Noname manuscript No.
(will be inserted by the editor)

Algorithms for Covering Multiple Submodular Constraints and Applications

Chandra Chekuri · Tanmay Inamdar ⋆ · Kent Quanrud · Kasturi Varadarajan · Zhao Zhang ⋆⋆

the date of receipt and acceptance should be inserted later

Abstract We consider the problem of covering multiple submodular constraints. Given a finite ground set N, a weight function $w : N \to \mathbb{R}_+$, r monotone submodular functions $f_1, f_2, \ldots, f_r$ over N and requirements $k_1, k_2, \ldots, k_r$ the goal is to find a minimum weight subset $S \subseteq N$ such that $f_i(S) \geq k_i$ for $1 \leq i \leq r$. We refer to this problem as Mult-Submod-Cover and it was recently considered by Har-Peled and Jones [30] who were motivated by an application in geometry. Even with $r = 1$ Mult-Submod-Cover generalizes the well-known Submodular Set Cover problem (Submod-SC), and it can also be easily reduced to Submod-SC. A simple greedy algorithm gives an $O(\log(kr))$ approximation where $k = \sum_i k_i$ and this ratio cannot be improved in the general case.

In this paper, motivated by several concrete applications, we consider two ways to improve upon the approximation given by the greedy algorithm. First, we give a bicriteria approximation algorithm for Mult-Submod-Cover that covers each constraint to within a factor of $(1 - 1/e - \varepsilon)$ while incurring an approximation of $O(\frac{1}{\epsilon} \log r)$ in the cost. Second, we consider the special case

⋆ Corresponding Author. E-mail: tanmay-inamdar@uiowa.edu

⋆⋆ This paper combines and extends results from two previously unpublished manuscripts [34] and [20].

Chandra Chekuri E-mail: chekuri@illinois.edu
Dept. of Computer Science, University of Illinois, Urbana-Champaign, IL, 61820.
Tanmay Inamdar E-mail: tanmay-inamdar@uiowa.edu
Dept. of Computer Science, University of Iowa.
Kent Quanrud E-mail: krq@purdue.edu
Dept. of Computer Science, Purdue University, West Lafayette, Indiana, 47909. Work on this paper was mostly done while the author was at University of Illinois.
Kasturi Varadarajan E-mail: kasturi-varadarajan@uiowa.edu
Dept. of Computer Science, University of Iowa.
Zhao Zhang E-mail: zhaozhang@zjnu.edu.cn
College of Mathematics and Computer Science, Zhejiang Normal University, China. Work on this paper was done while the author was visiting University of Illinois.

when each f_i is a obtained from a truncated coverage function and obtain an algorithm that generalizes previous work on partial set cover (Partial-SC), covering integer programs (CIPs) and multiple vertex cover constraints [6]. Both these algorithms are based on mathematical programming relaxations that avoid the limitations of the greedy algorithm.

We demonstrate the implications of our algorithms and related ideas to several applications ranging from geometric covering problems to clustering with outliers. Our work highlights the utility of the high-level model and the lens of submodularity in addressing this class of covering problems.

Acknowledgments. Chandra Chekuri was partially supported by National Science Foundation (NSF) Grants CCF-1526799 and CCF-1910149. Work of Tanmay Inamdar and Kasturi Varadarajan was partially supported by NSF grants CCF-1318996 and CCF-1615845. Kent Quanrud was partially supported by NSF grant CCF-1526799. Work of Zhao Zhang was partially supported by National Science Foundation of China (NSFC) (11771013, 61751303, 11531011) and ZJ-NSFC (LD19A010001).

We thank Timothy Chan and Sariel Har-Peled for helpful comments and discussion on the problem formulation and applications.

1 Introduction

Set Cover is a well-studied problem in combinatorial optimization and is a canonical covering problem. The input is a set system $(\mathcal{U}, \mathcal{S})$ consisting of a finite set $\mathcal{U}$ and a collection $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$ of subsets of $\mathcal{U}$. The goal is to find a minimum cardinality subcollection $\mathcal{S}' \subseteq \mathcal{S}$ such that $\cup_{A \in \mathcal{S}'} A = \mathcal{U}$. In the weighted version each S_i has a weight $w_i \geq 0$ and the goal is to find a minimum weight subcollection of sets whose union is $\mathcal{U}$. Set Cover is NP-Hard and approximation algorithms have been extensively studied. A very simple greedy algorithm yields a $(1 + \ln d)$-approximation where $d = \max_i |S_i|$ even in the weighted case [25]. Moreover this bound is essentially tight unless $\mathsf{P} = \mathsf{NP}$ [24].

Various special cases and generalizations of Set Cover have been studied over the years for their applications and theoretical interest. We describe three generalizations that are of interest to us.

- **Partial Set Cover (**Partial-SC**)**: In Partial-SC the input is a set system $(\mathcal{U}, \mathcal{S})$ and an integer parameter k and the goal is to find a minimum (weight) subcollection of the given sets whose union is of size at least k. Set Cover is a special case when $k = |\mathcal{U}|$.
- **Covering Integer Program (**CIP**)**: A CIP is an integer program of the form $\min\{wx \mid Ax \geq b, x \leq d, x \in \mathbb{Z}_+^n\}$ where A is a non-negative $m \times n$ matrix and $b \geq 0$. Set Cover is a special case of CIP when A is a $\{0, 1\}$ matrix and b and d are the all ones vectors — each constraint row of A corresponds to covering an element of $\mathcal{U}$.

– **Submodular Set Cover (**Submod-SC**)**: In Submod-SC we are given a finite ground set N, a non-negative weight function $w : N \to \mathbb{R}_+$, and a polymatroid $f : 2^N \to \mathbb{Z}_+$ via a value oracle[1]. The goal is to find a minimum weight subset $S \subseteq N$ such that $f(S) = f(N)$. Set Cover is a special case where N represents the sets in the set system and f captures the coverage function which is submodular.

Submodularity is a powerful abstraction and Submod-SC can be seen to generalize Partial-SC and CIPs. The greedy algorithm for Set Cover admits a natural generalization to Submod-SC— Wolsey [46] showed that it yields a $(1 + \ln d)$-approximation where $d = \max_{i \in N} f(i)$. The abstraction of submodularity comes at a cost, however. For instance CIPs admit an $O(\ln m)$-approximation via an LP relaxation strengthened with knapsack cover (KC) inequalities [13, 19, 22, 36] while using the greedy algorithm yields only an $O(\ln d)$ approximation where d depends on the maximum sum of the entries in a column of A, and in fact can be as large as m [25]. CIPs provide the explicit ability to model multiple covering constraints and this is often useful in applications. In this paper we consider an abstraction that generalizes Submod-SC by explicitly allowing multiple submodular covering constraints.

Multiple Submodular Covering Constraints: The input consists of a ground set N and a weight function $w : N \to \mathbb{R}_+$. The input consists of r polymatroids $f_1, f_2, \ldots, f_r$ over N and integers $k_1, k_2, \ldots, k_r$. The goal is to find $S \subseteq N$ of minimum weight such that $f_i(S) \geq k_i$ for $1 \leq i \leq r$. We refer to this as Mult-Submod-Cover.

Har-Peled and Jones [30], motivated by an application from computational geometry, appear to be the first ones to consider Mult-Submod-Cover explicitly. As noted in [30], it is not hard to reduce Mult-Submod-Cover to Submod-SC. We simply define a new submodular set function $g : 2^N \to \mathbb{R}_+$ where $g(A) = \sum_{i=1}^{r} \min\{k_i, f_i(A)\}$. Via Wolsey's result for Submod-SC this implies an $O(\log r + \log K)$ approximation via the greedy algorithm where $K = \sum_{j=1}^{r} k_j$. Although Mult-Submod-Cover can be reduced to Submod-SC it is useful to treat it separately when the functions f_i are known to belong to a special class of submodular functions. For instance CIP can be seen as a special case of Mult-Submod-Cover where each f_i is a truncated/partial linear function. Another example, which is the main motivation for this work, comes from [6, 34] who considered the case when each f_i is a truncated/partial coverage function (partial vertex cover in [6] and partial set cover in [34]). These special cases have several applications that we outline below.

We mention that prior work has considered multiple submodular objectives from a *maximization* perspective [18, 21] rather than from a minimum cost perspective. There are useful connections between these two perspectives. Consider Submod-SC. We could recast the exact version of this problem as

[1] A polymatroid is an integer valued monotone submodular function that is also normalized ($f(\emptyset) = 0$). A real-valued set function $f : 2^N \to \mathbb{R}$ is submodular iff $f(A \cup B) + f(A \cap B) \leq f(A) + f(B)$ for all $A, B \subseteq N$. A set function is monotone if $f(A) \leq f(B)$ for all $A \subset B$. A value oracle for f outputs $f(A)$ when queried with a set $A \subseteq N$.

$\max f(S)$ subject to the constraint $w(S) \leq \text{OPT}$ where OPT is the optimum cost of the given instance of Submod-SC. This is submodular function maximization subject to a knapsack constraint and admits a $(1 - 1/e)$-approximation [40]. Using this algorithm iterarively will also yield an approximation algorithm for Submod-SC.

We describe several applications that motive this and some previous work and then state our results formally.

1.1 Motivating Applications

Splitting point sets: Har-Peled and Jones [30], as we remarked, were motivated to study Mult-Submod-Cover due to a geometric application that has connections to the classical Ham-Sandwich theorem as well as problems in feature selection in machine learning. Their problem is the following. Given m point sets $P_1, \ldots, P_m$ in $\mathbb{R}^d$ they wish to find the smallest number of hyperplanes (or other geometric shapes) such that no point set P_i has more than a constant factor of its points in any cell of the arrangement induced by the chosen hyperplanes; in particular when the constant is a half, the problem is related to the Ham-Sandwich theorem which implies that when $m \leq d$ just one hyperplane suffices! From this one can infer that $\lceil m/d \rceil$ hyperplanes always suffice. However for a given instance it may be possible to do much better. In [30] the authors considered the problem of approximating the smallest number of hyperplanes required for the desired partitioning and reduced this problem to Mult-Submod-Cover. Via Wolsey's greedy algorithm they obtain an $O(\log m + \log N)$ approximation where N is the total number of points. In applications N is likely to be large while m is likely to be quite small and hence an approximation that does not depend on N is desirable. It is not obvious how to model this problem as a special case of Mult-Submod-Cover— [30] describes one such reduction and we describe a slightly different one later in the paper.

Multiple Partial Set Cover Constraints in Geometric Settings: There has been extensive work on Set Cover specialized to geometric settings via sophisticated techniques [7, 14, 23, 38, 41, 42]. Consider for example the problem of covering a given collection of points in the plane by a minimum weight subcollection of a given collection of weighted disks. This admits a constant factor approximation via the natural LP relaxation [14] in contrast to the logarithmic integrality gap and hardness known for general Set Cover instances. A natural question is whether this improved result also holds for Partial-SC version; here we are only required to cover k of the given points rather than all of them. Inamdar and Varadarajan [33] developed a simple and elegant black box technique for this purpose via a standard LP relaxation. They show that if there is a β-approximation for a deletion-closed class of Set

Cover instances[2] via the standard LP, then there is $2(\beta+1)$ approximation for Partial-SC for the same family. A natural extension of Partial-SC is to have multiple constraints. Consider the setting where the points are colored by r colors (equivalently they are partitioned into r sets) and the goal is to find the minimum weight subset of a given collection of disks such that at least k_i points from color class i are covered; one can also consider the setting where the color classes are not disjoint. Bera et al. [6] considered multiple partial covering constraints in the restricted setting of Vertex Cover and obtained an $O(\log r)$-approximation. This was generalized in [31] to instances of Set Cover with maximum frequency Δ to obtain a $(\Delta H_r + H_r)$-approximation via a primal-dual algorithm. A natural open question here was whether one can obtain an $O(\Delta + \log r)$-approximation and whether one can generalize further and obtain an $O(\beta + \log r)$-approximation for all deletion-closed families of Set Cover that admit a β-approximation. We refer to this problem as Multi-Partial-SC. Note Multi-Partial-SC is a special case of Mult-Submod-Cover where each f_i is a truncated coverage function (equivalently a partial set cover function).

