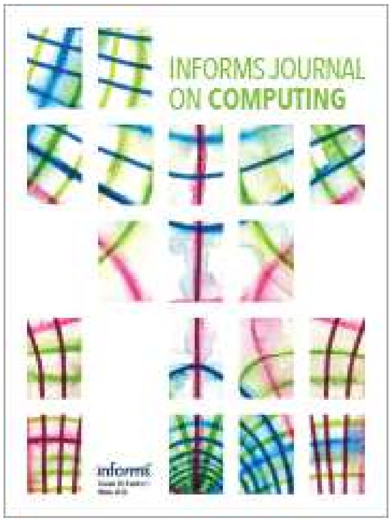

INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Breaking the rmax Barrier: Enhanced Approximation Algorithms for Partial Set Multicover Problem

Yingli Ran, Zhao Zhang, Shaojie Tang, Ding-Zhu Du

informs
http://pubsonline.informs.org/journal/ijoc

INFORMS JOURNAL ON COMPUTING
Vol. 33, No. 2, Spring 2021, pp. 774–784
ISSN 1091-9856 (print), ISSN 1526-5528 (online)

Breaking the r_{max} Barrier: Enhanced Approximation Algorithms for Partial Set Multicover Problem

Yingli Ran,[a] Zhao Zhang,[a] Shaojie Tang,[b] Ding-Zhu Du[c]

[a] College of Mathematics and Computer Sciences, Zhejiang Normal University, Jinhua, Zhejiang 321004, China; [b] Naveen Jindal School of Management, University of Texas at Dallas, Richardson, Texas 75080; [c] Department of Computer Science, University of Texas at Dallas, Richardson, Texas 75080

Contact: 724609171@qq.com (YR); hxhzz@sina.com, https://orcid.org/0000-0003-4191-7598 (ZZ); shaojie.tang@utdallas.edu, https://orcid.org/0000-0001-9261-5210 (ST); dzdu@utdallas.edu (D-ZD)

Received: September 17, 2019
Revised: January 26, 2020; February 16, 2020
Accepted: March 4, 2020
Published Online in Articles in Advance: October 13, 2020

https://doi.org/10.1287/ijoc.2020.0975

Abstract. Given an element set E of order n, a collection of subsets $S \subseteq 2^E$, a cost c_S on each set $S \in S$, a covering requirement r_e for each element $e \in E$, and an integer k, the goal of a minimum partial set multicover problem (MinPSMC) is to find a subcollection $\mathcal{F} \subseteq S$ to fully cover at least k elements such that the cost of F is as small as possible and element e is fully covered by F if it belongs to at least r_e sets of F. This problem generalizes the minimum k-union problem (Min$k$U) and is believed not to admit a subpolynomial approximation ratio. In this paper, we present a $(4\log nH(\Delta)\ln k + 2\log n\sqrt{n})$-approximation algorithm for MinPSMC, in which Δ is the maximum size of a set in S. And when $k = \Omega(n)$, we present a bicriteria algorithm fully covering at least $(1 - \frac{\varepsilon}{2\log n})k$ elements with approximation ratio $O(\frac{1}{\varepsilon}(\log n)^2 H(\Delta))$, where $0 < \varepsilon < 1$ is a fixed number. These results are obtained by studying the minimum density subcollection problem with (or without) cardinality constraint, which might be of interest by itself.

History: Accepted by Andrea Lodi, Area Editor for Design & Analysis of Algorithms – Discrete.
Funding: This work was supported by the National Science Foundation [Grant 1907472], the National Natural Science Foundation of China [Grants 11531011, 11771013, 11901533, and 61751303], and the Zhejiang Provincial Natural Science Foundation of China [Grants LD19A010001 and LY19A010018].

Keywords: partial set multicover • minimum *k* union • approximation algorithm

1. Introduction

In this paper, we consider the following partial set multicover problem. We are given an element set E of order n, a collection of subsets $\mathcal{S} \subseteq 2^E$, a cost c_S on each set $S \in \mathcal{S}$, a covering requirement r_e for each element $e \in E$, and an integer k; the goal is to find a subcollection $\mathcal{F} \subseteq \mathcal{S}$ to fully cover at least k elements such that the cost of $\mathcal{F}$ is as small as possible and element e is fully covered by $\mathcal{F}$ if it belongs to at least r_e sets of $\mathcal{F}$. We call this the *minimum partial set multicover problem* (MinPSMC). MinPSMC can be viewed as a generalization of both the *minimum partial set cover problem* (MinPSC) and the *minimum set multicover problem* (MinSMC); MinPSC is the special case of MinPSMC with $r_e = 1$ for all $e \in E$, and MinSMC is the special case of MinPSMC with $k = n$. There are a lot of studies on MinPSC and MinSMC, achieving tight performance ratios (Slavík 1997, Vazirani 2013). However, study on MinPSMC is very rare.

