On Partial Covering For Geometric Set Systems

Tanmay Inamdar
Department of Computer Science, University of Iowa,
Iowa City, IA, USA.
tanmay-inamdar@uiowa.edu

Kasturi Varadarajan
Department of Computer Science, University of Iowa,
Iowa City, IA, USA.
kasturi-varadarajan@uiowa.edu

Abstract

We study a generalization of the Set Cover problem called the *Partial Set Cover* in the context of geometric set systems. The input to this problem is a set system $(X, \mathcal{R})$, where X is a set of elements and $\mathcal{R}$ is a collection of subsets of X, and an integer $k \le |X|$. Each set in $\mathcal{R}$ has a non-negative weight associated with it. The goal is to cover at least k elements of X by using a minimum-weight collection of sets from $\mathcal{R}$. The main result of this article is an LP rounding scheme which shows that the integrality gap of the Partial Set Cover LP is at most a constant times that of the Set Cover LP for a certain projection of the set system $(X, \mathcal{R})$. As a corollary of this result, we get improved approximation guarantees for the Partial Set Cover problem for a large class of geometric set systems.

2012 ACM Subject Classification Theory of computation → Packing and covering problems, Theory of computation → Rounding techniques, Theory of computation → Computational geometry, Mathematics of computing → Approximation algorithms

Keywords and phrases Partial Set Cover, Geometric Set Cover

Digital Object Identifier 10.4230/LIPIcs.SoCG.2018.47

Funding This material is based upon work supported by the National Science Foundation under Grants CCF-1318996 and CCF-1615845.

1 Introduction

In the Set Cover (SC) problem, the input is a set system $(X, \mathcal{R})$, where X is a set of n elements, and $\mathcal{R}$ is a collection of subsets of X . The goal is to find a minimum-size collection $\mathcal{R}' \subseteq \mathcal{R}$ that *covers* X, i.e., the union of the sets in $\mathcal{R}'$ contains the elements of X. In the weighted version, each set $S_i \in \mathcal{R}$ has a non-negative weight w_i associated with it, and we seek to minimize the weight of $\mathcal{R}'$. A simple greedy algorithm finds a solution that is guaranteed to be within $O(\log n)$ factor from the optimal (see [33] for references), and it is not possible to do better in general using any polynomial-time algorithm , under certain standard complexity theoretic assumptions [15, 10]. In the rest of this article, we assume polynomial running time in any statement that we make about an algorithm.

The question of whether we can improve the $O(\log n)$ bound has been extensively studied for geometric set systems. We focus on three important classes – covering, hitting, and art gallery problems. In the Geometric Set Cover problem, X typically consists of points in $\mathbb{R}^d$, and $\mathcal{R}$ contains sets induced by a certain class of geometric objects via containment. For example, each set in $\mathcal{R}$ might be the subset of X contained in a hypercube. Some of the well-studied examples include covering points by disks in the plane, fat triangles, etc. In the

34th International Symposium on Computational Geometry (SoCG 2018).
Editors: Bettina Speckmann and Csaba D. Tóth; Article No. 47; pp. 47:1–47:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Geometric Hitting Set problem, X is a set of geometric objects, and each set in $\mathcal{R}$ is the subset consisting of all objects in X that are stabbed by some point. In an example of the art gallery problem, X consists of a set of points in a simple polygon, and each set in $\mathcal{R}$ is the subset consisting of all points in X that can be seen by some vertex of the polygon [25]. Thus, the set system here is defined by visibility.

For many such geometric set systems, it is possible to obtain approximation guarantees better than $O(\log n)$. We survey two of the main approaches to obtain such guarantees. The first and the most successful approach is based on the SC Linear Program (LP) and its connection to ε-nets. For completeness, we state the standard SC LP for the weighted case.

$$\begin{aligned}
\text{minimize} & \sum_{S_i \in \mathcal{R}} w_i x_i \\
\text{subject to} & \sum_{i: e_j \in S_i} x_i \geq 1, \qquad e_j \in X \qquad (1)\\
& x_i \geq 0, \qquad S_i \in \mathcal{R} \qquad (2)
\end{aligned}$$

For the unweighted case, Even et al. [13] showed that, if for a certain set system, $O\left(\frac{1}{\varepsilon} \cdot g\left(\frac{1}{\varepsilon}\right)\right)$ size ε-nets exist, then the integrality gap of the SC LP is $O(g(OPT))$, where OPT is the size of the optimal solution. This result is constructive, in that an efficient algorithm for constructing ε-nets also yields an efficient algorithm for obtaining an $O(g(OPT))$ approximation. (A similar result was obtained earlier by Brönnimann and Goodrich [3], without using LP machinery). It is fairly well-known ([8, 20]) that, for a large class of geometric set systems, ε-nets of size $O\left(\frac{1}{\varepsilon} \log\left(\frac{1}{\varepsilon}\right)\right)$ can be computed efficiently, which implies $O(\log(OPT))$ approximation for the set cover problem on the corresponding geometric set system. Clarkson and Varadarajan [9] showed that if the *union complexity* of any set of n objects is $O(n \cdot h(n))$, then ε-nets of size $O\left(\frac{1}{\varepsilon} \cdot h\left(\frac{1}{\varepsilon}\right)\right)$ exist. Aronov et al. [1] gave a tighter bound of $O(\frac{1}{\varepsilon} \cdot \log h(\frac{1}{\varepsilon}))$ on the size of ε-nets for the objects of union complexity $O(n \cdot h(n))$ (see also [31]). Some of these results were extended to the weighted case in [32, 6] by a technique called *quasi-uniform sampling*. We summarize some of these ε-net based results for the set cover problem for geometric set systems in Table 1.

Another approach for tackling SC for geometric set systems is by combinatorial algorithms. The dominant paradigm from this class is a simple local search algorithm. The effectiveness of local search was first demonstrated by Mustafa and Ray [29], who gave the first PTAS for covering points by disks in plane. There have been a series of results that build on their work, culminating in Govindarajan et al. [19], who show that local search yields a PTAS for SC for a fairly general class of objects such as pseudodisks and non-piercing regions in plane. Krohn et al. [27] gave a PTAS for the terrain guarding problem, where the geometric set

Table 1 LP-based approximation ratios for SC. See [9, 1, 32, 14, 6] for the references establishing these bounds. Except for stabbing rectangles in $\mathbb{R}^3$ by points, these bounds hold for the weighted SC. For these problems, we obtain analogous results for weighted PSC.

X	Geometric objects inducing $\mathcal{R}$	Integrality Gap of SC LP
Points in $\mathbb{R}^2$	Disks (via containment)	$O(1)$
	Fat triangles (containment)	$O(\log \log^* n)$
Points in $\mathbb{R}^3$	Unit cubes (containment)	$O(1)$
	Halfspaces (containment)	$O(1)$
Rectangles in $\mathbb{R}^3$	Points (via stabbing)	$O(\log \log n)$
Points on 1.5D terrain	Points on terrain (via visibility)	$O(1)$ [11]

system is defined by visibility. These results for local search only hold for the unweighted set cover problem. Another common strategy, called the *shifting strategy*, was introduced by Hochbaum and Maass [22]. They give a PTAS for covering points by unit balls in $\mathbb{R}^d$; however in this case the set $\mathcal{R}$ consists of *all* unit balls in $\mathbb{R}^d$. Chan [5] gave a PTAS for stabbing a set of fat objects in $\mathbb{R}^d$ using a minimum number of points from $\mathbb{R}^d$. Erlebach and Van Leeuwen [12] combine the shifting strategy with a sophisticated dynamic program to obtain a PTAS for weighted set cover with unit disks in the plane.