Now we discuss two geometric variants of Set Cover that induce deletion-closed set systems, and for which β is known to be sub-logarithmic in some special settings. In HittingSet, we are given a collection of geometric objects $\mathcal{U}$ and a collection of points $\mathcal{P}$. If a point $p \in \mathcal{P}$ is contained in an object $U \in \mathcal{U}$, then U is said to be hit by p. In the weighted version, each point has a non-negative weight. The goal is to find a minimum-weight set of points that hits all objects from $\mathcal{U}$. In the Geometric DominatingSet, we are given an intersection graph $G = (V, E)$ of geometric objects such as disks, with non-negative weights on vertices. A vertex v is said to dominate itself and its neighbors. The goal is to find a minimum-weight subset of vertices V' that dominates at all vertices from V. In the partial version of DominatingSet (resp. HittingSet), the goal is to dominate at least k vertices (resp. hit at least k objects). We summarize known results for Geometric Set Cover, HittingSet and Geometric DominatingSet in the following table.

Problem	$\mathcal{U}$	Geometric objects	β
Set Cover	Points in $\mathbb{R}^2$	Disks	$O(1)$
		Fat triangles	$O(\log \log^* n)$
	Points in $\mathbb{R}^3$	Unit cubes	$O(1)$
		Halfspaces	$O(1)$
HittingSet	Rectangles in $\mathbb{R}^3$	Points	$O(\log \log n)$
DominatingSet	Disks in $\mathbb{R}^2$		$O(1)$

Table 1: LP-based bounds for Set Cover and related geometric covering problems. See [3, 14, 23, 26, 27, 29, 42] for the references establishing these bounds.

[2] We say that a family of set systems is *deletion closed* if removing an element or removing a set from a set system in the family yields another set system in the same family.

Facility Location with Multiple Outliers: Facility location is an extensively studied problem and there are several variants. In the basic Uncapacitated Facility Location problem (which we abbreviate to Facility Location) the input consists of a set of facilities F and a set of clients C in a metric space $(F \cup C, d)$. Each facility $i \in F$ has a non-negative opening cost f_i. The goal is to open a set of facilities and connect the clients to them to minimize the sum of the opening costs of the facilities plus the sum of the distances of each client to the nearest open facility — mathematically we want to find $S \subseteq F$ to minimize $\sum_{i \in S} f_i + \sum_{j \in C} d(j, S)$. In many scenarios there are outliers and instead of asking for all the clients to be connected we only seek to connect some specified number k of clients — this has been studied under the name the Robust Facility Location problem by Charikar et al. [16] who obtained a constant factor approximation. We consider here the setting of multiple outlier classes. We have r disjoint classes of clients $C_1, C_2, \ldots, C_r$ and we need to connect to the open facilities some specified number b_i of clients from C_i for $1 \leq i \leq r$. An $O(r)$-approximation is easy by considering each client class separately but the natural question is whether we can find an $O(\log r)$-approximation; via a reduction from Set Cover one can show an $\Omega(\log r)$ lower bound on the approximability of this problem. We refer to Facility Location with Multiple Outliers as FL-Multi-Ouliers. We note that FL-Multi-Ouliers is not a special case of Mult-Submod-Cover since the objective function has both facility opening cost as well as client connection cost. Nevertheless, the problem is sufficiently close to Mult-Submod-Cover that the techniques we develop are applicable to this problem as well.

We also consider a related problem of clustering to minimize the sum of radii. This problem was considered by Charikar et al. [17] who gave a constant approximation using a primal-dual algorithm. A constant approximation for the outlier version was given by Ahmadian et al. [2]. We consider a further generalization of covering r classes of clients, while minimizing the sum of radii. We call this problem MCC-Multi-Outliers. We formally define MCC-Multi-Outliers in Section 5, and give a tight $O(\log r)$-approximation using similar techniques.

1.2 Results and contributions

In this paper we examine approximation algorithms for Mult-Submod-Cover and Multi-Partial-SC and the motivating applications. Instead of relying on the greedy algorithm we use mathematical programming based approaches and tools from continuous extensions of submodular extensions that allow us to handle the special cases of interest that arise from the applications. In addition to the technical results we showcase the utility of the models in capturing interesting applications. Our algorithmic results are summarized below.

- For Mult-Submod-Cover we obtain a bicriteria approximation. We obtain a random solution S such that $f_i(S) \geq (1 - 1/e - \varepsilon)k_i$ for $1 \leq i \leq r$ and the expected weight of S is $O(\frac{1}{\varepsilon} \log r)\mathrm{OPT}$. We obtain the same bound even

in a more general setting where the system of constraints is r-sparse. We apply this result to the splitting points application and obtain an $O(\log m)$ bicriteria approximation that suffices in many scenarios. This improves the $O(\log m + \log N)$ approximation obtained in [30].
- We consider a simultaneous generalization of Multi-Partial-SC and CIPs for deletion-closed set systems that admit a β-approximation for Set Cover via the natural LP. We obtain a randomized $O(\beta + \log r)$ approximation where r is the sparsity of the system. This generalizes and improves bounds for multiple covering versions of Vertex-Cover from [6, 31]. In particular, we obtain $O(\Delta + \log r)$-approximation for Multi-Partial-SC in the set systems with maximum frequency Δ, improving on $H_r(\Delta + 1)$-approximation by [31]. Furthermore, we obtain $O(\beta+\log r)$-approximations for several geometric Multi-Partial-SC problems, where β is known to be sublogarithmic or constant (cf. Table 1).
- We obtain $O(\log r)$ approximations for FL-Multi-Ouliers and MCC-Multi-Outliers generalizing the previous bounds for one class of outliers to multiple classes of outliers. As noted before, these bounds are tight up to constant factors via simple reductions from Set Cover.
- For deletion-closed set systems that have a β-approximation (cf. Table 1) to Set Cover via the natural LP we obtain a $\frac{e}{e-1}(\beta + 1)$-approximation for Partial-SC. This slightly improves the bound of $2(\beta+1)$ in [33] while also simplifying the algorithm and analysis.

A brief discussion of technical ideas: Mult-Submod-Cover admits a reduction to Submod-SC for which the greedy algorithm is a known approach. To obtain our bicriteria approximation we take a different approach based on the multilinear relaxation or submodular functions which plays a fundamental role in submodular function maximization algorithms.

For addressing Multi-Partial-SC and its generalization we follow the high-level approach used already in the special setting of Vertex-Cover by Bera et al. [6] — they used an LP relaxation strengthened with knapsack cover inequalities. We bring two technical ingredients to bear on this problem. First we extend a probabilistic inequality used in [6] to the general set cover setting and this is not obvious. We provide a proof that relies on continuous extensions of submodular functions and certain concentration properties which we believe provides a clean and transparent explanation. Second, we use randomized rounding plus alteration to extend the results to the sparse setting — this is inspired by recent work on CIPs [19].

Finally, for Partial-SC we simplify the algorithm and analysis from [33] via connections to submodular function maximization and continuous extensions.

We believe that the problems, applications and technical tools that we demonstrate in this paper are likely to be useful for other problems in the future.

Other related work: Partial-SC has been well-studied in the past for special cases such as the Partial Vertex Cover (PartialVC) where the goal is to find a minimum weight subset of nodes in a graph to cover at least k edges. There are several 2-approximations known for PartialVC [4, 8, 28]. More generally, for set systems with maximum frequency Δ, similar techniques give $O(\Delta)$-approximations [6, 28, 37]. Surprisingly, the black box reduction of [33] from Partial-SC to Set Cover via the LP relaxation, that we mentioned earlier, is fairly recent. In some restricted geometric settings, PTASes — polynomial time $(1+\epsilon)$-approximations for any constant $\epsilon > 0$ — are known via the shifting technique and local search [15, 28, 32]. CIPs have been studied extensively for several years starting with the work of Dobson [25]. The introduction of KC inequalities [13] led to the first $O(\log m)$-approximation by Kolliopoulos and Young [36]. Recent work [19, 22] has obtained sharp bounds that depend on the ℓ_0 and ℓ_1 sparsity of the matrix A. Constrained submodular set function optimization (maximization and minimization) has been a topic of much interest in recent years and it is difficult to provide a concise summary. Continuous extensions of submodular functions and mathematical programming approachs have played an important role. We refer the reader to [9] for a survey on submodular function maximization which provides several pointers. The literature on clustering and facility location problems is vast. We are motivated by work on approximation algorithms for handling outliers and generalizing it to handle multiple groups of client. Many papers on clustering are in the model where the number of clusters k is specified and different objectives lead to well-studied problems such as k-median, k-means and k-center. Recent work on fair clustering (see [5] and pointers) has also considered multiple groups of clients. The specific problems we consider and techniques we use are different. We leave it to future work to better understand the relationship between clustering with fairness constraints and clustering with outliers.

Organization: In Section 2, we introduce necessary background on other related problems and submodular functions. In Section 3, we give a bicriteria approximation algorithm to Mult-Submod-Cover, and apply it for a geometric problem in 3.1. We consider the special case of Multi-Partial-SC in Section 4. Next, we adapt these techniques to obtain similar results for FL-Multi-Ouliers and MCC-Multi-Outliers in Section 5. In Section 6, we sketch a proof of the improved approximation for Partial-SC, using some of the similar techniques used elsewhere in the paper. We conclude in Section 7 with some open problems.

2 Preliminaries and Background

Set Cover and Partial-SC have natural LP relaxations and they are closely related to those for Max k-Cover and Max-Budgeted-Cover. The LP relaxation for Set Cover (SC-LP) is shown in Fig 1a. It has a variable x_i for each set $S_i \in \mathcal{S}$, which, in the integer programming formulation, indicates

whether S_i is picked in the solution. The goal is to minimize the weight of the chosen sets which is captured by the objective $\sum_{S_i \in \mathcal{S}} w_i x_i$ subject to the constraint that each element e_j is covered. The LP relaxation for Partial-SC (PSC-LP) is shown in Fig 1b. Now we need additional variables to indicate which k elements are going to be covered; for each $e_j \in \mathcal{U}$ we thus have a variable z_j for this purpose. In PSC-LP it is important to constrain z_j to be at most 1. The constraint $\sum_{e_j} z_j \geq k$ forces at least k elements to be covered fractionally.

(SC-LP)

$$\min \sum_{S_i \in \mathcal{S}} w_i x_i$$

$$\sum_{i: e_j \in S_i} x_i \geq 1 \quad e_j \in \mathcal{U}$$

$$x_i \geq 0 \qquad S_i \in \mathcal{S}$$

(a) Set Cover

(PSC-LP)

$$\min \sum_{S_i \in \mathcal{S}} w_i x_i$$

$$\sum_{i: e_j \in S_i} x_i \geq z_j \quad e_j \in \mathcal{U}$$

$$\sum_{e_j \in \mathcal{U}} z_j \geq k$$

$$z_j \in [0,1] \quad e_j \in \mathcal{U}$$

$$x_i \geq 0 \qquad S_i \in \mathcal{S}$$

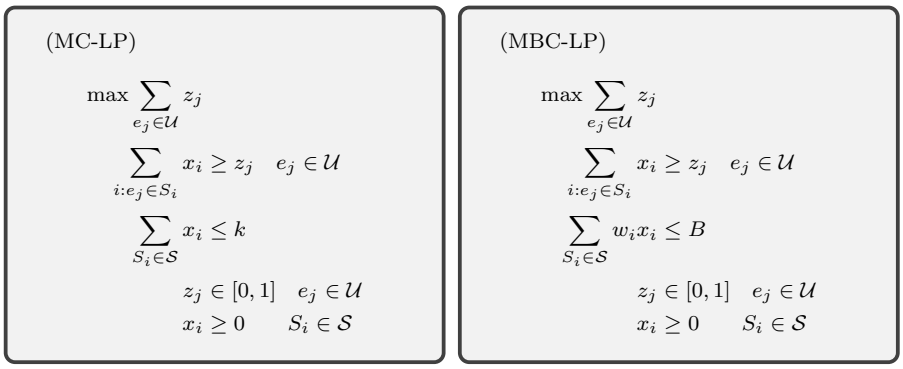
(b) Partial-SC

Fig. 1: LP Relaxations for covering.

As noted in prior work the integrality gap of PSC-LP can be made arbitrarily large but it is easy to fix by guessing the largest cost set in an optimum solution and doing some preprocessing. We discuss this issue in later sections.

Figs 2a and 2b show LP relaxations for Max k-Cover and Max-Budgeted-Cover respectively. In these problems we maximize the number of elements covered subject to an upper bound on the number of sets or on the total weight of the chosen sets.

Greedy algorithm: The greedy algorithm is a well-known and standard algorithm for the problems studied here. The algorithm iteratively picks the set with the current maximum bang-per-buck ratio and adds it to the current solution until some stopping condition is met. The bang-per-buck of a set S_i is defined as $|S_i \cap \mathcal{U}'|/w_i$ where $\mathcal{U}'$ is the set of uncovered elements at that point in the algorithm. For minimization problems such as Set Cover and Partial-SC the algorithm is stopped when the required number of elements are covered. For Max k-Cover and Max-Budgeted-Cover the algorithm is stopped when if adding the current set would exceed the budget. Since this is a standard algorithm that is extremely well-studied we do not describe all the formal details and the known results. Typically the approximation guarantee



(a) Max k-Cover (b) Max-Budgeted-Cover

Fig. 2: LP Relaxations for budgeted covering.

of Greedy is analyzed with respect to an optimum integer solution. We need to compare it to the value of the fractional solution. For the setting of the cardinality constraint this was already done in [39]. We need a slight generalization to the budgeted setting and we give a proof for the sake of completeness.