An important special case of MinPSMC is the *minimum k union* (Min$k$U) problem. In Min$k$U, we are given a hypergraph G with vertex set V and hyperedge set $\mathcal{H}$ together with an integer $1 \le k \le m$, where m is the number of hyperedges; the goal is to select k hyperedges such that their union is as small as possible. Notice that the Min$k$U problem is equivalent to finding a minimum vertex set $V' \subseteq V$, which contains at least k hyperedges, and thus, a Min$k$U instance $(V, \mathcal{H}, k)$ can be viewed as a MinPSMC instance $(E, \mathcal{S}, c, r, k)$ with $E = \mathcal{H}$, $\mathcal{S} = V$, $c \equiv 1$, and $r_e = f_e$ for every $e \in E$, where $f_e = |\{S \in \mathcal{S} : e \in S\}|$ is the number of sets containing element e (called *frequency* of e). To be more concrete, a vertex $v \in V = \mathcal{S}$ is viewed as a subset of $E = \mathcal{H}$ containing all those hyperedges $e = H \in \mathcal{H}$ with $v \in H$. In other words, $v \in \mathcal{S}$ covers element $e = H$ if $v \in H$. For an element $e = H \in \mathcal{H}$, its frequency $f_e = |H|$. So $r_e = f_e$ implies that element $e = H$ is fully covered by a subset $V' \subseteq V$ (which is viewed as a subcollection of $\mathcal{S}$) if and only if $H \subseteq V'$. The Min$k$U problem is closely related with the *small set vertex expansion problem* (SSVE) and the *small set expansion problem* (SSE). Min$k$U is also a generalization of the *smallest k-edge subgraph problem* (SkES), the dual of which is the *densest k-subgraph problem* (DkS). These problems have received a lot of attention in recent years. One reason is because of their relations with many important combinatorial problems such as the *node weighted Steiner network problem* (Nutov 2010) and the *Steiner k-forest problem* (Hajiaghayi and Jain 2006). Another reason is because of their applications in the fields of cryptographic systems (Applebaum et al. 2010) and financial derivatives (Arora et al. 2010), etc.

Another important special case of MinPSMC is the *minimum partial positive dominating set problem* (MinPPDS). For a graph G, a real number $0 < \rho \le 1$, and a subset of vertices $D \subseteq V(G)$, a vertex $v \in V(G)$ is *positive dominated* by D if it has at least $\lceil \rho d(v) \rceil$ neighbors in D, where $d(v)$ is the degree of vertex v in G. Given an integer $k \le |V(G)|$, the goal of MinPPDS is to select a minimum vertex set $D \subseteq V(G)$ such that at least k vertices are positive dominated by D. When $k = n$, the problem is called the *minimum positive dominating set* (MinPDS), which was formulated by Wang et al. (2009, 2011) as a model to minimize seed selection to influence all people in a social network. In the real world, it might be too expensive to influence all people. In view of cost effectiveness, it might be satisfactory to influence only some percentage of the people. Such a consideration leads to the MinPPDS problem, which was first studied by Ran et al. (2017b).

1.1. Related Works

Min$k$U was proposed by Chlamtáč et al. (2018), in which a $2\sqrt{m}$ approximation algorithm was given, in which m is the number of hyperedges. Later, Chlamtáč et al. (2017) improved the ratio to $O(m^{1/4+\varepsilon})$.

Min$k$U is equivalent to the *small set bipartite vertex expansion* problem proposed in Chlamtáč et al. (2017), which considers a bipartite graph $G = (U, V, E)$; the goal is to choose k nodes in U to minimize the size of their neighborhood. This is the bipartite version of the SSVE problem, in which we are given an arbitrary graph $G = (V, E)$ and are asked to choose a subset $S \subseteq V$ with size k to minimize $|N_G(S)|$, where $N_G(S) = \{u \in \bar{S} : \exists v \in S \text{ such that } uv \in E\}$ is the (vertex) neighbor set of S. A related problem is the SSE problem, in which we are given a graph $G = (V, E)$ and asked to choose a subset $S \subseteq V$ with size k to minimize $E(S, \bar{S})$, where $E(S, \bar{S}) = \{uv \in E : u \in S, v \in \bar{S}\}$ is the edge cut associated with S. Most studies on SSE and SSVE are focused on the case when k is very close to n. For SSVE, Louis and Makarychev (2014) gave a bicriteria algorithm when $k = \Omega(n)$. This algorithm was refined by Chlamtáč et al. (2017), outputting a solution with size at most $(1 + \varepsilon)k$ and approximation ratio $\min\{O(\sqrt{\log n} \cdot p^{-1} \log p^{-1} \log\log p^{-1}/\varepsilon), O(k \log n)\}$, where $0 < \varepsilon < 1$ is an arbitrary constant, and $p = k/n$.

The SkES is the Min$k$U problem restricted on simple graphs. For SkES, the current best approximation ratio is $O(m^{3-2\sqrt{2}+\varepsilon})$ for any constant $\varepsilon > 0$ (Chlamtáč et al. 2012), and m is the number of edges. This ratio is tight assuming "dense versus random" conjecture for DkS (Bhaskara et al. 2010), for which DkS is a dual version of SkES. Given a graph $G = (V, E)$ and an integer k, the goal of DkS is to find a subset $V' \subseteq V$ with cardinality k that maximizes the number of edges in the subgraph of G induced by V'. For DkS, Asahiro et al. (2002) presented a greedy algorithm with approximation ratio $O(n/k)$, where n is the number of vertices in the graph; Kortsarz and Peleg (1993) gave an $O(n^{2/5})$-approximation; Feige et al. (2001) improved the ratio to $n^{1/3-\varepsilon}$ for any constant $\varepsilon > 0$. The current best approximation ratio is $O(n^{1/4+\varepsilon})$ for any constant $\varepsilon > 0$, which was obtained in Bhaskara et al. (2010).