Now we turn to the Partial Set Cover (PSC) problem. The input to PSC is the same as that to the SC, along with an additional integer parameter $k \le |X|$. Here the goal is to cover at least k elements from X while minimizing the size (or weight) of the solution $\mathcal{R}' \subseteq \mathcal{R}$. It is easy to see that PSC is a generalization of SC, and hence it is at least as hard as SC. We note here that another classical problem that is related to both of these problems is the so-called Maximum Coverage (MC) problem. In this problem, we have an upper bound on the number of sets that can be chosen in the solution, and the goal is to cover the maximum number of elements. It is a simple exercise to see that an exact algorithm for the unweighted PSC can be used to solve MC exactly, and vice versa. However the reductions are not approximation-preserving. In particular, the greedy algorithm achieves $1 - 1/e$ approximation guarantee for MC — which is essentially the best possible — whereas it is NP-hard to approximate PSC within $o(\log n)$ factor in general. We refer the reader to [24] for a generalization of MC and a survey of results.

For PSC, the greedy algorithm is shown to be an $O(\log \Delta)$ approximation in [23, 30], where Δ is the size of the largest set in $\mathcal{R}$. Bar-Yehuda [2], using the local ratio technique, and Gandhi et al. [17], using the primal-dual method, give algorithms which achieve an approximation guarantee of f, where f is the maximum frequency of any element in the sets. A special case of PSC is the Partial Vertex Cover (PVC) problem, where we need to pick a minimum size (or weight) subset of vertices that covers at least k edges of the graph. Bshouty and Burroughs [4] and [21] present different approaches for obtaining a 2-approximation based on LP rounding for PVC. We refer the reader to [16] for a more detailed history of these foundational results, and to [28, 26] for more recent results on PVC, PSC, and related problems.

While SC for various geometric set systems has been studied extensively, there is relatively less work studying PSC in the geometric setting. Gandhi et al. [17] give a PTAS for a geometric version of PSC where $\mathcal{R}$ consists of *all* unit disks in the plane. They provide a dynamic program on top of the standard shifting strategy of Hochbaum and Maass [22], thus adapting it for PSC. Using a similar technique, Glaßer et al. [18] give a PTAS for a generalization of partial geometric covering, under a certain assumption on the *density* of the given disks. Chan and Hu [7] give a PTAS for PSC where $\mathcal{R}$ consists of a given set of unit squares in the plane, by combining the shifting strategy with sophisticated dynamic programming.

Our results and techniques

Suppose that we are given a PSC instance $(X, \mathcal{R}, k)$. For any set of elements $Y \subseteq X$, let $\mathcal{R}_{|Y} := \{S \cap Y \mid S \in \mathcal{R}\}$ denote the projected set system. Suppose also that for any projected SC instance $(Y, \mathcal{R}_{|Y})$, (where $Y \subseteq X$) and a corresponding feasible SC LP solution σ_1, we can round σ_1 to a feasible integral SC solution with cost at most β times that of σ_1. That is, we suppose that we can efficiently compute a β-approximation for the SC instance $(Y, \mathcal{R}_{|Y})$ by solving the natural LP relaxation and rounding it. Then, we show that we can round the solution to the natural PSC LP for $(X, \mathcal{R}, k)$ to an integral solution to within a $2\beta + 2$

factor. By the previous discussion about the existence of such rounding algorithms for SC LP for a large class of geometric objects (cf. Table 1), we get the same guarantees for the corresponding PSC instances as well, up to a constant factor. For clarity, we describe a sample of these applications.

1. Suppose we are given a set P of n points and a set $\mathcal{T}$ of *fat* triangles in the plane and a positive weight for each triangle in $\mathcal{T}$. We wish choose a subset $\mathcal{T}' \subseteq \mathcal{T}$ of triangles that covers P, and minimize the weight of $\mathcal{T}'$, defined to be the sum of the weights of the triangles in it. This is a special case of weighted SC obtained by setting $X = P$, and adding the set $t \cap P$ to $\mathcal{R}$ for each triangle in $t \in \mathcal{T}$, with the same weight. There is an $O(\log \log^* n)$ approximation for this problem based on rounding SC LP [14, 6]. We obtain the same approximation guarantee for the partial covering version, where we want a minimum weight subset of $\mathcal{T}$ covering any k of the points in P.

2. Suppose we are given a set $\mathcal{B}$ of n axis-parallel boxes and a set P of points in $\mathbb{R}^3$, and we wish to find a minimum cardinality subset of P that hits (or stabs) each box in $\mathcal{B}$. This a special case of SC obtained by setting $X = \mathcal{B}$, and adding the set $\{b \in \mathcal{B} \mid p \in b\}$ to $\mathcal{R}$ for each point $p \in P$. There is an $O(\log \log n)$ approximation for this problem based on rounding SC LP [1]. Thus, we obtain the same approximation guarantee for the partial version, where we want a minimum cardinality subset of P stabbing any k of the boxes in $\mathcal{B}$.

3. Suppose we have a 1.5D terrain (i.e., an x-monotone polygonal chain in $\mathbb{R}^2$), a set P of points and a set G of n points, called guards, on the terrain along with a positive weight for each guard in G. The goal is to choose a subset $G' \subset G$ such that each point in P is seen by some guard in G, and minimize the weight of G'. Two points p and g on the terrain see each other if the line segment connecting them does not contain a point below the terrain. This is a special case of SC obtained by setting $X = P$, and adding the set $\{p \in P \mid g \text{ sees } p\}$ to $\mathcal{R}$ for each guard $g \in G$. There is an $O(1)$-approximation guarantee for this problem based on rounding SC LP [11]. Thus, we obtain an $O(1)$-approximation for the partial version, where we want a minimum weight subset of G that sees any k of the points in P.

Our algorithm for rounding a solution to the natural PSC LP corresponding to partial cover instance $(X, \mathcal{R}, k)$ proceeds as follows. Let X_1 be the elements that are covered by the LP solution to an extent of at least 1/2. By scaling the LP solution by a factor of 2, we get a feasible solution to the SC LP corresponding to $(X_1, \mathcal{R}_{|X_1})$, which we round using the LP-based β-approximation algorithm. For the set $X \setminus X_1$, the LP solution provides a total fractional coverage of at least $k - |X_1|$. Crucially, each element of $X \setminus X_1$ is *shallow* in that it is covered to an extent of at most 1/2. We use this observation to round the LP solution to an integer solution, of at most twice the cost, that covers at least $k - |X_1|$ points of $X \setminus X_1$. This rounding step and its analysis are inspired by the PVC rounding scheme of [4], but there are certain subtleties in adapting it to the PSC problem. To the best of our knowledge, this connection between the SC LP and PSC LP was not observed before.