Lemma 2.1. *Let Z be the optimum value of (*MBC-LP*) for a given instance of* Max-Budgeted-Cover *with budget B.*

- *Suppose Greedy algorithm is run until the total weight of the chosen sets is equal to or exceeds B. Then the number of elements covered by greedy is at least $(1 - 1/e)Z$.*
- *Suppose no set covers more than cZ elements for some $c > 0$ then the weight of sets chosen by Greedy to cover $(1 - 1/e)Z$ elements is at most $(1 + ec)B$.*

Proof. We give a short sketch. Greedy's analysis for Max-Budgeted-Cover is based on the following key observation. Consider the first set S picked by Greedy. Then $|S|/w(S) \geq \mathrm{OPT}/B$ where OPT is the value of an optimum integer solution. And this follows from submodularity of the coverage function. This observation is applied iteratively with the residual solution as sets are picked and a standard analysis shows that when Greedy first meets or exceeds the budget B then the total number of elements covered is at least $(1-1/e)\mathrm{OPT}$. We claim that we can replace OPT in the analysis by Z. Given a fractional solution x, z we see that $Z = \sum_e z_e \leq \sum_{e \in \mathcal{U}} \min\{1, \sum_{i:e \in S_i} x_i\}$. Moreover $\sum_i w_i x_i \leq B$. Via simple algebra, we can obtain a contradiction if $|S_i|/w_i < Z/B$ holds for all sets S_i. Once we have this property the rest of the analysis is very similar to the standard one where OPT is replaced by Z.

Now consider the case when no set covers more than cZ elements. If Greedy covers $(1 - 1/e)Z$ elements before the weight of sets chosen exceeds B then there is nothing to prove. Otherwise let S_j be the set added by Greedy when

its weight exceeds B for the first time. Let $\alpha \leq |S_j|$ be the number of new elements covered by the inclusion of S_j. Since Greedy had covered less than $(1-1/e)Z$ elements, the value of the residual fractional solution is at least Z/e. From the same argument as the in the preceding paragraph, since Greedy chooses S_j at that point, $\frac{\alpha}{w(S_j)} \geq \frac{Z}{eB}$. This implies that $w(S_j) \leq eB\frac{\alpha}{Z} \leq ecB$. Since Greedy covers at least $(1-1/e)Z$ elements after choosing S_j (follows from the first claim of the lemma), the total weight of the sets chosen by Greedy is at most $B + w(S_j) \leq (1+ec)B$. □

We note that the conclusions of the preceding lemma hold even for the following generalization of Max-Budgeted-Cover. Here each element $e \in \mathcal{U}$ has a non-negative "profit" p_e associated with it, and the goal is to find a collection of sets with weight at most B, such that the overall profit of the covered elements is maximized. One difference in the argument is that the "bang-per-buck" of a set S is defined as $\frac{\sum_{e\in S} p_e}{w(S)}$.

2.1 Submodular set functions and continuous extensions

Continuous extensions of submodular set functions have played an important role in algorithmic and structural aspects. The idea is to extend a discrete set function $f : 2^N \to \mathbb{R}$ to the continous space $[0,1]^N$. Here we are mainly concerned with extensions motivated by maximization problems, and confine our attention to two extensions and refer the interested reader to [11, 43] for a more detailed discussion.

The *multilinear extension* of a real-valued set function $f : 2^N \to \mathbb{R}$, denoted by F, is defined as follows: For $x \in [0,1]^N$

$$F(x) = \sum_{S \subseteq N} f(S) \prod_{i \in S} x_i \prod_{j \notin S} (1 - x_j).$$

Equivalently $F(x) = \mathrm{E}[f(R)]$ where R is a random set obtained by picking each $i \in N$ independently with probability x_i.

The *concave closure* of a real-valued set function $f : 2^N \to \mathbb{R}$, denoted by f^+, is defined as the optimum of an exponential sized linear program:

$$f^+(x) = \max \sum_{S \subseteq N} f(S)\alpha_S \quad \text{s.t} \sum_{S \subseteq N} \alpha_S = 1, \quad \sum_{S \ni i} \alpha_S = x_i, \quad \forall i \in N \text{ and } \alpha_S \geq 0 \quad \forall S.$$

A special case of submodular functions are non-negative weighted sums of rank functions of matroids. More formally suppose N is a finite ground set and $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_\ell$ are ℓ matroids on the same ground set N. Let $g_1, \ldots, g_\ell$ be the rank functions of the matroids and these are monotone submodular. Suppose $f = \sum_{h=1}^{\ell} w_h g_h$ where $w_h \geq 0$ for all $h \in [\ell]$, then f is monotone submodular. We note that (weighted) coverage functions belongs to this class. For a such a submodular function we can consider an extension $\tilde{f}$ where $\tilde{f}(x) = \sum_h w_h g^+(x)$. We capture two useful facts which are shown in [11].

Lemma 2.2 ([11]). *Suppose $f = \sum_{h=1}^{\ell} w_h g_h$ is the weighted sum of rank functions of matroids. Then $F(x) \geq (1-1/e)\tilde{f}(x)$. Assuming oracle access to the rank functions $g_1, \ldots, g_\ell$, for any $x \in [0,1]^N$, there is a polynomial-time solvable LP whose optimum value is $\tilde{f}(x)$.*

Remark 2.3. *Let $f : 2^{\mathcal{S}} \to \mathbb{Z}_+$ be the coverage function associated with a set system $(\mathcal{U}, \mathcal{S})$. Then $\tilde{f}(x) = \sum_{e \in \mathcal{U}} \min\{1, \sum_{i:e \in S_i} x_i\}$ where $\tilde{f} = \sum_{e \in \mathcal{U}} g_e^+$ and $g_e(x) = \min\{1, \sum_{i:e \in S_i} x_i\}$ is the rank function of a simple uniform matroid. One can see* PSC-LP *in a more compact fashion:*

$$\min \sum_i w_i x_i \quad s.t \quad \tilde{f}(x) \geq k.$$

Concentration under randomized rounding: Recall the multilinear extension F of a submodular function f. If $x \in [0,1]^N$ then $F(x) = \mathrm{E}[f(R)]$ where R is a random set obtained by independently including each $i \in N$ in R with probability x_i. We can ask whether $f(R)$ is concentrated around $\mathrm{E}[f(R)] = F(x)$. And indeed this is the case when f is Lipscitz. For a parameter $c \geq 0$, f is c-Lipschitz if $|f_A(i)| \leq c$ for all $i \in N$ and $A \subset N$, where $f_A(i) = f(A \cup \{i\}) - f(A)$; for monotone functions this is equivalent to the condition that $f(i) \leq c$ for all $i \in N$.

Lemma 2.4 ([45]). *Let $f : 2^N \to \mathbb{R}_+$ be a 1-Lipschitz monotone submodular function. For $x \in [0,1]^N$ let R be a random set drawn from the product distribution induced by x. Then for $\delta \geq 0$,*

- $\Pr[f(R) \geq (1+\delta)F(x)] \leq (\frac{e^\delta}{(1+\delta)^{(1+\delta)}})^{F(x)}$.
- $\Pr[f(R) \leq (1-\delta)F(x)] \leq e^{-\delta^2 F(x)/2}$.

Greedy algorithm under a knapsack constraint: Consider the problem of maximizing a monotone submodular function subject to a knapsack constraint; formally $\max f(S)$ s.t $w(S) \leq B$ where $w : N \to \mathbb{R}_+$ is a non-negative weight function on the elements of the ground set N. Note that when all $w(i) = 1$ and $B = k$ this is the problem of maximizing a monotone submodular function subject to a cardinality constraint. For the cardinality constraint case, the simple greedy algorithm that iteratively picks the element with the largest marginal value yields a $(1-1/e)$-approximation [39]. Greedy extends in a natural fashion to the knapsack constraint setting; in each iteration the element $i = \arg\max_j f_S(j)/w_j$ is chosen where S is the set of already chosen elements (here, $f_S(j) = f(S \cup \{j\}) - f(S)$ denotes the marginal increase). Sviridenko [40], building on earlier work on the coverage function [35], showed that greedy with some partial enumeration yields a $(1-1/e)$-approximation for the knapsack constraint. The following lemma quantifies the performance of the basic Greedy when it is stopped after meeting or exceeding the budget B.

Lemma 2.5. *Consider an instance of monotone submodular function maximization subject to a knapsack constraint. Let Z be the optimum value for the given knapsack budget B. Suppose the greedy algorithm is run until the total weight of the chosen sets is equal to or exceeds B. Letting S be the greedy solution we have $f(S) \geq (1-1/e)Z$.*

3 A bicriteria approximation for Mult-Submod-Cover

In this section we consider Mult-Submod-Cover. Let N be a finite ground set. For each $j \in [h]$ we are given a submodular function $f_j : 2^N \to \mathbb{R}_+$. We are also given a non-negative weight function $w : N \to \mathbb{R}_+$. The goal is to solve the following covering problem:

$$\min_{S \subseteq N} w(S) \quad \text{s.t}$$
$$f_j(S) \geq k_j \quad 1 \leq j \leq h$$

We say that $i \in N$ is active in constraint j if $f_j(i) > 0$, otherwise it is inactive. We say that the given instance is r-sparse if each element $i \in N$ is active in at most r constraints.

Theorem 3.1. *There is a randomized polynomial-time approximation algorithm that given an r-sparse instance of* Mult-Submod-Cover *outputs a set $S \subseteq N$ such that (i) $f_j(S) \geq (1 - 1/e - \varepsilon)k_j$ for $1 \leq j \leq h$, and (ii) $E[w(S)] = O(\frac{1}{\varepsilon} \ln r)\mathrm{OPT}$.*

The rest of the section is devoted to the proof of the preceding theorem. We will assume without loss of generality that $k_i = 1$ for each i which can be arranged by scaling. Also, we will assume that $f_i(N) \leq 1$; otherwise we can work with the truncated function $\min\{1, f_i(S)\}$ which is also submodular. This technical assumption plays a role in the analysis later.

We consider a continuous relaxation of the problem based on the multilinear extension. Instead of finding a set S we consider finding a fractional point $x \in [0,1]^N$. For any value $B \geq \mathrm{OPT}$ where OPT is the optimum value of the original problem, the following continuous optimization problem has a feasible solution.

$$\text{(MP-Submod-Relax)} \quad \sum_i w_i x_i \leq B$$
$$F_j(x) \geq 1 \quad 1 \leq j \leq h$$
$$x \geq 0$$

One cannot hope to solve the preceding continuous optimization problem since it is NP-Hard. However the following approximation result is known and is based on extending the continuous greedy algorithm of Vondrak [12, 44].

Theorem 3.2 ([18, 21])**.** *There is a randomized polynomial-time algorithm that given an instance of MP-Submod-Relax and value oracle access to the submodular functions $f_1, \ldots, f_h$, with high probability, either correctly outputs that the instance is not feasible or outputs an x such that (i) $\sum_i w_i x_i \leq B$ and (ii) $F_i(x) \geq (1 - 1/e - \varepsilon)$ for $1 \leq i \leq h$.*

Using the preceding theorem and binary search one can obtain an x such that $\sum_{i \in N} w_i x_i \leq \mathrm{OPT}$ and $F_j(x) \geq (1 - 1/e - \varepsilon)$ for $1 \leq j \leq h$. It remains to round this solution. We use the following algorithm based on the high-level framework of randomized rounding plus alteration.

1. Let $S_1, S_2, \ldots, S_\ell$ be random sets obtained by picking elements independently and randomly ℓ times according to the fractional solution x. Let $S = \cup_{k=1}^{\ell} S_k$.
2. For each $j \in [h]$ if $f(S) < (1-1/e-2\varepsilon)$, fix the constraint. That is, find a set T_j using the greedy algorithm (via Lemma 2.5) such that $f_j(T_j) \geq (1-1/e)$. We implicitly set $T_j = \emptyset$ if $f(S) \geq (1 - 1/e - 2\varepsilon)$.
3. Output $S \cup T$ where $T = \cup_{j=1}^{h} T_j$.

It is easy to see that $S \cup T$ satisfies the property that $f_j(S \cup T) \geq (1 - 1/e - 2\varepsilon)$ for $j \in [h]$. It remains to choose ℓ and bound the expected cost of $S \cup T$.