MinPDS (the full version of MinPPDS) was first proposed by Wang et al. (2009, 2011). It was shown that MinPDS is APX-hard, and a greedy algorithm achieves approximation ratio $H(\delta_{\max})$, where $\delta_{\max}$ is the maximum degree of the graph and $H(\delta) = \sum_{i=1}^{\delta} 1/i$ is the δth harmonic number. Dinh et al. (2014) showed that MinPDS cannot be approximated within a factor of $(1-\epsilon)\ln\max\{\delta_{\max}, \sqrt{|V|}\}$, where ε is an arbitrary positive real number smaller than one. They also presented a $(\ln\delta_{\max} + O(1))$-approximation algorithm (recall that $\ln\delta \le H(\delta) \le \ln\delta + 1$) by observing that MinPDS is a special case of the MinSMC problem.

Considering partial requirement, MinPDS was generalized to MinPPDS by Ran et al. (2017b). A greedy algorithm was proposed achieving approximation ratio $\gamma H(\delta_{\max})$, where $\gamma = 1/(1-(1-p)\eta)$, $p = k/n$ is the required covering percentage, $\eta \approx \delta_{\max}^2/\delta_{\min}$, and $\delta_{\max}, \delta_{\min}$ are the maximum and minimum degrees of the graph, respectively.

For MinSMC, Vazirani (2013) and Rajagopalan and Vazirani (1998) both presented algorithms achieving approximation ratio $H(n)$, where n is the number of elements to be covered. MinPSC was first studied by Kearns (1990), and a greedy algorithm was presented with approximation ratio at most $2H(n) + 3$. Slavík (1997) improved the algorithm, obtaining approximation ratio $\min\{H(\Delta), H(k)\}$, where $\Delta = \max\{|S| : S \in \mathcal{S}\}$. Using the primal-dual method, Gandhi et al. (2004) gave an algorithm with approximation ratio f, where f is the maximum number of sets containing a common element. The same ratio f was also obtained by Bar-Yehuda (2001) using the local ratio method. From these related works, it can be seen that both MinPSC and MinSMC admit approximation ratios matching those best ratios for the classic set cover problem.

Ran et al. (2017b) were the first to study an approximation algorithm for MinPSMC. However, their ratio is meaningful only when the covering percentage $p = k/n$ is very close to one. Afterward, Ran et al. (2017a) presented a simple greedy algorithm for MinPSMC achieving approximation ratio Δ. Zhang et al. (2017) claimed to have obtained an $O(r_{\max} \log^3 n)$-approximation for MinPSMC, in which $r_{\max} = \max\{r_e : e \in E\}$ is the maximum covering requirement. However, there is a flaw in it. In their corrected version (Shi et al. 2019), under the assumption that $k = pn$ with

$0 < p < 1$, they in fact obtained a bicriteria algorithm with cost at most $O(r_{\max}(\log n)^2(1+\ln(1/\varepsilon)+\frac{1-p}{\varepsilon p}))$ times the optimal value while the number of elements fully covered is at least $(1-\varepsilon)k$. Under the same assumptions, Shi et al. (2020) gave a randomized bicriteria algorithm with approximation ratio $O(b/p\varepsilon)$, where $b = \max\{\binom{f_e}{r_e} : e \in E\}$. In a general case when $r_{\max}$ is large, these ratios might be very large. Liu and Huang (2018) show the advantage of using partial multicover than using full multicover.

1.2. Motivation and Results

In previous works (Shi et al. 2019, 2020; Ran et al. 2020), the authors studied the MinPSMC problem under the assumption that $r_{\max}$ is upper bounded by a constant, and $r_{\max} = \max\{r_e : e \in E\}$ is the maximum covering requirement for elements. One main reason for such an assumption is that their approximation ratios depend on a parameter $b = \max\{\binom{f_e}{r_e} : e \in E\}$. Notice that $b \le f^{r_{\max}}$. If $r_{\max}$ is not upper bounded by a constant, then b might be very large. For many real-world applications, such an assumption may not hold. For example, in the MinPPDS problem with $\rho = 1/2$, we have $r_{\max} = \lceil \delta_{\max}/2 \rceil$, where $\delta_{\max}$ is the maximum degree of the graph, which can be as large as $n-1$ in a worst case. One typical feature of a social network is the *power-law property*, and it is known that in a power-law graph, the maximum degree is $\Theta(\sqrt[\beta]{n})$ for some parameter β. In this case, the performance of those algorithms proposed in Ran et al. (2020) and Shi et al. (2020, 2019) could be very bad. The motivation of this paper is trying to design an approximation algorithm that does not require $r_{\max}$ to be a constant. This is done through studying a related *minimum density subcollection problem* (MinDSC) the formal definition of which is given in Section 2.1.

The MinDSC problem was first proposed in Zhang et al. (2017) and was proved to be NP-hard by a reduction from the three-dimensional matching problem. An $O(r_{\max}\log^2 n)$-approximation algorithm was presented. In this paper, we improve the algorithm, obtaining a ratio at most $4\log nH(\Delta)$ with Δ being the maximum size of a set in $\mathcal{S}$, which no longer depends on $r_{\max}$. Making use of this result, we design a greedy algorithm for MinPSMC whose approximation ratio is at most $4\log nH(\Delta)\ln k + 2\log n\sqrt{n}$.