The rest of this article is organized as follows. In Section 2, we describe the standard LP formulation for the PSC problem, and give an integrality gap example. We describe how to circumvent this integrality gap by preprocessing the input in Section 3. Finally, in Section 4, we describe and analyze the main LP rounding algorithm.

2 Preliminaries

We use the following Integer Programming formulation of PSC (see left side of display below). Here, for each element $e_j \in X$, the variable z_j denotes whether it is one of the k elements that are chosen by the solution. For each such chosen element e_j, the first constraint ensures that at least one set containing it must be chosen. The second constraint ensures that at least k elements are covered by the solution. We relax the integrality Constraints 3, and 4, and formulate it as a Linear Program (see right side).

Integer Program

$$
\begin{aligned}
\text{minimize} \quad & \sum_{S_i \in \mathcal{R}} w_i x_i \\
\text{subject to} \quad & \sum_{i: e_j \in S_i} x_i \geq z_j, \quad && e_j \in X \\
& \sum_{e_j \in X} z_j \geq k, \\
& z_j \in \{0,1\}, && e_j \in X \quad (3)\\
& x_i \in \{0,1\}, && S_i \in \mathcal{R} \quad (4)
\end{aligned}
$$

Linear Program

$$
\begin{aligned}
\text{minimize} \quad & \sum_{S_i \in \mathcal{R}} w_i x_i && (5)\\
\text{subject to} \quad & \sum_{i: e_j \in S_i} x_i \geq z_j, \quad && e_j \in X \quad (6)\\
& \sum_{e_j \in X} z_j \geq k, && (7)\\
& z_j \in [0,1], && e_j \in X \quad (8)\\
& x_i \in [0,1], && S_i \in \mathcal{R} \quad (9)
\end{aligned}
$$

Since SC is a special case of PSC where $k = n$, the corresponding LP can be obtained by setting k appropriately in Constraint 7. However, in this case, the LP can be further simplified as described earlier. We denote the cost of a PSC LP solution $\sigma = (x, z)$, for the instance $(X, \mathcal{R})$, as $cost(\sigma) := \sum_{S_i \in \mathcal{R}} w_i x_i$, and the cost of an SC LP solution is defined in exactly the same way. Also, for any collection of sets $\mathcal{R}' \subseteq \mathcal{R}$, we define $w(\mathcal{R}') := \sum_{S_i \in \mathcal{R}'} w_i$. Finally, for a PSC instance $(X, \mathcal{R}, k)$, let $OPT(X, \mathcal{R}, k)$ denote the cost of an optimal solution for that instance.

Unlike SC LP, the integrality gap of PSC LP can be $\Omega(n)$, even for the unweighted case. **Integrality Gap:** Consider the set system $(X, \mathcal{R})$, where $X = \{e_1, \ldots, e_n\}$, and $\mathcal{R} = \{S_1\}$, where $S_1 = X$. Here, $k = 1$, so at least one element has to be covered. The size of the optimal solution is 1, because the only set S_1 has to be chosen. However, consider the following fractional solution $\sigma = (x, z)$, where $z_j = \frac{1}{n}$ for all $e_j \in X$, and $x_1 = \frac{1}{n}$, which has the cost of $\frac{1}{n}$. This shows the integrality gap of n.

However, Gandhi et al. [17] show that after "guessing" the heaviest set in the optimal solution, the integrality gap of the LP corresponding to the residual instance is at most f, where f is the maximum frequency of any element in the set system. In this article, we show that after guessing the heaviest set in the optimal solution, the residual instance has integrality gap at most $2\beta + 2$, where β is the integrality gap of the SC LP for some projection of the same set system.

3 Preprocessing

Henceforth, for convenience, we let $(X', \mathcal{R}', k')$ denote our original input instance of PSC. To circumvent the integrality gap, we preprocess the given instance to "guess" the heaviest set in the optimal solution, and solve the residual instance as in [4, 17] – see Algorithm 1. Let us renumber the sets $\mathcal{R}' = \{S_1, \ldots, S_m\}$, so that $w_1 \leq w_2 \leq \ldots \leq w_m$. For each $S_i \in \mathcal{R}'$, let $\mathcal{R}_i = \{S_1, S_2, \ldots, S_{i-1}\}$, and $X_i = X' \setminus S_i$. We find the approximate solution Σ_i for this residual instance $(X_i, \mathcal{R}_i, k_i)$ with coverage requirement $k_i = k - |S_i|$, if it is feasible (i.e.

Algorithm 1 PartialCover$(X', \mathcal{R}', k')$.

Sort and renumber the sets in $\mathcal{R}' = \{S_1, \ldots, S_m\}$ such that $w_1 \leq \ldots \leq w_m$.
for $i = 1$ to m **do**
  $\mathcal{R}_i \leftarrow \{S_1, \ldots, S_{i-1}\}$
  $X_i \leftarrow X' \setminus S_i$
  $k_i \leftarrow k' - |S_i|$
  if $(X_i, \mathcal{R}_i, k_i)$ is feasible **then**
    $\Sigma_i \leftarrow$ approximate solution to $(X_i, \mathcal{R}_i, k_i)$
  else
    $\Sigma_i \leftarrow \perp$
  end if
end for
$\ell \leftarrow \arg\min_{i:\Sigma_i \neq \perp} w(\Sigma_i \cup \{S_i\})$
return $\Sigma_\ell \cup \{S_\ell\}$

$\left|\bigcup_{S \in \mathcal{R}_i} S \cap X_i\right| \geq k_i)$. We return $\Sigma = \arg\min_{S_i \in \mathcal{R}'} w(\Sigma_i \cup \{S_i\})$ over all S_i such that the residual instance $(X_i, \mathcal{R}_i, k_i)$ is feasible.

▶ **Lemma 1.** *Let Σ^* be an optimal partial cover for the instance $(X', \mathcal{R}', k')$, and let S_p be the heaviest set in Σ^*. Let Σ_p be the approximate solution to $(X_p, \mathcal{R}_p, k_p)$ returned by the Rounding Algorithm of Theorem 3, and Σ' be the solution returned by Algorithm 1. Then,*
1. $OPT(X', \mathcal{R}', k') = OPT(X_p, \mathcal{R}_p, k_p) + w_p$
2. $w(\Sigma') \leq w(\Sigma_p \cup \{S_p\}) \leq (2\beta + 2) \cdot OPT(X', \mathcal{R}', k')$

Proof. Since the optimal solution Σ^* contains S_p, $\Sigma_p^* := \Sigma^* \setminus \{S_p\}$ covers at least $k' - |S_p| = k_p$ elements from $X' \setminus S_p$. Therefore, Σ_p^* is feasible for $(X_p, \mathcal{R}_p, k_p)$. Now, an easy and standard argument implies that Σ_p^* is an optimal solution for $(X_p, \mathcal{R}_p, k_p)$. Thus, $w(\Sigma_p^*) = OPT(X_p, \mathcal{R}_p, k_p)$ and the first part follows.