The following is easy from randomized rounding stage of the algorithm.

Lemma 3.3. $\mathrm{E}[w(S)] = \ell \sum_{i=1}^{h} w_i x_i \leq \ell \mathrm{OPT}$.

We now bound the probability that any fixed constraint is not satisfied after the randomized rounding stage of the algorithm. Let I_j be the indicator for the event that $f_j(S) < (1 - 1/e - 2\varepsilon)$.

Lemma 3.4. *For any $j \in [h]$, $\Pr[I_j] \leq \alpha^\ell$, where $\alpha \leq 1 - \varepsilon$ for sufficiently small $\varepsilon > 0$.*

Proof. Let $I_{j,k}$ be indicator for the event that $f_j(S_k) < (1 - 1/e - 2\varepsilon)$. From the definition of the multilinear extension, for any $k \in [\ell]$, $\mathrm{E}[f_j(S_k)] = F_j(x)$. Hence, $\mathrm{E}[f_j(S_k)] \geq (1 - 1/e - \varepsilon)$. Let $\alpha = \Pr[I_{j,k}]$. We upper bound α as follows. Recall that $f_j(N) \leq 1$ and hence by monotonicity we have $f_j(A) \leq 1$ for all $A \subseteq N$. Since $\mathrm{E}[f_j(S_k)] \geq (1 - 1/e - \varepsilon)$ we can upper bound α by the following:

$$\alpha(1 - 1/e - 2\varepsilon) + (1 - \alpha) \geq (1 - 1/e - \varepsilon).$$

Rearranging we have $\alpha \leq \frac{(1+e\varepsilon)}{(1+2e\varepsilon)} = \frac{1}{1+\frac{e\varepsilon}{1+e\varepsilon}}$. Using the fact that for $\frac{1}{1+x} \leq 1 - x/2$ for sufficiently small $x > 0$, we simplify and see that $\alpha \leq 1 - \frac{e\varepsilon}{2(1+e\varepsilon)} \leq 1 - \varepsilon$ for sufficiently small $\varepsilon > 0$. Since the sets $S_1, \ldots, S_\ell$ are chosen independently,

$$\Pr[I_j] \leq \prod_{k=1}^{\ell} \Pr[I_{j,k}] \leq \alpha^\ell.$$

□

Remark 3.5. *The simplicity of the previous proof is based on the use of the multilinear extension which is well-suited for randomized rounding. The assumption that $f_j(N) \leq 1$ is technically important and it is easy to ensure in the general submodular case but is not straightforward when working with specific classes of functions.*

Lemma 3.6. *Let OPT_j be the value of an optimum solution to the problem $\min w(S)$ s.t $f_j(S) \geq 1$. Then, $\sum_{j=1}^{h} \mathrm{OPT}_j \leq r\mathrm{OPT}$.*

Proof. Let S^* be an optimum solution to the problem of covering all h constraints. Let N_j be the set of active elements for constraint j. It follows that $S^* \cap N_j$ is a feasible solution for the problem of covering just f_j. Thus $\mathrm{OPT}_j \leq w(S^* \cap N_j)$. Hence

$$\sum_j \mathrm{OPT}_j \leq \sum_j w(S^* \cap N_j) = \sum_{i \in S^*} w_i \sum_{j : i \in N_j} 1 \leq r w(S^*) = r \cdot \mathrm{OPT}.$$

□

We now bound the expected cost of T

Lemma 3.7. $\mathrm{E}[w(T)] \leq 2\alpha^\ell \sum_j \mathrm{OPT}_j \leq 2\alpha^\ell r \mathrm{OPT}$.

Proof. We claim that $w(T_j) \leq 2\mathrm{OPT}_j$. Assuming the claim, from the description of the algorithm, we have

$$\mathrm{E}[w(T)] \leq \sum_{j=1}^{h} \Pr[I_j] w(T_j) \leq 2\alpha^\ell \sum_j \mathrm{OPT}_j \leq 2\alpha^\ell r \mathrm{OPT}.$$

Now we prove the claim. Consider the problem $\min w(S)$ s.t $f_j(S) \geq 1$. OPT_j is the optimum solution value to this problem. Now consider the following submodular function maximization problem subject to a knapsack constraint: $\max f_j(S)$ s.t $w(S) \leq \mathrm{OPT}_j$. Clearly the optimum value of this maximization problem is at least 1. From Lemma 2.1, the greedy algorithm when run on the maximization problem, outputs a solution T_j such that $f(T_j) \geq (1 - 1/e)$ and $w(T_j) \leq \mathrm{OPT}_j + \max_i w_i$. By guessing the maximum weight element in an optimum solution to the maximization problem we can ensure that $\max_i w_i \leq \mathrm{OPT}_j$. Thus, $w(T_j) \leq 2\mathrm{OPT}_j$ and $f(T_j) \geq (1 - 1/e)$. □

From the preceding lemmas it follows that

$$\mathrm{E}[w(S \cup T)] \leq \mathrm{E}[w(S)] + \mathrm{E}[w(T)] \leq \ell \mathrm{OPT} + 2\alpha^\ell r \cdot \mathrm{OPT}.$$

If we set $\ell = \lceil \log_{1/\alpha} r \rceil = O(\frac{1}{\varepsilon} \ln r)$, one can see that $\mathrm{E}[w(S \cup T)] \leq O(\frac{1}{\varepsilon} \ln r)\mathrm{OPT}$.

3.1 An application to splitting point sets

Har-Peled and Jones [30], as we remarked, were motivated to study Mult-Submod-Cover due to a geometric application. We recall the problem. Given m point sets $P_1, \ldots, P_m$ in $\mathbb{R}^d$ find the smallest number of hyperplanes (or other geometric shapes) such that no point set P_i has more than α fraction of its points in any cell of the arrangement induced by the chosen hyperplanes; in particular when $\alpha = 1/2$ the problem is related to the Ham-Sandwich theorem which implies that when $m \leq d$ just one hyperplane suffices[3]. From this one

[3] An algorithm to find such a hyperplane that runs in polynomial time (in dimension d) is not known.

can infer that $\lceil m/d \rceil$ hyperplanes always suffice, however we are interested in approximating the optimum number of hyperplanes for a given instance. Let $k_i = |P_i|$ and let $P = \cup_i P_i$. We will assume, for notational simplicity, that the sets P_i are disjoint. The assumption can be dispensed with.

In [30] the authors reduce their problem to Mult-Submod-Cover as follows. Let N be the set of all hyperplanes in $\mathbb{R}^d$; we can confine attention to a finite subset by restricting to those half-spaces that are supported by d points of P. For each point set P_i they consider a complete graph G_i on the vertex set P_i. For each $p \in \cup_i P_i$ they define a submodular function $f_p : 2^N \to \mathbb{R}_+$ where $f_p(S)$ is the number of edges incident to p that are cut by S; an edge (p, q) with $p, q \in P_i$ is cut if p and q are separated by at least one of the hyperplanes in S. Thus one can formulate the original problem as choosing the smallest number of hyperplanes such that for each $p \in P$ the number of edges that are cut is at least k_p where k_p is the demand of p. To ensure that P_i is partitioned such that no cell has more than $k_i/2$ points we set $k_p = k_i/2$ for each $p \in P_i$; more generally if we wish no cell to have more than βk_i points of P_i we set $k_p = (1 - \beta)k_i$ for each $p \in P_i$. As a special case of Mult-Submod-Cover we have

$$\begin{aligned} \min_{S \subseteq N} |S| \quad & \text{s.t} \\ f_p(S) \geq k_p \quad & p \in P \end{aligned}$$

Using Wolsey's result for Submod-SC, [30] obtain an $O(\log(mn))$ approximation where $n = \sum_i k_i$.

We now show that one can obtain an $O(\log m)$-approximation if we settle for a bicriteria approximation where we compare the cost of the solution to that of an optimum solution, but guarantee a slightly weaker bound on the partition quality. This could be useful since one can imagine several applications where m, the number of different point sets, is much smaller than the total number of points. Consider the formulation from [30]. Suppose we used our bicriteria approximation algorithm for Mult-Submod-Cover. The algorithm would cut $(1 - 1/e - \varepsilon)k_p$ edges for each p and hence for $1 \leq i \leq m$ we will only be guaranteed that each cell in the arrangement contains at most $(1 - (1 - 1/e - \varepsilon)/2)k_i$ points from P_i. This is acceptable in many applications. However, the approximation ratio still depends on n since the number of constraints in the formulation is n. We describe a related but slightly modified formulation to obtain an $O(\log m)$-approximation by using only m constraints.

Given a collection $S \subseteq N$ let $f_i(S)$ denote the number of pairs of points in P_i that are separated by S (equivalently the number of edges of G_i cut by S). It is easy to see that $f_i(S)$ is a monotone submodular function over N. Suppose $S \subseteq N$ induces an arrangement such that no cell in the arrangement contains more than $(1 - \beta)k_i$ points for some $0 < \beta < 1$. Then S cuts at least $\beta k_i(k_i - 1)/2$ edges from G_i; in particular if $\beta = 1/2$ then S cuts at least $k_i(k_i - 1)/4$ edges. Conversely if S cuts at least $\alpha k_i(k_i - 1)$ edges for some $\alpha < 1/2$ then no cell in the arrangement induced by S has more than $(1 - \Omega(\alpha))k_i$ points from P_i. Given this we can consider the formulation below.

$$\min_{S \subseteq N} |S| \quad \text{s.t}$$
$$f_i(S) \geq k_i(k_i - 1)/4 \quad 1 \leq i \leq m$$

We apply our bicriteria approximation for Mult-Submod-Cover with some fixed ε to obtain an $O(\log m)$-approximation to the objective but we are only guaranteed that the output S satisfies the property that $f_i(S) \geq (1 - 1/e - \varepsilon)k_i(k_i - 1)/4$ for each i. This is sufficient to ensure that no P_i has more than a constant factor in each cell of the arrangement.

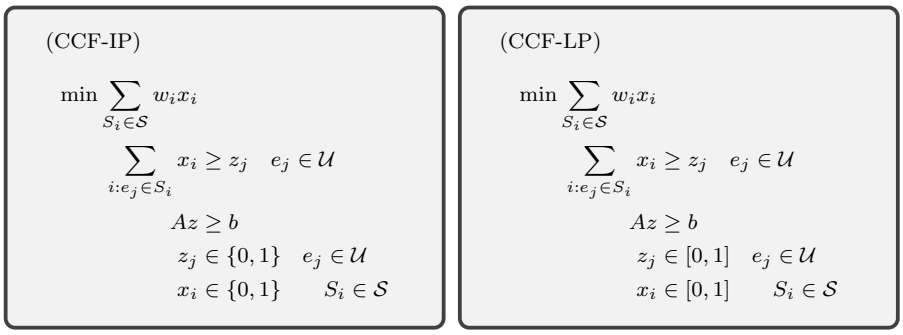
The running time of the algorithm we describe depends polynomially on N and m and N can be upper bounded by n^d. The running time in [30] is $O(mn^{d+2})$. Finding a running time that depends polynomially on n, m and d is an interesting open problem.

4 Approximating Multi-Partial-SC

In this section we consider a problem that generalizes Multi-Partial-SC and CIPs while being a special case of Mult-Submod-Cover. We call this problem CCF (Covering Coverage Functions). Bera et al. [6] already considered this version in the restricted context of Vertex-Cover. Formally the input is a weighted set system $(\mathcal{U}, \mathcal{S})$ and a set of inequalities of the form $Az \geq b$ where $A \in [0,1]^{h \times n}$ matrix and $b \in \mathbb{R}^h_+$ is a positive vector. The goal is to optimize the integer program CCF-IP shown in Fig 3a. Multi-Partial-SC is a special case of CCF when the matrix A contains only $\{0, 1\}$ entries. On the other hand CIP is a special case when the set system is very restricted and each set S_i consists of a single element. We say that an instance is r-sparse if each set S_i "influences" at most r rows of A; in other words the elements of S_i have non-zero coefficients in at most r rows of A. This notion of sparsity coincides in the case of CIPs with column sparsity and in the case of Mult-Submod-Cover with the sparsity that we saw in Section 3. It is useful to explicitly see why CCF is a special case of Mult-Submod-Cover. The ground set $N = [m]$ corresponds to the sets $S_1, \ldots, S_m$ in the given set system $(\mathcal{U}, \mathcal{S})$. Consider the row k of the covering constraint matrix $Az \geq b$. We can model it as a constraint $f_k(S) \geq b_k$ where the submodular set function $f_k : 2^N \to \mathbb{R}_+$ is defined as follows: for a set $X \subseteq N$ we let $f_k(X) = \sum_{e_j \in \cup_{i \in X} S_i} A_{k,j}$ which is simply a weighted coverage function with the weights coming from the coefficients of the matrix A. Note that when formulating via these submodular functions, the auxiliary variables $z_1, \ldots, z_n$ that correspond to the elements $\mathcal{U}$ are unnecessary.