Although the main idea follows that of Shi et al. (2019), we make two significant and nontrivial improvements to their work. First, we exploit a relation between their linear program (LP) formulation with the MinDSC problem to remove the dependence on $r_{\max}$. Second, their approximation algorithm is a bicriteria one with a violation of factor $(1+\varepsilon)$ to the total covering requirement, and their approximation ratio depends on $1/\varepsilon$ and $1/p$, where $p = k/n$. So, if k is very small, then their ratio is very large, and the approximation ratio we obtain in this paper is in its classic sense (without violation) and also works for small k.

Furthermore, we present a bicriteria algorithm fully covering at least $(1-\frac{\varepsilon}{2\log n})k$ elements with approximation ratio $\frac{8n}{\varepsilon k}(\log n)^3$, where $\varepsilon > 0$ is an arbitrary constant. In particular, our ratio is $O(\frac{1}{\varepsilon}(\log n)^2 H(\Delta))$ when $k = pn$, and $0 < p < 1$ is a constant. For this purpose, we define a new problem called *minimum density subcollection fully covering* $\ge k$ *elements* (Min DSC$^{\ge k}$), the goal of which is to find a minimum density subcollection among those that fully cover at least k elements. The cardinality constraint makes the problem much more complicated. New ideas and more delicate analysis are involved in studying MinDSC$^{\ge k}$. We design an approximation algorithm for MinDSC$^{\ge k}$, the output of which either fully covers at least $(1-\frac{\varepsilon}{2\log n})k$ elements and has density at most $8(\log n)^2 H(\Delta)\cdot opt_{DSC^{\ge k}}$ or is a feasible solution with density at most $\frac{8(\log n)^2 H(\Delta)}{\varepsilon} opt_{DSC^{\ge k}}$, where $opt_{DSC^{\ge k}}$ is the optimal value of MinDSC$^{\ge k}$. Based on this algorithm, a bicriteria algorithm for MinPSMC is obtained. Compared with the studies in Shi et al. (2019, 2020), our ratio is independent of $r_{\max}$. Furthermore, the studies in Shi et al. (2019, 2020) only consider the case when $k = \Omega(n)$, and our results work for a general k. Our ratio is also an improvement on that obtained in Shi et al. (2019).

2. Algorithm for MinPSMC

In the following, we first give the approximation algorithm for the MinDSC problem. Then, we make use of the solution for MinDSC to solve MinPSMC.

2.1. Algorithm for MinDSC

We present the formal definition of the MinDSC problem as follows.

Definition 1 (MinDSC). Suppose E is an element set, $\mathcal{S} \subseteq 2^E$ is a collection of subsets of E, $c : \mathcal{S} \mapsto \mathbb{R}^+$ is a cost function on $\mathcal{S}$, $r : E \mapsto \mathbb{Z}^+$ is a covering requirement function on E. For a subcollection $\mathcal{F} \subseteq \mathcal{S}$, the *density* of $\mathcal{F}$ is defined as

$$den(\mathcal{F}) = \frac{c(\mathcal{F})}{|\mathcal{C}(\mathcal{F})|}.$$

Here, $\mathcal{C}(\mathcal{F})$ is the set of elements fully covered by $\mathcal{F}$ and $|\mathcal{C}(\mathcal{F})|$ is the cardinality of $\mathcal{C}(\mathcal{F})$. The goal of MinDSC is to find a subcollection $\mathcal{F}$ with the minimum density.

We design an approximation algorithm for MinDSC based on an LP formulation. As shown by Shi et al. (2020), a natural LP for MinDSC has integrality gap arbitrarily large. So, a new LP formulation is given in Shi et al. (2020). To introduce this LP formulation, we need the following terminologies.

For an element e, an r_e-*cover set* is a subcollection $\mathcal{S}' \subseteq \mathcal{S}_e = \{S \in \mathcal{S} : e \in S\}$ with $|\mathcal{S}'| = r_e$. So any r_e-cover set fully covers element e. Denote by Ω_e the family of all r_e-cover sets and $\Omega = \bigcup_{e \in E} \Omega_e$ the family of all cover sets. The following example illustrates theses concepts.

Example 1. Let $E = \{e_1, e_2, e_3\}$, $\mathcal{S} = \{S_1, S_2, S_3\}$ with $S_1 = \{e_1, e_2\}$, $S_2 = \{e_1, e_3\}$, $S_3 = \{e_1, e_2, e_3\}$, $r(e_i) = 2$ for $i = 1, 2, 3$. For this example, $\Omega_{e_1} = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}$ with $\mathcal{Q}_1 = \{S_1, S_2\}^{e_1}$, $\mathcal{Q}_2 = \{S_1, S_3\}^{e_1}$, $\mathcal{Q}_3 = \{S_2, S_3\}^{e_1}$, $\Omega_{e_2} = \{\mathcal{Q}_4\}$ with $\mathcal{Q}_4 = \{S_1, S_3\}^{e_2}$, $\Omega_{e_3} = \{\mathcal{Q}_5\}$ with $\mathcal{Q}_5 = \{S_2, S_3\}^{e_3}$, and $\Omega = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_5\}$.

Remark 1. Notice that different elements may have a same collection of sets as cover sets. For the preceding example, $\{S_1, S_3\}$ is an r_{e_1}-cover set as well as an r_{e_2}-cover set. In this case, this collection of sets should be viewed as different r_e-cover sets. This is why we use superscript e_1 and e_2 to distinguish them. The idea behind this definition is that if an r_e-cover set $\mathcal{Q} \in \Omega$ is taken, then e is fully covered by those sets in $\mathcal{Q}$.