From Theorem 3, we have an approximate solution Σ_p to the instance $(X_p, \mathcal{R}_p, k_p)$ such that $w(\Sigma_p) \leq (2\beta+2) \cdot OPT(X_p, \mathcal{R}_p, k_p) + B$, where $B \leq w_p$ is the weight of the heaviest set in $\mathcal{R}_p$. Now Algorithm 1 returns a solution whose cost is at most $w(\Sigma_p \cup \{S_p\}) \leq (2\beta + 2) \cdot OPT(X_p, \mathcal{R}_p, k_p) + w_p + w_p \leq (2\beta+2) \cdot (OPT(X_p, \mathcal{R}_p, k_p) + w_p) \leq (2\beta+2) \cdot OPT(X', \mathcal{R}', k')$. We use the result from part 1 in the final inequality. ◀

We summarize our main result in the following theorem, which follows easily from Lemma 1.

▶ **Theorem 2.** *Given our input partial set cover instance $(X', \mathcal{R}', k')$, assume there is a $\beta \geq 1$ such that for any $X_1 \subseteq X'$, we can round a solution to SC LP for the projected set system $(X_1, \mathcal{R}'_{|X_1})$ to within a β factor. Then, we can find a $(2\beta + 2)$-factor approximation for the partial set cover instance $(X', \mathcal{R}', k')$.*

4 Rounding algorithm

Suppose that we have guessed the maximum weight set $S_p \in \mathcal{R}'$ in the optimal solution for the original instance $(X', \mathcal{R}', k')$, as described in the previous section. Thus, we now have the residual instance $(X_p, \mathcal{R}_p, k_p)$, where $X_p = (X' \setminus S_p), \mathcal{R}_p = \{S_1, S_2, \ldots, S_{p-1}\}$, and $k_p = k' - |S_p|$. We solve the LP corresponding to the PSC instance $(X_p, \mathcal{R}_p, k_p)$ to obtain an optimal LP solution $\sigma^* = (x, z)$. In the following, we describe a polynomial time algorithm to round PSC LP on this instance.

Let $0 < \alpha \leq 1/2$ be a parameter (eventually we will set $\alpha = 1/2$). Let $Y = \{e_j \in X_p \mid \sum_{i:e_j \in S_i} x_i \geq \alpha\}$ be the set of elements that are covered to an extent of at least α by the LP solution.

We create a solution σ_1 of a feasible set cover LP for the instance $(Y, \mathcal{R}_{p|Y})$ as follows. For all sets $S_i \in \mathcal{R}_p$, we set $x_i' = \min\{\frac{x_i}{\alpha}, 1\}$. Note that cost of this fractional solution is at most $\frac{1}{\alpha}$ times that of σ^*. Also, note that σ_1 is feasible for the SC LP because for any element $e_j \in Y$, we have that

$$\sum_{i:e_j \in S_i} x_i' = \sum_{i:e_j \in S_i} \min\left\{1, \frac{x_i}{\alpha}\right\} \geq \min\left\{1, \frac{1}{\alpha} \sum_{i:e_j \in S_i} x_i\right\} \geq 1.$$

Suppose that there exists an efficient rounding procedure to round a feasible SC LP solution σ_1, for the instance $(Y, \mathcal{R}_{p|Y})$ to a solution with weight at most $\beta \cdot cost(\sigma_1)$. In the remainder of this section, we describe an algorithm (Algorithm 2) for rounding $\sigma^* = (x, z)$ into a solution that (1) covers at least $k_p - |Y|$ elements from $X_p \setminus Y$, and (2) has cost at most $\frac{1}{\alpha} \cdot cost(\sigma^*) + B$, where B is the weight of the heaviest set in $\mathcal{R}_p$. Combining the two solutions thus acquired, we get the following theorem.

▶ **Theorem 3.** *There exists a rounding algorithm to round a partial cover LP corresponding to $(X_p, \mathcal{R}_p, k_p)$, which returns a solution Σ_p such that $w(\Sigma_p) \leq (2\beta + 2) \cdot OPT(X_p, \mathcal{R}_p, k_p) + B$, where B is the weight of the heaviest set in $\mathcal{R}_p$.*

Proof. Let $\Sigma_p = \Sigma_{p1} \cup \Sigma_{p2}$, where Σ_{p1} is the solution obtained by rounding σ_1, and $\Sigma_{p2} = \Sigma \cup \mathcal{R}_{\text{end}}$ is the solution returned by Algorithm 2. By assumption, Σ_{p1} covers Y, and Σ_{p2} covers at least $k_p - |Y|$ elements from $X_p \setminus Y$ by Lemma 8. Therefore, Σ_p covers at least k_p elements from X_p.

By assumption, we have that $w(\Sigma_{p1}) \leq \beta \cdot cost(\sigma_1) \leq \frac{\beta}{\alpha} cost(\sigma^*)$. Also, from Lemma 9, we have that $w(\Sigma_{p2}) \leq \frac{1}{\alpha} cost(\sigma^*) + B$. We get the claimed result by combining the previous two inequalities, setting $\alpha = 1/2$, and noting that $cost(\sigma^*) \leq OPT(X_p, \mathcal{R}_p, k_p)$. ◀

Let $\mathcal{H} = \{S_i \in \mathcal{R}_p \mid x_i \geq \alpha\}$ be the sets that have x_i value at least α. Note that without loss of generality, we can assume that $\cup_{S_i \in \mathcal{H}} S_i \subseteq Y$. If $|Y| \geq k_p$, we are done. Otherwise, let $X \leftarrow X_p \setminus Y$, $\mathcal{R} \leftarrow \mathcal{R}_p \setminus \mathcal{H}$, and $k \leftarrow k_p - |Y|$. Let $\sigma = (x, z)$ be the LP solution σ^* restricted to the instance $(X, \mathcal{R}, k)$, that is, $x = (x_i \mid S_i \in \mathcal{R}), z = (z_j \mid e_j \in X)$. We show how to round σ on the instance $(X, \mathcal{R}, k)$ to find a collection of sets that covers at least k elements from X. In the following lemma, we show that the LP solution σ is feasible for the instance $(X, \mathcal{R}, k)$.

▶ **Lemma 4.** *The LP solution $\sigma = (x, z)$ is feasible for the instance $(X, \mathcal{R}, k)$. Furthermore, $cost(\sigma) \leq cost(\sigma^*)$.*

Proof. Note that x_i and z_j values are unchanged from the optimal solution σ^*, therefore the Constraints 9, and 8 (from PSC LP) are satisfied.

Note that by definition, for any element $e_j \in X$, $e_j \notin \cup_{S_{i'} \in \mathcal{H}} S_{i'}$, and $e_j \notin Y$. Therefore, by Constraint 6, we have that $\displaystyle\sum_{i:e_j \in S_i} x_i = \sum_{i:e_j \in S_i, S_i \in \mathcal{R}} x_i \geq z_j$.