We prove the following theorem.

Theorem 4.1. *Consider an instance of r-sparse* CCF *induced by a set system $(\mathcal{U}, \mathcal{S})$ from a deletion-closed family with a β-apprximation for* Set Cover *via the natural LP. There is a randomized polynomial-time algorithm that outputs a feasible solution of expected cost $O(\beta + \ln r)\mathrm{OPT}$.*



(a) IP Formulation for CCF (b) LP relaxation of CCF-IP

Fig. 3: Modeling CCF

The natural LP relaxation for CCF is show in Fig 3b. It is well-known that this LP relaxation, even for CIPs with only one constraint, has an unbounded integrality gap [13]. For CIPs knapsack-cover inequalities are used to strengthen the LP. KC-inequalities in this context were first introduced in the influential work of Carr et al. [13] and have since become a standard tool in developing stronger LP relaxations. Bera et al. [6] adapt KC-inequalities to the setting of PartitionVC, and it is straightforward to extend this to CCF (this is implicit in [6]).

Remark 4.2. *Weighted coverage functions are a special case of sums of weighted rank functions of matroids. The natural LP for* CCF *can be viewed as using a different, and in fact a tighter extension, than the multilinear relaxation [11]. The fact that one can use an LP relaxation here is crucial to the scaling idea that will play a role in the eventual algorithm. The main difficulty, however, is the large integrality gap which arises due to the partial covering constraints.*

We set up and the explain the notation to describe the use of KC-inequalities for CCF. It is convenient here to use the reduction of CCF to Mult-Submod-Cover. For row k in $Ax \geq b$ we will use f_k to denote the submodular function that we set up earlier. Recall that $f_k(D)$ captures the coverage to constraint k if set D is chosen. The residual requirement after choosing D is $b_k - f_k(D)$. The residual requirement must be covered by elements from sets outside D. The maximum contribution that $i \notin D$ can provide to this is $\min\{f_k(D+i) - f_k(D), b_k - f_k(D)\}$. Hence the following constraint is valid for any $D \subset N$:

$$\sum_{i \notin D} \min\{f_k(D+i) - f_k(D), b_k - f_k(D)\} x_i \geq b_k - f_k(D) \tag{1}$$

Writing the preceding inequality for every possible choice of D and for every k we obtained a strengthened LP that we show in Fig 4.

(CCF-KC-LP)

$$\min \sum_{S_i \in \mathcal{S}} w_i x_i$$

$$\sum_{i: e_j \in S_i} x_i \geq z_j \quad e_j \in \mathcal{U}$$

$$Az \geq b$$

$$\sum_{i \notin D} \min\{f_k(D+i) - f_k(D), b_k - f_k(D)\} x_i \geq b_k - f_k(D) \quad D \subset [m], 1 \leq k \leq h$$

$$z_j \in [0,1] \quad e_j \in \mathcal{U}$$

$$x_i \in [0,1] \quad S_i \in \mathcal{S}$$

Fig. 4: CCF-LP with KC-Inequalities

CCF-KC-LP has an exponential number of constraints and the separation problem involves submodular functions. A priori it is not clear that there is even an approximate separation oracle. However, one can combine rounding and separation, as shown in [6]. It is instructive to first see the rounding procedure assuming that the LP can be solved exactly.

Rounding and analysis assuming LP can be solved exactly: Let (x, z) be an optimum solution to CCF-KC-LP. We can assume without loss of generality that for each element $e_j \in \mathcal{U}$ we have $z_j = \min\{1, \sum_{i: e_j \in S_i} x_i\}$. As in Section 6 we split the elements in $\mathcal{U}$ into heavily covered elements and shallow elements. For some fixed threshold τ that we will specify later, let $\mathcal{U}_{\text{he}} = \{e_j \mid z_j \geq \tau\}$, and $\mathcal{U}_{\text{sh}} = \mathcal{U} \setminus \mathcal{U}_{\text{he}}$. We will also choose another threshold. The rounding algorithm is the following.

1. Solve a Set Cover problem via the natural LP to cover all elements in $\mathcal{U}_{\text{he}}$. Let Y_1 be the sets chosen in this step.
2. Let $Y_2 = \{S_i \mid x_i \geq \tau\}$ be the heavy sets.
3. Repeat for $\ell = \Theta(\ln r)$ rounds: independently pick each set S_i in $\mathcal{S} \setminus (Y_1 \cup Y_2)$ with probability $\frac{1}{\tau} x_i$. Let Y_3 be the sets chosen in this randomized rounding step.
4. For $k \in [h]$ do
   (a) Let $b_k - f_k(Y_1 \cup Y_2 \cup Y_3)$ be the residual requirement of k'th constraint.
   (b) Run the *modified* Greedy algorithm to satisfy the residual requirement. Let F_k be the sets chosen to fix the constraint (could be empty).
5. Output $Y_1 \cup Y_3 \cup (\cup_{k=1}^{h} F_k)$.

The algorithm at a high level is similar to that in [6]. There are two main differences. First, we explicitly fix the constraints after the randomized rounding phase using a slight variant of the Greedy algorithm. This ensures that the output of the algorithm is always a feasible solution; this makes it easier to analyze the r-sparse case while a straight forward union bound will

not work. Second, the analysis relies on a probabilistic inequality that is simpler in Vertex-Cover case while it requires a more sophisticated approach here. We now describe the modified Greedy algorithm to fix a constraint. For an unsatisfied constraint k we consider the collection of sets that influence the residual requirement for k, and partition them it into H_k and L_k. H_k is the collection of all sets such that choosing any of them completely satisfies the residual requirement for k, and L_k are the remaining sets. The modified Greedy algorithm for fixing constraint k picks the better of two solutions: (i) the first solution is the cheapest set in H_k (this makes sense only if $H_k \neq \emptyset$) and (ii) the second solution is obtained by running Greedy on sets in L_k until the constraint is satisfied.

Analysis: We now analyze the expected cost of the solution output by the algorithm. The lemma below bounds the cost of Y_1.

Lemma 4.3. *The cost of Y_1, $w(Y_1)$ is at most $\beta \frac{1}{\tau} \sum_i w_i x_i$.*

Proof. Recall that $z^*_j \geq \tau$ for each $e_j \in \mathcal{U}_{\text{he}}$. Consider $x'_i = \min\{1, \frac{1}{\tau} x_i\}$. It is easy to see that x' is a feasible fractional solution for SC-LP to cover $\mathcal{U}_{\text{he}}$ using sets in $\mathcal{S}$. Since the set family is deletion-closed, and the integrality gap of the SC-LP is at most β for all instances in the family, there is an integral solution covering $\mathcal{U}_{he}$ of cost at most $\beta \sum_i w_i x'_i \leq \frac{1}{\tau} \beta \sum_i w_i x_i$. □

The expected cost of randomized rounding in the second step is easy to bound.

Lemma 4.4. *The expected cost of Y_3 is at most $\frac{\ell}{\tau} \sum_i w_i x_i$.*

An analog of the following lemma for PartitionVC was proved by [6]. However, in PartitonVC, each element is contained in at most two sets, which is crucially used in their proof. Consequently, their proof does not readily generalize to set systems with unbounded frequency. We rely on tools from submodularity to prove this lemma even in the general case of CCF.

Lemma 4.5. *Fix a constraint k. If τ is a sufficiently small but fixed constant, the probability that constraint k is satisfied after one round of randomized rounding is at least a fixed constant c_τ.*

Before we give a proof of this lemma, we finish the rest of the analysis first. Let $I_k = \{i \mid S_i \text{ influences constraint } k\}$. Note that for any $i \in N$, $|\{k \in [h] : S_i \in I_k\}| \leq r$ by our sparsity assumption.

Lemma 4.6. *Let ρ_k be the cost of fixing constraint k if it is not satisfied after randomized rounding. Then $\rho_k \leq c'_\tau \sum_{i \in I_k} w_i x_i$ for some constant c'_τ.*

Proof. We will assume that $\tau < (1 - 1/e)/2$. Let $D = Y_1 \cup Y_2$ and let $b'_k = b_k - f_k(D)$ be residual requirement of constraint k after choosing Y_1 and Y_2. Let $\mathcal{U}' = \mathcal{U} \setminus \mathcal{U}_D$ be elements in the residual instance; all these are shallow elements. Consider the scaled solution x' where $x'_i = 1$ if $S_i \in D$ and $x'_i = \frac{1}{\tau} x_i$

for other sets. For any shallow element e_j let $z'_j = \min\{1, \sum_{i:j\in S_i} x'_i\}$; since e_j is shallow we have $z'_j = \frac{1}{\tau} z_j = \sum_{i:j\in S_i, i\notin D} x'_i$.

Recall from the description of the modified Greedy algorithm that a set S_i is in $H_k \subseteq I_k$ iff adding S_i to D satisfies constraint k. In other words $i \in H_k$ iff $f_k(D+i) - f_k(D) \geq b'_k$. Suppose $\sum_{i\in H_k} x'_i \geq 1/2$. Then it is not hard to see that the cheapest set from H_k will cover the residual requirement and has cost at most $2\sum_{i\in H_k} w_i x'_i$ and we are done. We now consider the case when $\sum_{i\in H_k} x'_i < 1/2$. Let $L_k = I_k \setminus H_k$. For each $j \in \mathcal{U}'$ let $z''_j = \sum_{i:j\in S_i, i\in L_k} x'_i$. We claim that $\sum_{j\in\mathcal{U}'} A_{k,j} z''_j \geq \frac{1}{2\tau} b'_k$. Since $\tau \leq (1-1/e)/2$ this implies $\sum_{j\in\mathcal{U}'} A_{k,j} z''_j \geq \frac{1}{(1-1/e)} b'_k$. Assuming the claim, if we run Greedy on L_k to cover at least b'_k elements then the total cost, by Lemma 2.1, is at most $(1+e)\sum_{i\in L_k} w_i x'_i$; note that we use the fact that no set in L_k has coverage more than b'_k and hence $c = 1$ in applying Lemma 2.1.

We now prove the claim. Since the x, z satisfy KC inequalities:

$$\sum_{i\notin D, i\in I_k} \min\{f_k(D+i) - f_k(D), b'_k\} x_i \geq b'_k.$$

We split the LHS into two terms based on sets in H_k and L_k. Note that if $i \in H_k$ then $f_k(D+i) - f_k(D) \geq b'_k$ and if $i \in L_k$ then $f_k(D+i) - f_k(D) < b'_k$. Furthermore, $f_k(D+i) - f_k(D) \leq \sum_{e_j\in S_i} A_{k,j}$. We thus have

$$\begin{aligned}\sum_{i\notin D, i\in I_k} \min\{f_k(D+i) - f_k(D), b'_k\} x_i &\leq \sum_{i\in H_k} b'_k x_i + \sum_{i\in L_k} x_i \sum_{e_j\in S_i} A_{k,j} \\ &\leq b'_k \sum_{i\in H_k} x_i + \sum_{i\in L_k} x_i \sum_{e_j\in S_i} A_{k,j}\end{aligned}$$

Putting together the preceding two inequalities and the condition that $\sum_{i\in H_k} x'_i < 1/2$ (recall that $x'_i = x_i/\tau$ for each $i \in I_k$),

$$\sum_{i\in L_k} x'_i \sum_{e_j\in S_i} A_{k,j} \geq \frac{1}{2\tau} b'_k.$$

We have, by swapping the order of summation,

$$\sum_{i\in L_k} x'_i \sum_{e_j\in S_i} A_{k,j} = \sum_{e_j\in\cup_{i\in L_k} S_i} A_{k,j} \sum_{i\in L_k: e_j\in S_i} x'_i \leq \sum_{j\in\mathcal{U}'} A_{k,j} \sum_{i\in L_k: e_j\in S_i} x'_i = \sum_{j\in\mathcal{U}'} A_{k,j} z''_j.$$

The preceding two inequalities prove the claim. □

With the preceding lemmas we can finish the analysis of the total expected cost of the sets output by the algorithm. From Lemma 4.5 the probability that any fixed constraint k is not satisfied after the randomized rounding step is at most c^ℓ, for some constant $c < 1$. By choosing $\ell \geq 1 + \log_{1/c} r$ we can reduce this probability to at most $1/r$. Thus, as in the preceding section, the expected fixing cost is $\sum_k \frac{1}{r} w(F_k)$. From Lemma 4.6,

$$\sum_k w(F_k) \leq c' \sum_k \sum_{i\in I_k} w_i x_i \leq c' r \sum_i w_i x_i$$

since the given instance is r-sparse. Thus the expected fixing cost is at most $c' \sum_i w_i x_i$. The cost of Y_1 is $O(\beta) \sum_i w_i x_i$, the cost of Y_2 is $O(1) \sum_i w_i x_i$, and the expected cost of Y_3 is $O(\log r) \sum_i w_i x_i$. Putting together, the total expected cost is at most $O(\beta + \log r) \sum_i w_i x_i$ where the constants depend on τ. We need to choose τ to be sufficiently small to ensure that Lemma 4.5 holds. We do not attempt to optimize the constants or specify them here.

