4  $\max_{j \in [s]} \bar{Q}_j^- \leq \min_{j \in [s]} \bar{Q}_j^+$

**Lemma 4.3.** Let  $M_i \in \mathcal{F}$  such that  $D(M) = \{N_1, \dots, N_s\}$ ,  $x \leq \varphi(M_i)$  and  $Q^-, Q^+ \in \mathcal{P}_\infty$  where  $Q^- \leq Q^+$ . Then,

$$\lambda_{x, Q^-, Q^+}^{M_i} = \max \left\{ \sum_{j=1}^s \lambda_{\bar{x}_j, \bar{Q}_j^-, \bar{Q}_j^+}^{N_i} : (\bar{x}, \bar{Q}^-, \bar{Q}^+) \in \Lambda_{x, Q^-, Q^+}^{M_i} \right\}.$$

With Lemma 4.3, we are able to compute all values of  $\lambda_{x, Q^-, Q^+}^{M_i}$ . First, we compute the values  $\lambda_{x, Q^-, Q^+}^{M_i}$  for the minimal elements of  $\mathcal{F}$ . The running time of this step is in  $\mathcal{O}(|\mathcal{F}|(m+n)^2n^2)$ . Applying the recursive formula of Lemma 4.3 involves checking all the configurations for the direct subsets. Observe that the number of direct subsets in a laminar family can be bounded by 2: Assume we have a laminar family  $\mathcal{F}$  where an element  $M$  has more than 2 direct subsets  $D(M) = \{N_1, N_2, \dots\}$ . In this case, we add  $N_1 \cup N_2$  to  $\mathcal{F}$  and set  $\varphi(N_1 \cup N_2) = \varphi(N_1) + \varphi(N_2)$ . Observe that  $\mathcal{F}$  remains a laminar family. And as such it holds that  $|\mathcal{F}| \leq 2|E|$  (cf. [19]). For a family where the direct subsets of an element are at most 2, the running time of applying the recursion takes times of order  $\mathcal{O}((m+n)^2 \cdot n^4)$ . Finally, we can compute the maximum revenue according to Lemma 4.1.

**Theorem 4.1.** STACKELBERG LAMINAR MATROID *with one follower can be solved in polynomial time.*

It follows that STACKELBERG PARTITION MATROID with one follower can be solved in polynomial time since laminar matroids generalize partition matroids.

## 5 Partition Matroid

For a ground set  $E$ , let  $\mathcal{A}$  be a partition of  $E$  into blocks  $A_1, \dots, A_s$ . Moreover, there are *capacities*  $\varphi : \mathcal{A} \rightarrow \mathbb{N}$  associated with each block of  $\mathcal{A}$ . The *partition matroid*  $\mathcal{S}$  with respect to  $\mathcal{A}$  and  $\varphi$  contains a subsets of  $E$  if its intersection with each block is at most the block's capacity. Formally,

$$\mathcal{S} = \{S \subseteq E : |A_i \cap S| \leq \varphi(A_i) \text{ for each } i \in [s]\}.$$

In this section, we show that STACKELBERG PARTITION MATROID is computationally hard to solve if there are multiple followers. An instance on a ground set  $E$  with  $\ell \geq 2$  followers is given by partition matroids  $\mathcal{S}_1, \dots, \mathcal{S}_\ell$  (each with a possibly different partition and capacities).

**Theorem 5.1.** STACKELBERG PARTITION MATROID *with  $\ell$  followers is NP-hard, for  $\ell \geq 2$ .*

It follows that STACKELBERG LAMINAR MATROID with multiple followers is also NP-hard since partition matroids are a special case of laminar matroids.

*Proof (incomplete).* We consider the decision variant of STACKELBERG PARTITION MATROID where we need to decide if the leader can make more revenue than a given threshold. Our reduction is from the HITTING SET problem which is known to be NP-complete (cf. [14]). Here, we are given a set  $T$ , a value  $t \in \mathbb{N}$  and a family of subsets  $\mathcal{U} = \{U_1, \dots, U_r\}$  of  $T$ . The question is if we can find a *hitting set*  $H \subset T$  such that  $H \cap U \neq \emptyset$  for all  $U \in \mathcal{U}$  and  $|H| \leq t$ .

Given an instance of HITTING SET, we construct an instance of STACKELBERG PARTITION MATROID as follows:

- We have  $E_f = \{g_1, g_2\}$  with costs  $c(g_1) = 1$  and  $c(g_2) = 2$ .
- The set of priceable items is  $E_p = T = \{h_1, \dots, h_n\}$ .

In total, there are  $2 \cdot r + n$  followers.

- For each  $i \in [r]$  there are two identical followers  $F_{i,1}$  and  $F_{i,2}$  with blocks  $X = U_i \cup g_1$  and  $Y = E \setminus X$  where the capacities are  $\varphi(X) = 1$  and  $\varphi(Y) = 0$ .
- For each  $h \in T$  there is one follower  $\bar{F}_h$  with blocks  $X = \{h, g_2\}$  and  $Y = E \setminus X$  where the capacities are  $\varphi(X) = 1$  and  $\varphi(Y) = 0$ .