As for Constraint 7, note that

$$\sum_{e_j \in X_p} z_j \geq k_p \qquad \text{(By feasibility of optimal solution } \sigma^*\text{)}$$
$$\implies \sum_{e_j \in X} z_j \geq k_p - \sum_{e_j \in Y} z_j \qquad (X = X_p \setminus Y)$$
$$\implies \sum_{e_j \in X} z_j \geq k_p - |Y| \qquad (z_j \leq 1 \text{ for } e_j \in Y \text{ by feasibility})$$
$$\implies \sum_{e_j \in X} z_j \geq k \qquad (k = k_p - |Y|)$$

Finally, note that $cost(\sigma) = \sum_{S_i \in \mathcal{R}} w_i x_i \leq \sum_{S_i \in \mathcal{R}_p} w_i x_i = cost(\sigma^*)$, because $\mathcal{R} \subseteq \mathcal{R}_p$, and the x_i values are unchanged. ◀

4.1 Algorithm for rounding shallow elements

We have an LP solution σ for the PSC instance $(X, \mathcal{R}, k)$. Note that for any $S_i \in \mathcal{R}$, $x_i < \alpha$, and for any $e_j \in X, \alpha > \sum_{i:e_j \in S_i} x_i \geq z_j$, i.e. each element is *shallow*. For convenience, we let $z_j = \sum_{i:e_j \in S_i} x_i$. We now describe Algorithm 2, which rounds σ to an integral solution to the instance $(X, \mathcal{R}, k)$. At the beginning of Algorithm 2, we initialize $\mathcal{R}_{\text{cur}}$, the collection of "unresolved" sets, to be $\mathcal{R}$; and X_{cur}, the set of "uncovered" elements, to be X.

At the heart of the rounding algorithm is the procedure RoundTwoSets, which takes input two sets $S_1, S_2 \in \mathcal{R}_{\text{cur}}$, and rounds the corresponding variables x_1, x_2 such that either x_1 is increased to α, or x_2 is decreased to 0 (cf. Lemma 5 part 3). A set is removed from $\mathcal{R}_{\text{cur}}$ if either of these conditions is met. In addition, if x_i reaches α, then the set S_i is added to Σ, which is a part of the output, and all the elements in S_i are added to the set Ξ; furthermore, x_i is set to 1. At a high level, the goal of Algorithm 2 is to resolve all of the sets either way, while maintaining the cost and the feasibility of the LP.

Given the procedure RoundTwoSets, we carefully choose the order in which the sets are paired up for rounding; however, there is some degree of freedom. We pick a set from $\mathcal{R}_{\text{cur}}$ in a careful way as the *leader*. We use variable a to denote the index of the leader; thus, the leader is S_a. The leader S_a is chosen arbitrarily in Line 4 but in a specific way in Line 20. We keep pairing the leader S_a up with another arbitrary set $S_b \in \mathcal{R}_{\text{cur}}$, until S_a is removed from $\mathcal{R}_{\text{cur}}$, or it is the only set remaining in $\mathcal{R}_{\text{cur}}$. To ensure that the Constraint 7 is maintained, we carefully determine whether to increase x_a and decrease x_b in RoundTwoSets, or vice versa. Thinking of $\frac{|X_{\text{cur}} \cap S_a|}{w_a}$, and $\frac{|X_{\text{cur}} \cap S_b|}{w_b}$ as the "cost-effectiveness" of the sets S_a and S_b respectively, we increase x_a at the expense of x_b, if S_a is more cost-effective than S_b or vice versa.

Notice that, instead of fixing a set S_a and pairing it up with other sets S_b, if we arbitrarily chose the pairs of sets to be rounded, then the feasibility of the LP may not be maintained. In particular, we cannot ensure that for all elements $e_j \in X_{\text{cur}}$, $z_j \leq 1$ (Constraint 8). To this end, we show that, our order of pairing up sets maintains the following two invariants:

1. Let $X_\alpha = \{e_j \in X_{\text{cur}} \mid z_j \geq \alpha\}$. During the execution of while loop of Line 3, the elements of X_α are contained in the leader $S_a \in \mathcal{R}_{\text{cur}}$, that is chosen in Line 4 or Line 20.
2. Fix any set $S_i \in \mathcal{R}_{\text{cur}} \setminus \{S_a\}$. The x_i value is unchanged since the beginning of the algorithm until the beginning of the current iteration of while loop of Line 5; the x_i value can change in the current iteration only if S_i is paired up with S_a.

In Lemma 7, we show that these invariants imply that Constraint 8 is maintained.

Algorithm 2 RoundLP$(X, \mathcal{R}, w, k, \sigma)$.

1: $\Sigma \leftarrow \emptyset$, $\Xi \leftarrow \emptyset$.
2: $X_{\text{cur}} \leftarrow X$, $\mathcal{R}_{\text{cur}} \leftarrow \mathcal{R}$.
3: **while** $|\mathcal{R}_{\text{cur}}| \geq 2$ **do**
4:     $a \leftarrow \ell$, where S_ℓ is an arbitrary set from $\mathcal{R}_{\text{cur}}$.
5:     **while** $0 < x_a < \alpha$ and $|\mathcal{R}_{\text{cur}} \setminus \{S_a\}| \geq 1$ **do**
6:         $S_b \leftarrow$ an arbitrary set from $\mathcal{R}_{\text{cur}} \setminus \{S_a\}$.
7:         **if** $\frac{|X_{\text{cur}} \cap S_a|}{w_a} \geq \frac{|X_{\text{cur}} \cap S_b|}{w_b}$ **then**
8:             $(x_a, x_b, z) \leftarrow$RoundTwoSets$(S_a, S_b, w, \sigma, X_{\text{cur}}, \mathcal{R}_{\text{cur}})$
9:             **if** $x_b = 0$ **then**
10:                 $\mathcal{R}_{\text{cur}} \leftarrow \mathcal{R}_{\text{cur}} \setminus \{S_b\}$.
11:             **end if**
12:             **if** $x_a = \alpha$ **then**
13:                 $\Xi \leftarrow \Xi \cup S_a$, $X_{\text{cur}} \leftarrow X_{\text{cur}} \setminus S_a$.
14:                 $\Sigma \leftarrow \Sigma \cup \{S_a\}, \mathcal{R}_{\text{cur}} \leftarrow \mathcal{R}_{\text{cur}} \setminus \{S_a\}$, $x_a \leftarrow 1$.
15:             **end if**
16:         **else**
17:             $(x_b, x_a, z) \leftarrow$RoundTwoSets$(S_b, S_a, w, \sigma, X_{\text{cur}}, \mathcal{R}_{\text{cur}})$.
18:             **if** $x_a = 0$ **then**
19:                 $\mathcal{R}_{\text{cur}} \leftarrow \mathcal{R}_{\text{cur}} \setminus \{S_a\}$.
20:                 $a \leftarrow b$.
21:             **end if**
22:             **if** $x_b = \alpha$ **then**
23:                 $\Xi \leftarrow \Xi \cup S_b$, $X_{\text{cur}} \leftarrow X_{\text{cur}} \setminus S_b$.
24:                 $\Sigma \leftarrow \Sigma \cup \{S_b\}, \mathcal{R}_{\text{cur}} \leftarrow \mathcal{R}_{\text{cur}} \setminus \{S_b\}$, $x_b \leftarrow 1$.
25:             **end if**
26:         **end if**
27:     **end while**
28: **end while**
29: $\mathcal{R}_{\text{end}} \leftarrow \mathcal{R}_{\text{cur}}$.
30: **return** $\Sigma \cup \mathcal{R}_{\text{end}}$.