Submodularity and proof of Lemma 4.5: We follow some notation that we used in the proof of Lemma 4.6. Let $D = Y_1 \cup Y_2$ and consider the residual instance obtained by removing the elements covered by D and reducing the coverage requirement of each constraint. The lemma is essentially only about the residual instance. Fix a constraint k and recall that b'_k is the residual coverage requirement and that each set in H_k fully satisfies the requirement by itself. Recall that $x'_i = \frac{1}{\tau} x_i \leq 1$ for each set $i \notin D$ and $z'_j = \frac{1}{\tau} z_j = \sum_{i:e_j \in S_i} x'_i$ for each residual element e_j. As in the proof of Lemma 4.6 we consider two cases. If $\sum_{i \in H_k} x'_i \geq 1/2$ then with probability $(1 - 1/\sqrt{e})$ at least one set from H_k is picked and will satisfy the requirement by itself. Thus the interesting case is when $\sum_{i \in H_k} x'_i < 1/2$. Let $\mathcal{U}'' = \cup_{i \in L_k} S_i$. As we saw earlier, in this case

$$\sum_{j \in \mathcal{U}''} A_{k,j} \min\{1, \sum_{i: j \in S_i} x'_i\} \geq \frac{1}{2\tau} b'_k.$$

For ease of notation we let $N = L_k$ be a ground set. Consider the weighted coverage function $g : 2^N \to \mathbb{R}_+$ where $g(T)$ for $T \subseteq L_k$ is given by $\sum_{j \in \cup_{i \in T} S_i} A_{k,j}$. Then for a vector $y \in [0,1]^N$ the quantity $\sum_{j \in \mathcal{U}''} A_{k,j} \min\{1, \sum_{i: j \in S_i} y_i\}$ is the continuous extension $\tilde{g}(y)$ discussed in Section 2. Thus we have $\tilde{g}(x') \geq \frac{1}{2\tau} b'_k$. From Lemma 2.2, we have $G(x') \geq (1 - 1/e) \frac{1}{2\tau} b'_k$ where G is the multilinear extension of g. If we choose $\tau \leq (1 - 1/e)/4$ then $G(x') \geq 2b'_k$. Let Z be the random variable denoting the value of $g(R)$ where $R \simeq x'$. Independent random rounding of x' preserves $G(x')$ in expectation by the definition of the multilinear extension, therefore $\mathrm{E}[Z] = G(x') \geq 2b'_k$. Moreover, by Lemma 2.4, Z is concentrated around its expectation since $G(i) \leq b'_k$ for each $i \in L_k$. An easy calculation shows that $\Pr[Z < b'_k] \leq e^{1/4} < 0.78$. Thus with constant probability $g(R) \geq b'_k$.

Solving the LP with KC inequalities: As noted in [6], one can combine the rounding procedure with the Ellipsoid method to obtain the desired guarantees even though we do not obtain a fractional solution that satisfies all the KC inequalities. This observation holds for our rounding as well. We briefly sketch the argument.

The proof of the performance guarantee of the algorithm relies on the fractional solution satisfying KC inequalities with respect to the set $D = Y_1 \cup Y_2$. Thus, given a fractional solution (x, z) for the LP we can check the easy constraints in polynomial time and implement the first two steps of the algorithm. Once Y_1, Y_2 are determined we have D and one can check if (x, z) satisfies KC inequalities with respect to D (for each row of A). If it does

then the rest of the proof goes through and performance guarantee holds with respect to the cost of (x, z) which is a lower bound on OPT. If some constraint does not satisfy the KC inequality with respect to D we can use this as a separation oracle in the Ellipsoid method.

5 Facility Location and Minimum Sum of Radii Clustering

In this section, we consider two well-studied problems related to clustering in the setting where there are r partial covering constraints. As we mentioned previously, the generalization of Facility Location problem does not quite fit in the Mult-Submod-Cover framework. However, we are able to adapt the techniques to this problem. We obtain $O(\log r)$-approximations for these two problems and they are treated in the next two sections.

5.1 Facility Location with Multiple Outliers

Here, we consider a generalization of the Facility Location Problem that is analogous to Multi-Partial-SC. We show how to adapt the randomized rounding framework, along with existing LP-based approximation algorithms for the standard Facility Location problem, to obtain an $O(\log r)$ approximation for this generalization.

In FL-Multi-Ouliers, we are given a set of facilities F, a set of clients C, belonging to a metric space $(F \cup C, d)$. Each facility $i \in F$ has a non-negative opening cost f_i. We are given r non-empty subsets of clients $C_1, \ldots, C_r$, that partition the set C of clients. Each color class C_k has a connection requirement $1 \leq b_k \leq |C_k|$. The objective of FL-Multi-Ouliers is to find a solution (F^*, C^*) that minimizes $\sum_{i \in F'} f_i + \sum_{j \in C'} d(j, F')$ over all feasible solutions (F', C'). We say that a solution (F', C') is feasible if (i) $|F'| \geq 1$ and (ii) For all classes C_k, $|C_k \cap C'| \geq b_k$. Note that the special case with just one class is the Robust Facility Location problem, first considered by Charikar et al. [16].

A natural LP formulation of this problem is as follows.

$$\text{minimize} \sum_{i \in F} f_i x_i + \sum_{i \in F, j \in C} y_{ij} \cdot d(i,j)$$

$$\text{subject to} \sum_{i \in F} y_{ij} \geq z_j, \qquad \forall j \in C \qquad (2)$$

$$\sum_{j \in C_k} z_j \geq b_k, \qquad \forall 1 \leq k \leq r \qquad (3)$$

$$0 \leq y_{ij} \leq x_i \leq 1, \qquad \forall i \in F, \forall j \in C \qquad (4)$$

$$z_j \in [0,1], \qquad \forall j \in C \qquad (5)$$

Next, we discuss strengthening of this LP by adding KC inequalities and solving the strengthened LP.

Solving a Strengthened LP: First, we convert the LP into a feasibility LP by guessing the optimal cost up to a factor of 2, say Δ, and by adding a *cost constraint* $\sum_{i \in F} f_i x_i + \sum_{i \in F, j \in C} y_{ij} \cdot d(i,j) \leq \Delta$. Similar to Section 4, we use the Ellipsoid algorithm to find a feasible LP solution that satisfies the cost constraint, constraints 2 to 5, and an additional KC inequality specified below. Fix an LP solution that satisfies all these constraints (except possibly the additional one). With respect to this solution, let $H = \{j \in C \mid \sum_{i \in F} y_{ij} \geq \tau\}$ be the set of *heavy* clients, where τ is a constant as in Section 4.

Let $L = C \setminus H$ be the set of *light* clients. Note that for any light client $j \in L$, $z_j \leq \sum_{i \in F} y_{ij} < \tau$. Also, for a class C_k, let $C_k(H) := C_k \setminus H$ denote the light clients from C_k, and let $b'_k := b_k - |C_k \cap H|$ denote its residual connection requirement. Now, we check whether the following constraint holds for all color classes C_k:

$$\sum_{i \in F} \min \left\{ x_i \cdot b'_k, \sum_{j \in C_k(H)} y_{ij} \right\} \geq b'_k \tag{6}$$

First, note that this can be formulated as an LP constraint by introducing auxiliary variables. It is easy to see that any integral solution satisfies this constraint for any $H \subseteq C$, and hence it is valid. If this constraint is not satisfied for some class C_k, we report it as a violated constraint to the Ellipsoid algorithm.

Rounding: Now, suppose we have an LP solution (x, y, z) that satisfies 2 to 5, and has cost at most Δ. Let H and L be the corresponding heavy and light clients. Furthermore, suppose the LP solution satisfies Constraint 6 with respect to H and L.

For any $i \in F$, let $x'_i := \min\{1, \frac{1}{\tau} x_i\}$, and for any $i \in F, j \in H$, let $y'_{ij} := \min\{1, \frac{1}{\tau} y_{ij}\}$. It is easy to see that (x', y') is a feasible Facility Location (without outliers) LP solution for the instance induced by the heavy clients, and its cost is at most Δ/τ. We use an LP-based algorithm (such as [10]) with a constant approximation guarantee to round this solution to an integral solution (F_H, H), where $F_H \subseteq F$.

For handling light clients, we "split" the facilities into multiple co-located copies if necessary, we ensure the following two conditions hold:

1. For any facility $i \in F$, $x_i < \tau$.
2. For any client $j \in L$ and any facility $i \in F$, $y_{ij} > 0 \implies y_{ij} = x_i$.

This has to be done in a careful manner – we give the details in appendix A. This procedure results in a feasible LP solution of essentially the same cost. Henceforth, we treat all co-located copies of a facility as distinct facilities for the sake of the analysis. We now show that the rounding for the light clients

can be reduced to the randomized rounding algorithm for Multi-Partial-SC from the previous section.

For any facility $i \in F$, let $S_i := \{j \in L \mid x_i = y_{ij}\}$ denote the set of light clients that are fractionally connected to i. The cost of opening facility i and connecting all $j \in S_i$ to i is equal to $w_i := f_i + \sum_{j \in S_i} d(i,j)$. Consider a Multi-Partial-SC instance $(\mathcal{U}, \mathcal{S})$, where $\mathcal{S} = \{S_i \mid i \in F\}$ with weights w_i, and residual coverage requirement b'_k for each class $C_k(H)$. We obtain an LP solution (x, z) for this instance of Multi-Partial-SC from the LP solution (x, y, z) for the Facility Location problem, by taking the variables x_i for $i \in F$ and z_j for $j \in L$. The following properties are satisfied by this LP solution (x, z).

1. All the elements are light, and all the sets $S_i \in \mathcal{S}$ have $x_i < \tau$.
2. The costs of the two LP solutions are equal:

$$\sum_{S_i \in \mathcal{S}} w_i x_i = \sum_{i \in F} x_i \cdot \left(f_i + \sum_{j \in S_i} d(i,j) \right) = \sum_{i \in F} f_i x_i + \sum_{i \in F,\ j \in L} y_{ij} \cdot d(i,j).$$

3. Constraint 6 is equivalent to:

$$\sum_{S_i \in \mathcal{S}} x_i \cdot \min\{b'_k, |S_i \cap C_k(H)|\} \geq b'_k \quad \forall 1 \leq k \leq r.$$

This is exactly the KC inequality (1) required for rounding Multi-Partial-SC.

Therefore, we can use the randomized rounding plus alteration algorithm from Section 4 to obtain a solution Y' for the Multi-Partial-SC. It has cost at most $O(\log r) \cdot \Delta$, and for each class $C_k(H)$, it covers at least b'_k clients. To obtain a solution for the original facility location problem, we open a facility $i \in F$, if its corresponding set S_i is selected in Y', and connect to it all the clients in S_i. Notice that we connect b'_k clients from $C_k(H)$ to the set of opened facilities in this manner. The cost of this solution is upper bounded by $w(Y') \leq O(\log r) \cdot \Delta$. Combining this with the solution (F_H, H) for the heavy clients of cost at most $O(1) \cdot \Delta$, we obtain our overall solution for the given instance. It is easy to see that this is an $O(\log r)$ approximation.

Theorem 5.1. *There is a randomized polynomial-time algorithm that outputs a feasible solution of expected cost* $O(\log r) \cdot$ OPT *for* FL-Multi-Ouliers.

5.2 Minimum Sum of Radii Clustering with Multiple Outliers

Here, we are given a set of facilities F, a set of clients C and a metric space $(F \cup C, d)$. Each facility $i \in F$ has a non-negative opening cost f_i. We are given r classes of clients $C_1, \ldots, C_r$. Each color class C_k has a coverage requirement b_k where $1 \leq b_k \leq |C_k|$. A ball centered at a facility $i \in F$ of radius $\rho \geq 0$ is the set $B(i, \rho) := \{j \in C \mid d(i,j) \leq \rho\}$. The goal is to select a set of

balls $\mathcal{B} = \{B_i = B(i, \rho_i) \mid i \in F' \subseteq F\}$ centered at some subset of facilities $F' \subseteq F$, such that (i) the set of balls $\mathcal{B}$ satisfies the coverage requirement of each color class and (ii) the sum $\sum_{i \in F'}(f_i + \rho_i^\gamma)$ is minimized. Here, $\gamma \geq 1$ is a constant, and is a parameter of the problem. We refer to this problem as MCC-Multi-Outliers.