The following is an integer program for MinDSC:

$$\min \frac{\sum_{S \in \mathcal{S}} c_S x_S}{\sum_{e \in E} y_e}$$
$$\text{s.t.} \begin{cases} \sum_{\mathcal{Q} : \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}} \ge y_e, & \forall e \in E, \\ x_S \ge \max_{\mathcal{Q} : S \in \mathcal{Q} \in \Omega} l_{\mathcal{Q}}, & \forall S \in \mathcal{S}, \\ x_S \in \{0, 1\}, & \forall S \in \mathcal{S}, \\ y_e \in \{0, 1\}, & \forall e \in E, \\ l_{\mathcal{Q}} \in \{0, 1\}, & \forall \mathcal{Q} \in \Omega. \end{cases} \tag{1}$$

In this integer program, $y_e = 1$ indicates that element e is fully covered. The first constraint says that, in order that e is fully covered, at least one r_e-cover set is chosen. The second constraint says that a set S must be picked if it belongs to some cover set that has been chosen. It should be remarked that there exists an optimal solution to (1) such that, for any element e with $y_e = 1$, there is *exactly one* r_e-cover set Q that has $l_{\mathcal{Q}} = 1$ and all the other r_e-cover sets have l-value zero. In fact, suppose (x, y, l) is an optimal solution to (1), then, for any element e with $y_e = 1$, to satisfy the first constraint of (1), there is at least one r_e-cover set with l-value one. If there are more r_e-cover sets with l-value one, we can modify the solution as follows. Suppose $\Omega_e = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_t\}$ and $l(\mathcal{Q}_1) = \cdots = l(\mathcal{Q}_s) = 1$, $l(\mathcal{Q}_{s+1}) = \cdots = l(\mathcal{Q}_t) = 0$, where $2 \le s \le t$. Reset $l(\mathcal{Q}_1) = 1$ and $l(\mathcal{Q}_i) = 1$ for $i = 2, \ldots, t$. After such a reassignment, all constraints in (1) are still satisfied but the object value is not affected. Hence, we may replace the second constraint of (1) by

$$x_S \ge \sum_{\mathcal{Q} : S \in \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}}, \forall S \in \mathcal{S} \text{ and } \forall e \in E$$

without changing the optimal value. The modified integer program can be relaxed to the following linear program (LP_1):

$$\min \sum_{S \in \mathcal{S}} c_S x_S$$
$$\text{s.t.} \begin{cases} \sum_{e \in E} y_e = 1 \\ \sum_{\mathcal{Q} : \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}} \ge y_e, & \forall e \in E, \\ x_S \ge \sum_{\mathcal{Q} : S \in \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}}, & \forall S \in \mathcal{S} \text{ and } \forall e \in E, \\ x_S \ge 0, & \forall S \in \mathcal{S}, \\ y_e \ge 0, & \forall e \in E, \\ l_{\mathcal{Q}} \ge 0, & \forall \mathcal{Q} \in \Omega. \end{cases} \tag{2}$$

Notice that there exists an optimal solution to (2) in which all variables have values at most one. This is obvious for y_e. If the second constraint is strict, then we may reduce the values of some $l_{\mathcal{Q}}$ that neither violates the feasibility nor increases the objective value. So we may assume that $\sum_{\mathcal{Q} : \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}} = y_e$, and thus, $l_{\mathcal{Q}} \le 1$ because $y_e \le 1$. As a consequence, the right-hand side of the third constraint is no more than one. Combining this with the fact that the objective is to minimize a linear combination of x_S's, we have $x_S \le 1$ for any $S \in \mathcal{S}$. The following lemma shows that (2) is indeed a relaxation of the MinDSC problem.

Lemma 1. *The optimal value of the linear program* (2)*, denoted as* opt_{LP_1}*, satisfies* $opt_{LP_1} \le opt_{MinDSC}$*, where* opt_{MinDSC} *is the optimal value for MinDSC.*

Proof. For any optimal solution $(x^*, y^*, \mathcal{Q}^*)$ of the MinDSC instance, suppose $\sum_{e \in E} y_e^* = N$, then $(x^*/N, y^*/N, \mathcal{Q}^*/N)$ satisfies all constraints of (2) and, thus, is a feasible solution to (2). As a consequence, $opt_{LP_1} \le \sum_{S \in \mathcal{S}} c_S(x_S^*/N) = \sum_{S \in \mathcal{S}} c_S x_S^* / \sum_{e \in E} y_e^* = opt_{MinDSC}$.