31: **function** RoundTwoSets$(S_1, S_2, w, \sigma, X_{\text{cur}}, \mathcal{R}_{\text{cur}})$
32:     $\delta \leftarrow \min\{\alpha - x_1, \frac{w_2}{w_1} \cdot x_2\}$.
33:     $x_1 \leftarrow x_1 + \delta$.
34:     $x_2 \leftarrow x_2 - \frac{w_1}{w_2} \cdot \delta$.
35:     For all elements $e_j \in X_{\text{cur}}$, update $z_j \leftarrow \sum_{i: e_j \in S_i} x_i$.
36:     **return** (x_1, x_2, z).
37: **end function**

The invariants are trivially true at the start of the first iteration of the while loops. Let $S_a \in \mathcal{R}_{\text{cur}}$ be a set chosen in Line 4, or Line 20. During the while loop, we maintain the invariants by pairing up the S_a with other arbitrary sets S_b, until S_a is removed from $\mathcal{R}_{\text{cur}}$ in one of the two ways; or until it is the last set remaining. It is easy to see that Invariant 2 is maintained.

Now we describe in detail how Invariant 1 is being maintained in the course of the algorithm. Consider the first case, i.e. in RoundTwoSets, we increase x_a and decrease x_b. If after this, x_b becomes 0, then we remove S_b from $\mathcal{R}_{\text{cur}}$. If, on the other hand, x_a increases to α, then all the elements in $X_{\text{cur}} \cap S_a$ are covered to an extent of at least α, and so we remove S_a from $\mathcal{R}_{\text{cur}}$ and $S_a \cap X_{\text{cur}}$ from X_{cur}. If x_b becomes 0, the set X_α continues to be a subset of S_a, and if x_a increases to α, it becomes empty. Thus, Invariant 1 is maintained.

In the second case, in RoundTwoSets, x_a is decreased and x_b is increased. This case is a bit more complicated, because z_j values of elements $e_j \in S_b$ are being increased by virtue of increase in x_b. Therefore, we need to explicitly maintain Invariant 1. If x_b reaches α, then S_b is removed from $\mathcal{R}_{\text{cur}}$ and all the elements covered by S_b are removed from X_{cur} (and thus the invariant is maintained). On the other hand, if x_a reaches 0, then the net change in the z_j values (since the beginning of the algorithm) for the elements $e_j \in S_a \setminus S_b$ is non-positive – this follows from Invariant 2, as the x_i values of the sets in $\mathcal{R}_{\text{cur}} \setminus \{S_a, S_b\}$ are unchanged, and x_a is now zero. Therefore, the set $X_\alpha \cap (S_a \setminus S_b) = \varnothing$. However, $X_\alpha \cap S_b$ may be non-empty because of the increase in x_b. Therefore, we reset a to b, thus obtaining a new leader $S_a = S_b$, and continue pairing the new leader up with other sets. Notice that we have maintained Invariant 1 although the leader S_a has changed.

From the above discussion, we have the following result.

▶ **Claim 1.** *Throughout the execution of the while loop of Line 3, Invariants 1, and 2 are maintained.*

Finally, after leaving the while loop of Line 3, we set $\mathcal{R}_{\text{end}}$ to be $\mathcal{R}_{\text{cur}}$, and add it to our solution. Note that at this point, $\mathcal{R}_{\text{cur}}$ contains at most one set. We show that the resulting solution $\Sigma \cup \mathcal{R}_{\text{end}}$ covers at least k elements.

4.2 Analysis

In this section, we analyze the behavior of Algorithm 2. In the following lemma, we show that in each iteration, we make progress towards rounding while maintaining the cost of the LP solution.

▶ **Lemma 5.** *Let $\sigma = (x, z), \sigma' = (x', z')$ be the LP solutions just before and after the execution of* RoundTwoSets$(S_1, S_2, w, \sigma, X_{cur}, \mathcal{R}_{cur})$ *for some sets $S_1, S_2 \in \mathcal{R}_{cur}$ in some iteration of the algorithm, such that σ is a feasible solution to the LP. Then,*

1. *$cost(\sigma) = cost(\sigma')$.*
2. $\sum_{e_j \in X_{cur}} z'_j \geq \sum_{e_j \in X_{cur}} z_j$.
3. *Either $x'_1 = \alpha$ or $x'_2 = 0$ (or both).*

Proof.

1. Note that the x_i variables corresponding to all the sets $S_i \notin \{S_1, S_2\}$ remain unchanged. The net change in the cost of the LP solution is

$$w_1 \cdot (x'_1 - x_1) + w_2 \cdot (x'_2 - x_2) = w_1 \cdot \delta - w_2 \cdot \left(\frac{w_1}{w_2} \cdot \delta\right) = 0.$$

2. Let $A = S_1 \cap X_{\text{cur}}$, and $B = S_2 \cap X_{\text{cur}}$. $z'_j = z_j$ for all elements $e_j \notin A \cup B$, i.e. z_j values are modified only for the elements $e_j \in A \cup B$.
   For $|A|$ elements $e_j \in A$, z_j value is increased by δ by virtue of increase in x_1. Similarly, for $|B|$ elements $e_{j'} \in B$, $z_{j'}$ value is decreased by $\frac{w_1}{w_2} \cdot \delta$. However by assumption, we have that $\frac{|A|}{w_1} \geq \frac{|B|}{w_2}$. Therefore, the net change in the sum of z_j values is

$$|A| \cdot \delta - |B| \cdot \left(\frac{w_1}{w_2} \cdot \delta\right) \geq |A| \cdot \delta - \left(\frac{|A|}{w_1} \cdot w_1\right) \cdot \delta \geq 0.$$

3. The value of δ is chosen such that $\delta = \min\{\alpha - x_1, \frac{w_2}{w_1} \cdot x_2\}$. If $\delta = \alpha - x_1 \leq \frac{w_2}{w_1} \cdot x_2$, then $x'_1 = x_1 + (\alpha - x_1) = \alpha$, and $x'_2 = x_2 - \frac{w_1}{w_2} \cdot (\alpha - x_1) \geq x_2 - x_2 = 0$. In the other case when $\delta = \frac{w_2}{w_1} \cdot x_2 < (\alpha - x_1)$, we have that $x'_1 = x_1 + \frac{w_2}{w_1} \cdot x_2 < x_1 + (\alpha - x_1) = \alpha$, and $x'_2 = x_2 - \frac{w_2}{w_1} \cdot \frac{w_1}{w_2} \cdot x_2 = 0$. ◀

▶ Remark. Note that Lemma 5 (in particular, the Part 2 of Lemma 5) alone is not sufficient to show the feasibility of the LP after an execution of RoundTwoSets – we also have to show that $z'_j \leq 1$. This is slightly involved, and is shown in Lemma 7 with the help of Invariants 1, and 2.

▶ **Corollary 6.** *Algorithm 2 runs in polynomial time.*

Proof. In each iteration of the inner while loop Line 5, RoundTwoSets is called on some two sets $S_1, S_2 \in \mathcal{R}_{\text{cur}}$, and as such from Lemma 5, either $x'_1 = \alpha$ or $x'_2 = 0$. Therefore, at least one of the sets is removed from $\mathcal{R}_{\text{cur}}$ in each iteration. Therefore, there are at most $O(|\mathcal{R}|)$ iterations of the inner while loop. It is easy to see that each execution of RoundTwoSets takes $O(|\mathcal{R}| \cdot |X|)$ time. ◀

In the following Lemma, we show that Constraints 8 is being maintained by the algorithm. This, when combined with Lemma 5, shows that we maintain the feasibility of the LP at all times.