The idea of the construction is that the leader chooses a hitting set  $H$  by setting  $p(h) = 1$  if  $h \in H$  and  $p(h) = 2$  if  $h \notin H$ . We claim that  $H$  is indeed a hitting set of size at most  $t$  if  $p$  yields revenue at least  $2r - t + 2n$ . The leader receives a revenue of 1 from each of the two identical follower if for each  $U_i$  there exists an priceable item  $h$  with  $p(h) = 1$ . To gain revenue 2 from a follower  $\bar{F}_h$ , the leader has to set  $p(h) = 2$ .

To complete the proof, we show (in the full version of the paper) that our STACKELBERG PARTITION MATROID instance admits revenue of at least  $2r - t + 2n$  if and only if the HITTING SET instance has a hitting set of size at most  $t$ .  $\square$

The decision version of STACKELBERG PARTITION MATROID is NP-complete since we can compute the leader's revenue for a given price function and compare it to a threshold. Moreover, our reduction covers several special cases. Note that we used only two different fixed-cost values and that the partition of each follower contains only 2 blocks.

## 6 Conclusion and Future Work

We make progress towards the more general question of determining the complexity of a Stackelberg pricing problem depending on the complexity of the underlying (follower) optimization problem. With the uniform matroid we identified a case that is solvable in polynomial time if there are more than one followers. We are not aware of another Stackelberg pricing problems of this kind.

A direction for further research is to consider the multiple followers scenario where the items are available in limited supply. Intuitively, pricing problems become harder in this setting (cf. [2]). Several models on how the followers coordinate themselves can be considered. For example, there might be a fixed order in which followers buy a subset of the available item.

## References

1. Baïou, M., Barahona, F.: Stackelberg bipartite vertex cover and the preflow algorithm. *Algorithmica* **74**(3), 1174–1183 (2016)
2. Balcan, M.-F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: Proceedings of the 9th ACM Conference on Electronic Commerce, pp. 50–59 (2008)
3. Bilò, D., Gualà, L., Leucci, S., Proietti, G.: Specializations and generalizations of the stackelberg minimum spanning tree game. *Theoret. Comput. Sci.* **562**, 643–657 (2015)
4. Bilò, D., Gualà, L., Proietti, G., Widmayer, P.: Computational aspects of a 2-player stackelberg shortest paths tree game. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 251–262. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-92185-1\\_32](https://doi.org/10.1007/978-3-540-92185-1_32)
5. Toni Böhnlein, S.K., Schaudt, O.: Revenue maximization in stackelberg pricing games: beyond the combinatorial setting. In: 44th International Colloquium on Automata, Languages, and Programming (ICALP 2017) (2017)
6. Böhnlein, T., Schaudt, O., Schauer, J.: Stackelberg packing games. In: Friggstad, Z., Sack, J.-R., Salavatipour, M.R. (eds.) WADS 2019. LNCS, vol. 11646, pp. 239–253. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-24766-9\\_18](https://doi.org/10.1007/978-3-030-24766-9_18)
7. Briest, P., Chalermsook, P., Khanna, S., Laekhanukit, B., Nanongkai, D.: Improved hardness of approximation for stackelberg shortest-path pricing. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 444–454. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-17572-5\\_37](https://doi.org/10.1007/978-3-642-17572-5_37)
8. Briest, P., Gualà, L., Hoefer, M., Ventre, C.: On stackelberg pricing with computationally bounded customers. *Networks* **60**(1), 31–44 (2012)
9. Briest, P., Hoefer, M., Krysta, P.: Stackelberg network pricing games. *Algorithmica* **62**(3–4), 733–753 (2012)
10. Cabello, S.: Stackelberg shortest path tree game, revisited. arXiv preprint [arXiv:1207.2317](https://arxiv.org/abs/1207.2317) (2012)
11. Cardinal, J., et al.: The stackelberg minimum spanning tree game. *Algorithmica* **59**(2), 129–144 (2011)
12. Cardinal, J., Demaine, E.D., Fiorini, S., Joret, G., Newman, I., Weimann, O.: The stackelberg minimum spanning tree game on planar and bounded-treewidth graphs. *J. Comb. Optim.* **25**(1), 19–46 (2013)

13. Joret, G.: Stackelberg network pricing is hard to approximate. *Networks* **57**(2), 117–120 (2011)
14. Karp, R.M.: Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W., Bohlinger, J.D. (eds.) *Complexity of Computer Computations*. The IBM Research Symposia Series, pp. 85–103. Springer, Boston (1972). [https://doi.org/10.1007/978-1-4684-2001-2\\_9](https://doi.org/10.1007/978-1-4684-2001-2_9)
15. Labbé, M., Marcotte, P., Savard, G.: A bilevel model of taxation and its application to optimal highway pricing. *Manag. Sci.* **44**(12-part-1), 1608–1622 (1998)
16. Labbé, M., Violin, A.: Bilevel programming and price setting problems. *Ann. Oper. Res.* **240**(1), 141–169 (2016)
17. Oxley, J.G.: *Matroid Theory*, vol. 3. Oxford University Press, USA (2006)
18. Roch, S., Savard, G., Marcotte, P.: An approximation algorithm for stackelberg network pricing. *Netw. Int. J.* **46**(1), 57–67 (2005)
19. Schrijver, A.: *Combinatorial Optimization: Polyhedra and Efficiency*, vol. 24. Springer Science & Business Media (2003)
20. Van Hoesel, S.: An overview of stackelberg pricing in networks. *Eur. J. Oper. Res.* **189**(3), 1393–1402 (2008)