Lemma 2. *Program* (2) *is polynomial-time solvable.*

Proof. Observe that the number of variables in the form of $l_{\mathcal{Q}}$ in program (2) is exponential. Consider the dual program of (2):

$$\max\ t$$
$$\text{s.t.} \begin{cases} \sum_{e \in E} p_{eS} \le c_S, & \forall S \in \mathcal{S} \\ z_e \le \sum_{S : S \in \mathcal{Q}} p_{eS}, & \forall e \in E \text{ and } \forall \mathcal{Q} \in \Omega_e \\ t \le z_e, & \forall e \in E, \\ p_{eS} \ge 0, & \forall S \in \mathcal{S} \text{ and } \forall e \in E \\ z_e \ge 0, & \forall e \in E. \\ t \ge 0 \end{cases} \tag{3}$$

By LP primal-dual theory (see, for example, Korte and Vygen 2008), optimal primal solution and optimal dual solution can be determined by each other through complementary slackness conditions. Notice that the number of nonzero primal variables is upper bounded by the rank of the primal coefficient matrix and, thus, is a polynomial; hence, the exponential number of primal variables does not cause trouble because we only need to determine those that are nonzero. As a consequence, one can solve (2) by solving (3). Although (3) has an exponential number of constraints, the ellipsoid algorithm tells us that, as long as one can find a polynomial time separation oracle, then the LP can be solved within any additive error in polynomial time. A procedure is called a separation oracle for an LP if, given a point x, it either correctly decides that x belongs to the feasible domain of the LP or finds a violated constraint at x. Because both $|\mathcal{S}|$ and $|E|$ are polynomial, to solve (3), it suffices to construct a separation oracle for the second set of constraints, which is accomplished in the following way.

For any element $e \in E$, let g_e be the function on Ω_e defined by $g_e(\mathcal{Q}) = \sum_{S:S\in\mathcal{Q}} p_{eS}$ for $\mathcal{Q} \in \Omega_e$. Notice that g_e is a modular set function on Ω_e; that is, for any two r_e-cover sets $\mathcal{Q}_1, \mathcal{Q}_2 \in \Omega_e$, $g_e(\mathcal{Q}_1 \cup \mathcal{Q}_2) + g_e(\mathcal{Q}_1 \cap \mathcal{Q}_2) = g_e(\mathcal{Q}_1) + g_e(\mathcal{Q}_2)$. Because the minimum value of a modular set function can be found in polynomial time even when its feasible domain has exponential size (in fact, even the minimization of a submodular set function can be done in polynomial time; Fujishige 2005), we can compare the minimum value of g_e with z_e for each e in polynomial time. If there exists an element e with $\min_{\mathcal{Q}\in\Omega_e} g_e(\mathcal{Q}) < z_e$, then we find a violated constraint. Otherwise, we can draw the conclusion that the current $(\{p_{eS}\}, \{z_e\}, t)$ is a feasible solution. This serves as a separation oracle for the second set of constraints, and the proof is completed because of the preceding argument.

Our algorithm is presented in Algorithm 1. It differs from the algorithm of Shi et al. (2020) in line 4. In Shi et al. (2020), the authors employ an approximation algorithm for the *minimum node weighted Steiner network problem* to find a multicover of Y_{i_0}, and our algorithm makes use of an approximation algorithm for MinSMC directly.

Algorithm 1 (Algorithm for MinDSC)
Input: A MinDSC instance $(E, \mathcal{S}, c, r)$.
Output: A subcollection $\mathcal{S}'$.
1: Find an optimal solution (x^f, y^f, l^f) to linear program (2).
2: Let $Y_i = \{e \in E : 2^{-(i+1)} < y_e^f \le 2^{-i}\}$ for $0 \le i \le I-1$ and $Y_I = \{e \in E : y_e^f \le 2^{-I}\}$, where $I = 2\lfloor \log n \rfloor - 1$.
3: Let i_0 be an index such that $|Y_{i_0}| \ge 2^{i_0}/(I+1)$.
4: Find an approximate solution $\mathcal{S}'$ to MinSMC on instance $(Y_{i_0}, \mathcal{S}, c, r)$.
5: Output $\mathcal{S}'$.

The improvement on the approximation ratio is achieved by exploiting a relation between (LP_1) and an LP formulation for MinSMC, denoted as (LP_2) as follows:

$$\min \sum_{S\in\mathcal{S}} c_S x_S$$
$$\text{s.t.} \begin{cases} \sum_{S:e\in S} x_S \ge r_e, & \forall e \in Y, \\ x_S \le 1, & \forall S \in \mathcal{S}, \\ x_S \ge 0, & \forall S \in \mathcal{S}. \end{cases} \tag{4}$$

where Y is taken to be Y_{i_0} here. Notice that, for MinSMC, a set can be taken at most once. Because, in a general case, we have $r_e > 1$, constraint $x_S \le 1$ cannot be omitted from the LP relaxation. It is known that the preceding program (4) has integrality gap $H(\Delta)$, where Δ is the maximum size of a set in $\mathcal{S}$ (Rajagopalan and Vazirani 1998).

Theorem 1. *For $n \ge 32$, Algorithm 1 has a performance ratio at most $4\log nH(\Delta)$ for MinDSC.*

Proof. We prove the theorem by first establishing the following two claims.

Claim 1. An index i_0 as in line 3 of Algorithm 1 exists, and $i_0 \le I-1$.

Proof. In fact, by $\sum_{e\in E} y_e = 1$, there exists an index i_0 such that $\sum_{e\in Y_{i_0}} y_e^f \ge 1/(I+1)$. Because $y_e^f \le 2^{-i_0}$ for every $e \in Y_{i_0}$, we have $|Y_{i_0}| \ge 2^{i_0}/(I+1)$. Notice that, for $i = I$, it can be calculated that $\sum_{e\in Y_I} y_e^f \le n2^{-I} < 1/(I+1)$ when $n \ge 32$. Hence, the preceding $i_0 \le I-1$.