▶ **Lemma 7.** *During the execution of Algorithm 2, for any element $e_j \in X_{cur}$, we have that $z_j \leq 2\alpha$. By the choice of range of α, the feasibility of the LP is maintained.*

Proof. At the beginning of the algorithm, we have that $z_j \leq \alpha$ for all elements $e_j \in X_{\text{cur}} = X$. Now at any point in the while loop, consider the set $X_\alpha = \{e_j \in X_{\text{cur}} \mid z_j \geq \alpha\}$ as defined earlier. For any element $e_j \in X_{\text{cur}} \setminus X_\alpha$, the condition is already met, therefore we need to argue only for the elements in X_α. We know by Invariant 1 that there exists a set $S_a \in \mathcal{R}_{\text{cur}}$ such that $X_\alpha \subseteq S_a$.

By Invariant 2, the x_i values of all sets $S_i \in \mathcal{R}_{\text{cur}} \setminus \{S_a\}$ are unchanged, and therefore for all elements $e_j \in X_{\text{cur}}$, the net change to the z_j variable is positive only by the virtue of increase in the x_a value. However, the net increase in the x_a value is at most α because a set S_i is removed from $\mathcal{R}_{\text{cur}}$ as soon as its x_i value reaches α. Accounting for the initial z_j value which is at most α, we conclude that $z_j \leq 2\alpha$. ◀

Note that after leaving the outer while loop (Line 28), we must have $|\mathcal{R}_{\text{cur}}| \leq 1$. That is in Line 29, we either let $\mathcal{R}_{\text{end}} \leftarrow \mathcal{R}_{\text{cur}} = \emptyset$, or $\mathcal{R}_{\text{end}} \leftarrow \mathcal{R}_{\text{cur}} = \{S_i\}$ for some set $S_i \in \mathcal{R}$.

To state the following claim, we introduce the following notation. Let $\sigma' = (x', z')$ be the LP solution at the end of Algorithm 2. Let $\mathcal{R}_r = \mathcal{R} \setminus \Sigma$, where Σ is the collection at the end of the while loop of Algorithm 2, and let $X_r = X \setminus \Xi$. Note that any element $e_j \in X_r$ is contained only in the sets of $\mathcal{R}_r$. Finally, let $Z_r = \sum_{e_j \in X_r} z'_j$.

▶ **Claim 2.** *If $\mathcal{R}_{end} \neq \emptyset$, then at least Z_r elements are covered by $\mathcal{R}_{end}$.*

Proof. By assumption, we have that $\mathcal{R}_{\text{end}} \neq \emptyset$, i.e. $\mathcal{R}_{\text{end}} = \{S_i\}$ for some $S_i \in \mathcal{R}$. For each $S_l \in \mathcal{R}_r$ with $l \neq i$, we have that $x_l = 0$, again by the condition of the outer while loop. Since Constraint 6 is made tight for all elements in each execution of RoundTwoSets, for any element $e_{j'} \in X_r$ but $e_{j'} \notin S_i$, we have that $z_{j'} = 0$. On the other hand, for elements $e_j \in X_r \cap S_i$, we have that $z'_j = x'_i \leq \alpha$. If the number of such elements is p, then we have that $Z_r \leq \alpha \cdot p$. The lemma follows since choosing S_i covers all of these p elements, and $p \geq Z_r/\alpha \geq Z_r$. ◀

In the following lemma, we show that Algorithm 2 produces a feasible solution.

▶ **Lemma 8.** *The solution $\Sigma \cup \mathcal{R}_{end}$ returned by Algorithm 2 covers at least k elements.*

Proof. There are two cases – $\mathcal{R}_{\text{end}} = \emptyset$, or $\mathcal{R}_{\text{end}} = \{S_i\}$ for some $S_i \in \mathcal{R}$. In the first case, all elements in X_r are uncovered, and for all such elements, $e_{j'} \in X_r$, we have that $z_{j'} = 0$. In this case, it is trivially true that the number of elements of X_r covered by $\mathcal{R}_{\text{end}}$ is Z_r $(= 0)$. In the second case, the same follows from Claim 2. Therefore, in both cases we have that,

$$
\begin{aligned}
\text{Number of elements covered} &\geq |\Xi| + Z_r \\
&\geq \sum_{e_j \in \Xi} z'_j + \sum_{e_j \in X_r} z'_j && \text{(By Lemma 7 and } z'_j \leq 1) \\
&= \sum_{e_j \in X} z'_j \\
&\geq \sum_{e_j \in X} z_j && \text{(Lemma 5, Part 2)} \\
&\geq k && \text{(By Lemma 4 and Constraint 7)}
\end{aligned}
$$

Recall that z_j refers to the z-value of an element e_j in the optimal LP solution σ, at the beginning of the algorithm. ◀

▶ **Lemma 9.** *Let $\Sigma \cup \mathcal{R}_{end}$ be the solution returned by Algorithm 2, and let B be the weight of the heaviest set in $\mathcal{R}$. Let $\sigma = (x, z)$ and $\sigma' = (x', z')$ denote the* LP *solutions at the beginning and end of Algorithm 2, respectively. Then,*

1. $w(\Sigma) = \sum_{S_i \in \Sigma} w_i x'_i \leq \sum_{S_i \in \mathcal{R}} w_i x'_i \leq \frac{1}{\alpha} \sum_{S_i \in \mathcal{R}} w_i x_i = \frac{1}{\alpha} cost(\sigma)$,
2. $w(\mathcal{R}_{end}) \leq B$, *and*
3. $w(\Sigma \cup \mathcal{R}_{end}) \leq \frac{1}{\alpha} cost(\sigma) + B$.

Proof. For the first part, the inequality $\sum_{S_i \in \mathcal{R}} w_i x'_i \leq \frac{1}{\alpha} \sum_{S_i \in \mathcal{R}} w_i x_i$ follows because (a) RoundTwoSets preserves the cost of the LP solution, and (b) when a set S_i is added to Σ, its contribution to the cost of the LP increases by a factor of $\frac{1}{\alpha}$. For the second part, note that $\mathcal{R}_{\text{end}}$ contains at most one set $S_i \in \mathcal{R}$. By definition, weight of any set in S_i is bounded by B, the maximum weight of any set in $\mathcal{R}$. The third part follows from the first two parts. ◀

From Lemma 8 and Lemma 9, we conclude that $\Sigma \cup \mathcal{R}_{\text{end}}$ is a solution that covers at least $k = k_p - |X_1|$ elements from $X_p \setminus X_1$, and whose cost is at most $\frac{1}{\alpha} cost(\sigma) + B$.

5 A generalization of PSC

Consider the following generalization of the PSC problem, where the elements $e_j \in X'$ have profits $p_j \geq 0$ associated with them. Now the goal is to choose a minimum-weight collection $\Sigma \subseteq \mathcal{R}'$ such that the total profit of elements covered by the sets of Σ is at least K, where $0 \leq K \leq \sum_{e_j \in X} p_j$ is provided as an input. Note that setting $p_j = 1$ for all elements we get the original PSC problem. This generalization has been considered in [26].