Note that even though the radius of a ball centered at $i \in F$ is allowed to be any non-negative real number, it can be restricted to the following set of "relevant" radii: $R_i := \{d(i, j) \mid j \in C\}$. Now, define a set system $(C, \mathcal{S})$. Here, C is the set of clients, and $\mathcal{S} = \{B(i, \rho) \mid i \in F, \rho \in R_i\}$, with weight of the set corresponding to a ball $B(i, \rho)$ being defined as $f_i + \rho^\gamma$. Now, we use the algorithm from Section 4 for this set system. Let H be the set of heavy clients (or elements) as defined in Section 4. We use the Primal-Dual algorithm of Charikar and Panigrahy [17] [4] with an approximation guarantee of $\beta = 3^\gamma$ (which is a constant) to obtain a cover for the heavy clients. For the remaining light clients, we use the randomized rounding algorithm as is. Note that this reduction is not exact, since the solution thus obtained may select sets corresponding to multiple concentric balls in the original instance. However, from each set of concentric balls, we can choose the ball with the largest radius. This pruning process does not affect the coverage, and can only decrease the cost of the solution. Therefore, it is easy to see that the resulting solution is an $O(\log r)$ approximation.

Theorem 5.2. *There is a randomized polynomial-time algorithm that outputs a feasible solution of expected cost* $O(3^\gamma + \log r) \cdot$ OPT *for* MCC-Multi-Outliers.

6 Approximating Partial-SC

In this section we consider the algorithm for Partial-SC from [33] and suggest a small variation that simplifies the algorithm and analysis and in the process also yields an improved approximation ratio. The approach of [33] is as follows. Given an instance of Partial-SC with a set system $(\mathcal{U}, \mathcal{S})$ their algorithm has the following high level steps.

1. Guess the largest weight set in an optimum solution. Remove all elements covered by it, remove all sets with weight larger than the guessed set. Adjust k to account for covered elements. We now work with the residual instance of Partial-SC.
2. Solve PSC-LP. Let (x^*, z^*) be an optimum solution. For some threshold τ let $\mathcal{U}_h = \{e_j \mid z_j^* \geq \tau\}$ be the highly covered elements and let $\mathcal{U}_\ell = \{e_j \mid z_j^* < \tau\}$ be the shallow elements.
3. Solve a Set Cover instance via the LP to cover all elements in $\mathcal{U}_h$. The cost of this solution is at most $\frac{1}{\tau}\beta \sum_i w_i x_i^*$ since one can argue that the

[4] Charikar and Panigrahy [17] consider the special case of $\gamma = 1$. However, their algorithm easily generalizes to arbitrary γ.

fractional solution x' where $x'_i = \min\{1, x^*_i/\tau\}$ for each i is a feasible fractional solution for SC-LP to cover $\mathcal{U}_h$.

4. Let $k' = k - |\mathcal{U}_h|$ be the residual number of elements that need to be covered from $\mathcal{U}_\ell$. Round (x^*, z^*) to cover k' elements from $\mathcal{U}_\ell$.

The last step of the algorithm is the main technical one, and also determines τ. In [33] τ is chosen to be $1/2$ and this leads to their $2(\beta+1)$-approximation. The rounding algorithm in [33] can be seen as an adaptation of pipage rounding [1] for Max-Budgeted-Cover. The details are somewhat technical and perhaps obscure the high-level intuition that scaling up the LP solution allows one to use a bicriteria approximation for Max-Budgeted-Cover. Our contribution is to simplify the fourth step in the preceding algorithm. Here is the last step in our algorithm; the other steps are the same modulo the specific choice of τ.

4'. Run Greedy to cover k' elements from $\mathcal{U}_\ell$.

We now analyze the performance of our modified algorithm.

Lemma 6.1. *Suppose $\tau \leq (1 - 1/e)$. Then running Greedy in the final step outputs a solution of total weight at most $\max_i w_i + \frac{1}{\tau}\sum_i w_i x^*_i$ to cover $k' = k - |\mathcal{U}_h|$ elements from $\mathcal{U}_\ell$.*

Proof. It is easy to see that $\sum_{e_j \in \mathcal{U}_\ell} z^*_j \geq k'$ since $\sum_{e_j \in \mathcal{U}} z^*_j \geq k$ and $z^*_j \leq 1$ for each e_j. Let $(\mathcal{U}_\ell, \mathcal{S}')$ be the set system obtained by restricting $(\mathcal{U}, \mathcal{S})$ to $\mathcal{U}_\ell$, and let (x', z') be the restriction of (x^*, z^*) to the set system $(\mathcal{U}_\ell, \mathcal{S}')$. We have (i) $\sum_i w_i x'_i \leq \sum_i w_i x^*_i$ and (ii) $\sum_{e_j \in \mathcal{U}_\ell} z'_j \geq k'$ and (iii) $z'_j \leq \tau \leq (1 - 1/e)$ for all $e_j \in \mathcal{U}_\ell$.

Consider (x'', z'') obtained from (x', z') as follows. For each $e_j \in \mathcal{U}_\ell$ set $z''_j = \frac{1}{\tau} z'_j$ and note that $z''_j \leq 1$. For each set S_i set $x''_i = \min\{1, \frac{1}{\tau} x'_i\}$. It is easy to see that (x'', z'') is a feasible solution to PSC-LP. Note that $Z = \sum_{e_j \in \mathcal{U}_\ell} z''_j \geq \frac{1}{\tau} k'$. Let $B = \sum_i w_i x''_i \leq \frac{1}{\tau}\sum_i w_i x^*_i$. The fractional solution (x'', z'') is also a feasible solution to the LP formulation MBC-LP. We apply Lemma 2.1 to this fractional solution. Suppose we stop Greedy when it covers k' elements or when it first crosses the budget B, whichever comes first. Clearly the total weight is at most $B + \max_i w_i$. We argue that at least k' elements are covered when we stop Greedy. The only case to argue is when Greedy is stopped when the weight of sets picked by it exceeds B for the first time. From Lemma 2.1 it follows that Greedy covers at least $(1 - 1/e)Z$ elements but since $Z \geq \frac{1}{\tau} k'$ it implies that Greedy covers at least k' elements when it is stopped. □

We formally state a lemma to bound the cost of covering $\mathcal{U}_h$. The proof of this lemma is identical to that of Lemma 4.3, and therefore omitted.

Lemma 6.2. *The cost of covering $\mathcal{U}_h$ is at most $\beta \frac{1}{\tau}\sum_i w_i x^*_i$.*

Finally, we can analyze the approximation guarantee of the overall solution.

Theorem 6.3. *Setting $\tau = (1-1/e)$, the algorithm outputs a feasible solution of total cost at most $\frac{e}{e-1}(\beta+1)\mathrm{OPT}$ where OPT is the value of an optimum integral solution.*

Proof. Fix an optimum solution. Let W be the weight of a maximum weight set in the optimum solution. In the first step of the algorithm we can assume that the algorithm has correctly guessed a maximum weight set from the fixed optimum solution. Let $\mathrm{OPT}' = \mathrm{OPT} - W$. In the residual instance the weight of every set is at most W. The optimum solution value for PSC-LP, after guessing the largest weight set and removing it, is at most OPT'. From Lemma 6.2, the cost of covering $\mathcal{U}_h$ is at most $\frac{e}{e-1}\beta\mathrm{OPT}'$. From Lemma 6.1, the cost of covering k' elements from $\mathcal{U}_\ell$ is most $\frac{e}{e-1}\mathrm{OPT}' + W$. Hence the total cost, including the weight of the guessed set, is at most

$$\begin{aligned} W + \frac{e}{e-1}\beta\mathrm{OPT}' + \frac{e}{e-1}\mathrm{OPT}' + W &= \frac{e}{e-1}(\beta+1)\mathrm{OPT} + W(2 - \frac{e}{e-1}(\beta+1)) \\ &\leq \frac{e}{e-1}(\beta+1)\mathrm{OPT} \end{aligned}$$

since $\beta \geq 1$. □

7 Concluding Remarks

The paper shows the utility of viewing Partial-SC and its generalizations as special cases of Mult-Submod-Cover. The coverage function in set systems is a submodular funtion that belongs to the class of sum of weighted matroid rank functions. Certain ideas for the coverage function extend to this larger class. Are there interesting problems that can be understood through this view point? Are there other special classes of submodular functions for which one can obtain uni-criteria approximation algorithms for Mult-Submod-Cover unlike the bicriteria one we presented? An interesting example is the problem considered in [30]. The algorithm in this paper for Multi-Partial-SC, like the ones in [6], relies on using the Ellipsoid method to solve the LP with KC inequalities. It may be possible to avoid the inherent inefficiency in this way of solving the LP via some ideas from recent and past work [13, 19].

References

1. Ageev, A.A., Sviridenko, M.I.: Pipage rounding: A new method of constructing algorithms with proven performance guarantee. Journal of Combinatorial Optimization **8**(3), 307–328 (2004)
2. Ahmadian, S., Swamy, C.: Approximation algorithms for clustering problems with lower bounds and outliers. In: 43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy, pp. 69:1–69:15 (2016). DOI 10.4230/LIPIcs.ICALP.2016.69. URL `https://doi.org/10.4230/LIPIcs.ICALP.2016.69`

3. Aronov, B., Ezra, E., Sharir, M.: Small-Size ϵ-Nets for Axis-Parallel Rectangles and Boxes. SIAM J. Comput. **39**(7), 3248–3282 (2010). DOI 10.1137/090762968. URL https://doi.org/10.1137/090762968
4. Bar-Yehuda, R.: Using Homogeneous Weights for Approximating the Partial Cover Problem. Journal of Algorithms **39**(2), 137–144 (2001). DOI 10.1006/jagm.2000.1150. URL http://www.sciencedirect.com/science/article/pii/S0196677400911507
5. Bera, S.K., Chakrabarty, D., Negahbani, M.: Fair algorithms for clustering. CoRR **abs/1901.02393** (2019). URL http://arxiv.org/abs/1901.02393
6. Bera, S.K., Gupta, S., Kumar, A., Roy, S.: Approximation algorithms for the partition vertex cover problem. Theoretical Computer Science **555**, 2–8 (2014)
7. Brönnimann, H., Goodrich, M.T.: Almost Optimal Set Covers in Finite VC-Dimension. Discrete & Computational Geometry **14**(4), 463–479 (1995). DOI 10.1007/BF02570718. URL https://doi.org/10.1007/BF02570718
8. Bshouty, N.H., Burroughs, L.: 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, pp. 298–308 (1998). DOI 10.1007/BFb0028569. URL https://doi.org/10.1007/BFb0028569
9. Buchbinder, N., Feldman, M.: Submodular functions maximization problems. Handbook of Approximation Algorithms and Metaheuristics **1**, 753–788 (2017)
10. Byrka, J., Ghodsi, M., Srinivasan, A.: Lp-rounding algorithms for facility-location problems. CoRR **abs/1007.3611** (2010). URL http://arxiv.org/abs/1007.3611
11. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a submodular set function subject to a matroid constraint (extended abstract). In: Integer Programming and Combinatorial Optimization (IPCO), pp. 182–196 (2007)
12. Călinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. SIAM J. Comput. **40**(6), 1740–1766 (2011)
13. Carr, R.D., Fleischer, L., Leung, V.J., Phillips, C.A.: Strengthening integrality gaps for capacitated network design and covering problems. In: Proceedings of ACM-SIAM SODA, pp. 106–115 (2000)
14. Chan, T.M., Grant, E., Könemann, J., Sharpe, M.: 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, pp. 1576–1585. Society for Industrial and Applied Mathematics (2012)
15. Chan, T.M., Hu, N.: Geometric red-blue set cover for unit squares and related problems. Comput. Geom. **48**(5), 380–385 (2015). DOI 10.1016/j.