Claim 2. $opt_{LP_2}(Y_{i_0}) \le 2^{i_0+1} opt_{LP_1}$, where $opt_{LP_2}(Y_{i_0})$ is the optimal value of linear program (4) on MinSMC instance $(Y_{i_0}, \mathcal{S}, c, r)$.

Proof. Consider x^f computed in line 1 of Algorithm 1. Construct a new vector $\{\hat{x}_S\}_{S\in\mathcal{S}}$ by setting $\hat{x}_S = \min\{1, 2^{i_0+1} x_S^f\}$ for $S \in \mathcal{S}$. We claim that

$$\{\hat{x}_S\}_{S\in\mathcal{S}} \text{ is a feasible solution to (4) on instance } (Y_{i_0}, \mathcal{S}, c, r). \tag{5}$$

Clearly, $\hat{x}_S \le 1$. So we only need to show that the first constraint in (4) is satisfied. For this purpose, denote $\mathcal{H} = \{S \in \mathcal{S} : 2^{i_0+1} x_S^f > 1\}$ and $\mathcal{H}_e = \{S \in \mathcal{H} : e \in S\}$. Then every $S \in \mathcal{H}$ has $\hat{x}_S = 1$. Let $\mathcal{S}^r = \mathcal{S} \setminus \mathcal{H}$, $r_e^r = \max\{0, r_e - |\mathcal{H}_e|\}$, and $E^r = Y_{i_0} \setminus \{e : r_e^r = 0\}$ (where superscript r represents *residual*). To prove the first constraint in (4), it suffices to show that

$$\sum_{S:e\in S\in\mathcal{S}^r} \hat{x}_S \ge r_e^r \text{ holds for any } e \in E^r. \tag{6}$$

Denote by $\mathcal{Q}^r, \Omega_e^r, \Omega^r$ the restriction of $\mathcal{Q}, \Omega_e, \Omega$ on $(E^r, \mathcal{S}^r, r^r)$ and set $\hat{l}_{\mathcal{Q}^r} = l_{\mathcal{Q}}^f$ for $Q^r \neq \emptyset$. For any element $e \in E^r$, we have $r_e^r > 0$, and thus, $r_e > |\mathcal{H}_e|$. So, for any $\mathcal{Q}^r \in \Omega_e^r$, the set

$$\mathcal{Q}^r = \mathcal{Q} \setminus \mathcal{H}_e \neq \emptyset \tag{7}$$

and $\hat{l}_{\mathcal{Q}^r}$ inherits the value $l_{\mathcal{Q}}^f$. Combining this argument with the fact that (x^f, l^f, y^f) is a feasible solution to (2), we have

$$\sum_{\mathcal{Q}^r\,:\,\mathcal{Q}^r \in \Omega_e^r} \hat{l}_{\mathcal{Q}^r} = \sum_{\mathcal{Q}\,:\,\mathcal{Q} \in \Omega_e} l_{\mathcal{Q}}^f \geq y_e^f. \tag{8}$$

By the nonnegativity of $l_{\mathcal{Q}}^f$, we have

$$x_S^f \geq \sum_{\mathcal{Q}\,:\,S \in \mathcal{Q} \in \Omega_e} l_{\mathcal{Q}}^f \geq \sum_{\mathcal{Q}^r\,:\,S \in \mathcal{Q}^r \in \Omega_e^r} \hat{l}_{\mathcal{Q}^r} \quad (\forall S \in \mathcal{S}^r). \tag{9}$$

Observe that the cover set $\mathcal{Q}^r$ defined in (7) contains at least r_e^r sets, and we have

$$\sum_{S\,:\,S \in \mathcal{Q}^r} \hat{l}_{\mathcal{Q}^r} \geq r_e^r. \tag{10}$$

It follows that, for any $e \in E^r$,

$$\begin{aligned}
\sum_{S\,:\,e \in S \in \mathcal{S}^r} \hat{x}_S &= \sum_{S\,:\,e \in S \in \mathcal{S}^r} 2^{i_0+1} x_S^f \\
&\geq 2^{i_0+1} \sum_{S\,:\,e \in S \in \mathcal{S}^r} \sum_{\mathcal{Q}^r\,:\,S \in \mathcal{Q}^r \in \Omega_e^r} \hat{l}_{\mathcal{Q}^r} \\
&\quad \text{(by (9))} \\
&= 2^{i_0+1} \sum_{\mathcal{Q}^r\,:\,\mathcal{Q}^r \in \Omega_e^r} \sum_{S\,:\,S \in \mathcal{Q}^r} \hat{l}_{\mathcal{Q}^r} \\
&\quad \text{(by exchangng the summation)} \\
&\geq 2^{i_0+1} \sum_{\mathcal{Q}^r\,:\,\mathcal{Q}^r \in \Omega_e^r} r_e^r \cdot \hat{l}_{\mathcal{Q}^r} \\
&\quad \text{(by (10))} \\
&\geq 2^{i_0+1} r_e^r y_e^f \\
&\quad \text{(by (8))} \\
&> r_e^r \qquad \Big(\text{since } e \in E^r \subseteq Y_{i_0} \text{ and every} \\
&\quad \times \text{element } e \in Y_{i_0} \text{ has } 2^{i_0+1} y_e^f > 1\Big).
\end{aligned}$$

Inequality (6) is proved, and (5) follows.