It is easy to modify our algorithm that for PSC, such that it returns a $2\beta + 2$ approximate solution for this generalization as well. We briefly describe the modifications required. Firstly, we modify Constraint 7 of PSC LP to incorporate the profits as follows:

$$\sum_{e_j \in X} z_j \cdot p_j \geq K$$

The preprocessing and the rounding algorithms work with the straightforward modifications required to handle the profits. One significant change is in the rounding algorithm (Algorithm

2). We compare the "cost-effectiveness" of the two sets S_a, S_b in Line 7 for the PSC as $\frac{|S_a \cap X_{\text{cur}}|}{w_a} \geq \frac{|S_b \cap X_{\text{cur}}|}{w_b}$. For handling the profits of the elements, we replace this with the following condition – $\frac{P_a}{w_a} \geq \frac{P_b}{w_b}$, where $P_a := \sum_{e_j \in S_a \cap X_{\text{cur}}} p_j$, and $P_b := \sum_{e_j \in S_b \cap X_{\text{cur}}} p_j$. With similar straightforward modifications, the analysis of Algorithm 2 goes through with the same guarantee on the cost of the solution. We remark here that despite the profits, the approximation ratio only depends on that of the standard SC LP, which is oblivious to the profits.

References

1 Boris Aronov, Esther Ezra, and Micha Sharir. Small-size ε-nets for axis-parallel rectangles and boxes. *SIAM J. Comput.*, 39(7):3248–3282, 2010.

2 Reuven Bar-Yehuda. Using homogeneous weights for approximating the partial cover problem. *Journal of Algorithms*, 39(2):137–144, 2001.

3 Hervé Brönnimann and Michael T. Goodrich. Almost optimal set covers in finite vc-dimension. *Discrete & Computational Geometry*, 14(4):463–479, 1995.

4 Nader H. Bshouty and Lynn Burroughs. Massaging a linear programming solution to give a 2-approximation for a generalization of the vertex cover problem. In *STACS 98, 15th Annual Symposium on Theoretical Aspects of Computer Science, Paris, France, February 25-27, 1998, Proceedings*, pages 298–308, 1998.

5 Timothy M. Chan. Polynomial-time approximation schemes for packing and piercing fat objects. *J. Algorithms*, 46(2):178–189, 2003.

6 Timothy M. Chan, Elyot Grant, Jochen Könemann, and Malcolm Sharpe. Weighted capacitated, priority, and geometric set cover via improved quasi-uniform sampling. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1576–1585, 2012.

7 Timothy M. Chan and Nan Hu. Geometric red-blue set cover for unit squares and related problems. *Comput. Geom.*, 48(5):380–385, 2015.

8 Kenneth L. Clarkson. New applications of random sampling in computational geometry. *Discrete Comput. Geom.*, 2(1):195–222, 1987.

9 Kenneth L. Clarkson and Kasturi Varadarajan. Improved approximation algorithms for geometric set cover. *Discrete & Computational Geometry*, 37(1):43–58, 2007.

10 Irit Dinur and David Steurer. Analytical approach to parallel repetition. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*, STOC '14, pages 624–633, New York, NY, USA, 2014. ACM.

11 Khaled M. Elbassioni, Erik Krohn, Domagoj Matijevic, Julián Mestre, and Domagoj Severdija. Improved approximations for guarding 1.5-dimensional terrains. *Algorithmica*, 60(2):451–463, 2011.

12 Thomas Erlebach and Erik Jan Van Leeuwen. Ptas for weighted set cover on unit squares. In *Proceedings of the 13th International Conference on Approximation, and 14 the International Conference on Randomization, and Combinatorial Optimization: Algorithms and Techniques*, APPROX/RANDOM'10, pages 166–177, Berlin, Heidelberg, 2010. Springer-Verlag.

13 Guy Even, Dror Rawitz, and Shimon (Moni) Shahar. Hitting sets when the vc-dimension is small. *Inf. Process. Lett.*, 95(2):358–362, jul 2005.

14 Esther Ezra, Boris Aronov, and Micha Sharir. Improved bound for the union of fat triangles. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11, pages 1778–1785, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics.

15 Uriel Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

16 Toshihiro Fujito. On combinatorial approximation of covering 0-1 integer programs and partial set cover. *Journal of Combinatorial Optimization*, 8(4):439–452, 2001.

17 Rajiv Gandhi, Samir Khuller, and Srinivasan Aravind. Approximation algorithms for partial covering problems. *J. Algorithms*, 53(1):55–84, 2004.

18 Christian Glaßer, Christian Reitwießner, and Heinz Schmitz. Multiobjective disk cover admits a PTAS. In *Algorithms and Computation, 19th International Symposium, ISAAC 2008, Gold Coast, Australia, December 15-17, 2008. Proceedings*, pages 40–51, 2008.

19 Sathish Govindarajan, Rajiv Raman, Saurabh Ray, and Aniket Basu Roy. Packing and covering with non-piercing regions. In *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, pages 47:1–47:17, 2016.

20 David Haussler and Emo Welzl. epsilon-nets and simplex range queries. *Discrete & Computational Geometry*, 2:127–151, 1987.

21 Dorit S. Hochbaum. The t-vertex cover problem: Extending the half integrality framework with budget constraints. In *Proceedings of the International Workshop on Approximation Algorithms for Combinatorial Optimization*, APPROX '98, pages 111–122, London, UK, UK, 1998. Springer-Verlag.

22 Dorit S. Hochbaum and Wolfgang Maass. Approximation schemes for covering and packing problems in image processing and vlsi. *J. ACM*, 32(1):130–136, 1985.

23 Michael J. Kearns. *Computational Complexity of Machine Learning.* MIT Press, Cambridge, MA, USA, 1990.

24 Samir Khuller, Anna Moss, and Joseph Naor. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70(1):39–45, 1999.

25 James King and David Kirkpatrick. Improved approximation for guarding simple galleries from the perimeter. *Discrete Comput. Geom.*, 46(2):252–269, 2011.

26 Jochen Könemann, Ojas Parekh, and Danny Segev. A unified approach to approximating partial covering problems. *Algorithmica*, 59(4):489–509, 2011.

27 Erik Krohn, Matt Gibson, Gaurav Kanade, and Kasturi R. Varadarajan. Guarding terrains via local search. *JoCG*, 5(1):168–178, 2014.

28 Julián Mestre. A primal-dual approximation algorithm for partial vertex cover: Making educated guesses. *Algorithmica*, 55(1):227–239, 2009.

29 Nabil H. Mustafa and Saurabh Ray. Improved results on geometric hitting set problems. *Discrete & Computational Geometry*, 44(4):883–895, 2010.

30 Petr Slavík. Improved performance of the greedy algorithm for partial cover. *Inf. Process. Lett.*, 64(5):251–254, dec 1997.

31 Kasturi R. Varadarajan. Epsilon nets and union complexity. In *Proceedings of the 25th ACM Symposium on Computational Geometry, Aarhus, Denmark, June 8-10, 2009*, pages 11–16, 2009.

32 Kasturi R. Varadarajan. Weighted geometric set cover via quasi-uniform sampling. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 641–648, 2010.

33 Vijay V. Vazirani. *Approximation Algorithms.* Springer-Verlag New York, Inc., New York, NY, USA, 2001.