comgeo.2014.12.005. URL https://doi.org/10.1016/j.comgeo.2014.12.005
16. Charikar, M., Khuller, S., Mount, D.M., Narasimhan, G.: Algorithms for facility location problems with outliers. In: Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms, pp. 642–651. Society for Industrial and Applied Mathematics (2001)
17. Charikar, M., Panigrahy, R.: Clustering to minimize the sum of cluster diameters. J. Comput. Syst. Sci. **68**(2), 417–441 (2004). DOI 10.1016/j.jcss.2003.07.014. URL https://doi.org/10.1016/j.jcss.2003.07.014
18. Chekuri, C., Jayram, T., Vondrák, J.: On multiplicative weight updates for concave and submodular function maximization. In: Proceedings of ITCS (2015)
19. Chekuri, C., Quanrud, K.: On approximating (sparse) covering integer programs. In: Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1596–1615. SIAM (2019)
20. Chekuri, C., Quanrud, K., Zhang, Z.: On approximating partial set cover and generalizations. CoRR **abs/1907.04413** (2019). URL http://arxiv.org/abs/1907.04413
21. Chekuri, C., Vondrák, J., Zenklusen, R.: Dependent randomized rounding via exchange properties of combinatorial structures. In: 51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA, pp. 575–584 (2010). DOI 10.1109/FOCS.2010.60
22. Chen, A., Harris, D.G., Srinivasan, A.: Partial resampling to approximate covering integer programs. In: Proceedings of 27th ACM-SIAM SODA, pp. 1984–2003 (2016)
23. Clarkson, K.L., Varadarajan, K.: Improved Approximation Algorithms for Geometric Set Cover. Discrete & Computational Geometry **37**(1), 43–58 (2007). DOI 10.1007/s00454-006-1273-8. URL http://dx.doi.org/10.1007/s00454-006-1273-8
24. Dinur, I., Steurer, D.: Analytical Approach to Parallel Repetition. In: Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing, STOC '14, pp. 624–633. ACM, New York, NY, USA (2014). DOI 10.1145/2591796.2591884. URL http://doi.acm.org/10.1145/2591796.2591884
25. Dobson, G.: Worst-case analysis of greedy heuristics for integer programming with nonnegative data. Mathematics of Operations Research **7**(4), 515–531 (1982)
26. Elbassioni, K.M., Krohn, E., Matijevic, D., Mestre, J., Severdija, D.: Improved Approximations for Guarding 1.5-Dimensional Terrains. Algorithmica **60**(2), 451–463 (2011). DOI 10.1007/s00453-009-9358-4. URL https://doi.org/10.1007/s00453-009-9358-4
27. Ezra, E., Aronov, B., Sharir, M.: Improved Bound for the Union of Fat Triangles. In: Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '11, pp. 1778–1785. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2011). URL

http://dl.acm.org/citation.cfm?id=2133036.2133172
28. Gandhi, R., Khuller, S., Srinivasan, A.: Approximation algorithms for partial covering problems. J. Algorithms **53**(1), 55–84 (2004). DOI 10.1016/j.jalgor.2004.04.002. URL https://doi.org/10.1016/j.jalgor.2004.04.002
29. Gibson, M., Pirwani, I.A.: Algorithms for dominating set in disk graphs: Breaking the log$n$ barrier - (extended abstract). In: Algorithms - ESA 2010, 18th Annual European Symposium, Liverpool, UK, September 6-8, 2010. Proceedings, Part I, pp. 243–254 (2010). DOI 10.1007/978-3-642-15775-2_21. URL https://doi.org/10.1007/978-3-642-15775-2_21
30. Har-Peled, S., Jones, M.: Few cuts meet many point sets. CoRR **abs/1808.03260** (2018). URL http://arxiv.org/abs/1808.03260
31. Hong, E., Kao, M.J.: Approximation Algorithm for Vertex Cover with Multiple Covering Constraints. In: W.L. Hsu, D.T. Lee, C.S. Liao (eds.) 29th International Symposium on Algorithms and Computation (ISAAC 2018), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 123, pp. 43:1–43:11. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2018). DOI 10.4230/LIPIcs.ISAAC.2018.43. URL http://drops.dagstuhl.de/opus/volltexte/2018/9991
32. Inamdar, T.: Local search for geometric partial covering problems. In: Proceedings of the 31st Canadian Conference on Computational Geometry, CCCG 2019, Edmonton, Alberta, Canada, August 8-10, 2019, pp. 242–249 (2019)
33. Inamdar, T., Varadarajan, K.R.: On partial covering for geometric set systems. In: 34th International Symposium on Computational Geometry, SoCG 2018, June 11-14, 2018, Budapest, Hungary, pp. 47:1–47:14 (2018). DOI 10.4230/LIPIcs.SoCG.2018.47. URL https://doi.org/10.4230/LIPIcs.SoCG.2018.47
34. Inamdar, T., Varadarajan, K.R.: On the partition set cover problem. CoRR **abs/1809.06506** (2018). URL http://arxiv.org/abs/1809.06506
35. Khuller, S., Moss, A., Naor, J.S.: The budgeted maximum coverage problem. Information processing letters **70**(1), 39–45 (1999)
36. Kolliopoulos, S.G., Young, N.E.: Approximation algorithms for covering/packing integer programs. J. Comput. Syst. Sci. **71**(4), 495–505 (2005). Preliminary version in FOCS 2001
37. Könemann, J., Parekh, O., Segev, D.: A Unified Approach to Approximating Partial Covering Problems. Algorithmica **59**(4), 489–509 (2011). DOI 10.1007/s00453-009-9317-0. URL https://doi.org/10.1007/s00453-009-9317-0
38. Mustafa, N.H., Raman, R., Ray, S.: Settling the APX-hardness status for geometric set cover. In: Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on, pp. 541–550. IEEE (2014)
39. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions—i. Mathematical Pro-

gramming **14**(1), 265–294 (1978)
40. Sviridenko, M.: A note on maximizing a submodular set function subject to a knapsack constraint. Operations Research Letters **32**(1), 41–43 (2004)
41. Varadarajan, K.R.: Epsilon nets and union complexity. In: Proceedings of the 25th ACM Symposium on Computational Geometry, Aarhus, Denmark, June 8-10, 2009, pp. 11–16 (2009). DOI 10.1145/1542362.1542366. URL `http://doi.acm.org/10.1145/1542362.1542366`
42. Varadarajan, K.R.: 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, pp. 641–648 (2010). DOI 10.1145/1806689.1806777. URL `http://doi.acm.org/10.1145/1806689.1806777`
43. Vondrák, J.: Submodularity in combinatorial optimization. Ph.D. thesis, Charles University (2007). Avaulable at `https://theory.stanford.edu/~jvondrak/data/KAM_thesis.pdf`
44. Vondrák, J.: Optimal approximation for the submodular welfare problem in the value oracle model. In: Proceedings of the fortieth annual ACM symposium on Theory of computing, pp. 67–74. ACM (2008)
45. Vondrák, J.: A note on concentration of submodular functions. CoRR **abs/1005.2791** (2010). URL `http://arxiv.org/abs/1005.2791`
46. Wolsey, L.A.: An analysis of the greedy algorithm for the submodular set covering problem. Combinatorica **2**(4), 385–393 (1982)

A Splitting Facilities

In this section, we show how to "split" the facilities in order to satisfy the following properties required by our rounding algorithm for the Facility Location with Multiple Outliers problem.

1. For any facility $i \in F$, $x_i < \tau$.
2. For any client $j \in L$ and any facility $i \in F$, $y_{ij} > 0 \implies y_{ij} = x_i$.
3. $$\sum_{i \in F} \min \left\{ x_i \cdot b'_k, \sum_{j \in C_k(H)} y_{ij} \right\} \geq b'_k \quad \forall\ 1 \leq k \leq r \tag{7}$$

Let $0 < \delta < \tau$ be a small quantity such that all positive x and y-values are integral multiples of δ. Note that assuming the all variables are rational, such a δ must exist.

Fix any facility $i \in F$. Let $X = x_i/\delta$ and $Y_j = y_{ij}/\delta$ for any $j \in L$. By assumption, X and Y_j are integers, and by LP constraint, $Y_j \leq X$ for any $j \in L$. We replace i with multiple (X in number) co-located copies, we denote this set by $\text{copies}(i) = \{i_1, i_2, \ldots, i_X\}$. The x-value of each copy is set to be δ. Now, fix any class C_k. For any client $j \in C_k(H)$ with $y_{ij} > 0$, we will connect j to Y_j distinct copies from $\text{copies}(i)$, and set the y-value of each such assignment equal to δ. Notice that this will satisfy the second property. For any $\ell \in \text{copies}(i)$, let C_k^ℓ denote the clients from $C_k(H)$ that will be assigned to ℓ in this manner. We will also satisfy the following property while making these assignments.

$$\sum_{\ell \in \text{copies}(i)} x_\ell \cdot \min \left\{ b'_k, |C_k^\ell| \right\} \geq \min \left\{ x_i \cdot b'_k, \sum_{j \in C_k(H)} y_{ij} \right\} \tag{8}$$

Notice that that the term on the RHS is exactly the contribution of i to the LHS of Constraint 7, whereas the sum on the LHS is the contribution of the copies after splitting.

Therefore, maintaining this property for all the facilities will guarantee that Constraint 7 holds at the end. Now, we consider two cases about the term on the RHS of Constraint 8.

Case 1. $x_i \cdot b'_k \geq \sum_{j \in C_k(H)} y_{ij}$. Equivalently, $X \cdot b'_k \geq \sum_{j \in C_k(H)} Y_j$.

We process clients $j \in C_k(H)$ with $Y_j > 0$ in an arbitrary order. Let j be the first client in this order. We connect it to the copies $i_1, i_2, \ldots, i_\ell$, i.e., set $y_{i_1 j} = y_{i_2 j} = \ldots = y_{i_\ell j} = \delta$. Here, $\ell = Y_j \leq X$, so we connect j to at most X copies. Now, we consider the second client j', and connect it to the copies $i_{\ell+1}, i_{\ell+2}, \cdots$ where the indices in the subscript are used modulo X. That is, after using i_X for a connection, we use i_1 for the next connection. We continue in this manner for all clients in $C_k(H)$.

Since $Y_j \leq X$ and $\sum_{j \in C_k(H)} Y_j \leq X \cdot b'_k$, it is easy to see we process all clients while connecting at most b'_k clients to each copy. Therefore, we ensure

$$\sum_{\ell \in \text{copies}(i)} \min\{b'_k, |C^\ell_k|\} \geq \sum_{j \in C_k(H)} Y_j.$$

Multiplying both sides by δ, we obtain $\sum_{\ell \in \text{copies}(i)} x_\ell \cdot \min\{k_t(H), |C^\ell_k|\} \geq \sum_{j \in C_k(H)} y_{ij}$, thus ensuring (8).

Case 2. $x_i \cdot b'_k < \sum_{j \in C_k(H)} y_{ij}$. Equivalently, $X \cdot b'_k < \sum_{j \in C_k(H)} Y_j$.

In this case, we arbitrarily and integrally decrease Y-values of clients to obtain Y'-values, so that we have $X \cdot b'_k = \sum_{j \in C_k(H)} Y'_j$. Now, we use the assignment scheme from the previous case to obtain:

$$\sum_{\ell \in \text{copies}(i)} \min\left\{b'_k, |C^\ell_k|\right\} = \sum_{j \in C_t(H)} Y'_j \quad = X \cdot b'_k \tag{9}$$

Now, we increase Y'_j to Y_j and arbitrarily connect client j to $Y_j - Y'_j$ copies to which it is already not connected. Again, since $Y_j \leq X$, there are enough copies available for making these extra connections. This may increase $|C_\ell|$, thereby increasing the LHS of (9). Therefore, we ensure that:

$$\sum_{\ell \in \text{copies}(i)} \min\{b'_k, |C^\ell_k|\} \geq X \cdot b'_k$$

Multiplying both sides by δ, we obtain $\sum_{\ell \in \text{copies}(i)} x_\ell \cdot \min\{b'_k, |C^\ell_k|\} \geq x_i \cdot b'_k = \min\{x_i \cdot b'_k, \sum_{j \in C_k(H)} y_{ij}\}$, thus ensuring (8).

Since any client $j \in L$ belongs to exactly one class, it is part of exactly one reassignment process for each original facility i. Therefore, processing each facility and each class in this manner will result in a new set of facilities that satisfies all the desired properties. In particular, note that for all facilities after splitting, we have $x_i = \delta < \tau$. Furthermore, $x_i = \sum_{\ell \in \text{copies}(i)} x_\ell$, and $y_{ij} = \sum_{\ell \in \text{copies}(i)} y_{\ell j}$, for all $i \in F, j \in L$. Therefore, the cost of the new solution is equal to the cost of the original solution.

Note that using standard tricks, we can ensure that $1/\delta$ is polynomial in the input, at the expense of a tiny but insignificant increase in the cost of the solution, which can be absorbed in the $O(\log r)$ approximation guarantee. Thus, the splitting procedure runs in polynomial time, and the size of resulting instance is also polynomial in the input.