Combining (5) with the observation that $\hat{x}_S \leq 2^{i_0+1} x_S^f$ (by the definition of $\hat{x}_S$), we have

$$opt_{LP_2}(Y_{i_0}) \leq \sum_{S\,:\,S \in \mathcal{S}} c_S \hat{x}_S \leq 2^{i_0+1} \sum_{S\,:\,S \in \mathcal{S}} c_S x_S^f = 2^{i_0+1} opt_{LP_1}.$$

Claim 2 is proved.

Then, we have the following sequence of inequalities:

$$\begin{aligned}
\frac{c(\mathcal{S}')}{|\mathcal{C}(\mathcal{S}')|} &\leq \frac{c(\mathcal{S}')}{|Y_{i_0}|} \\
&\quad \big(\text{since } \mathcal{S}' \text{ fully covers all the elements in } Y_{i_0}\big) \\
&\leq \frac{H(\Delta) \cdot opt_{LP_2}(Y_{i_0})}{2^{i_0}/(I+1)} \\
&\quad \big(\text{because the integrality gap for (4)} \\
&\quad \text{is } H(\Delta) \text{ and } |Y_{i_0}| \geq 2^{i_0}/(I+1)\big) \\
&\leq \frac{2^{i_0+1} H(\Delta) \cdot opt_{LP_1}}{2^{i_0}/(I+1)} \\
&\quad \text{(by Claim 2)} \\
&\leq 4 \log n H(\Delta) opt_{MinDSC} \\
&\quad \text{(by Lemma 1 and the definition of } I\text{)}.
\end{aligned} \tag{11}$$

The theorem is proved.

2.2. Algorithm for MinPSMC Using Our Algorithm for MinDSC

The algorithm for MinPSMC is presented in Algorithm 2. It iteratively picks an approximate densest subcollection of sets until at least k elements are fully covered. We use $\mathcal{S}_j$ to denote the approximate densest subcollection picked in the jth iteration and denote by $\mathcal{F}^{(j)} = \bigcup_{i=1}^{j} \mathcal{S}_i$. Every time an approximate densest subcollection $\mathcal{S}_j$ is picked, the instance is updated into a *residual instance* $(E^{(j)}, \mathcal{S}^{(j)}, c, r^{(j)}, k^{(j)})$ as follows: $\mathcal{S}^{(j)} = \mathcal{S} \setminus \mathcal{F}^{(j)}$ is the collection of remaining sets, $r_e^{(j)} = \max\{0, r_e - |\mathcal{F}_e^{(j)}|\}$ is the residual covering requirement of element e, $C^{(j)} = \mathcal{C}(\mathcal{F}^{(j)})$ is the set of elements fully covered after the jth iteration, $E^{(j)} = E \setminus C^{(j)}$ is the set of elements with positive residual covering requirement, $k^{(j)} = \max\{0, k - |C^{(j)}|\}$ is the total residual covering requirement. Suppose the while loop is executed g times; that is, g is the index with $|\mathcal{C}(\mathcal{F}^{(g-1)})| < k$ and $|\mathcal{C}(\mathcal{F}^{(g)})| \geq k$. One problem is that the last subcollection $\mathcal{S}_g$ picked by line 4 of the algorithm may fully cover more residual elements than needed, and thus, its cost might be too large although its density is small. To solve such a problem, the algorithm outputs the better solution of $\mathcal{F}^{(g)}$ (which is $\mathcal{F}^{(g-1)} \cup \mathcal{S}_g$) and $\mathcal{F}^{(g-1)} \cup \mathcal{H}$, where $\mathcal{H}$ consists of $k^{(g-1)} = k - |\mathcal{C}(\mathcal{F}^{(g-1)})|$ *cheapest bundles* of the $(g-1)$th residual instance. To be more concrete, $\mathcal{H}$ is computed by Algorithm 3, the idea of which is as follows. A *bundle* for element e is a collection consisting of exactly $r_e^{(g-1)}$ sets from $\mathcal{S}_e^{(g-1)}$. So the residual covering requirement of e is satisfied by choosing one of its bundles. For each element, we choose one of its cheapest bundles as its *representative bundle*. Then, k cheapest representative bundles fully covers at least k residual elements.

Algorithm 2 (Greedy Algorithm for MinPSMC)
Input: An instance $(E, \mathcal{S}, c, r, k)$ for MinPSMC.
Output: A subcollection of sets $\mathcal{F}$ fully cover at least k elements.

1: $j \leftarrow 0$, $\mathcal{F}^{(0)} \leftarrow \emptyset$, $\mathcal{S}^{(0)} \leftarrow \mathcal{S}$, $E^{(0)} \leftarrow E$, $C^{(0)} \leftarrow \emptyset$, $r^{(0)} \leftarrow r$, $k^{(0)} \leftarrow k$.
2: **while** $k^{(j)} > 0$ **do**
3: $j \leftarrow j+1$.
4: Compute subcollection $\mathcal{S}_j$ on MinDSC instance $(E^{(j-1)}, \mathcal{S}^{(j-1)}, c, r^{(j-1)})$.
5: Let $\mathcal{F}^{(j)} \leftarrow \mathcal{F}^{(j-1)} \cup \{\mathcal{S}_j\}$ and update the instance into a residual one with data $\mathcal{S}^{(j)}, C^{(j)}, r^{(j)}, E^{(j)}, k^{(j)}$ as described in the preceding paragraph.
6: **end while**
7: **if** $|\mathcal{C}(\mathcal{F}^{(j)})| = k$ **then**
8: $\mathcal{H} \leftarrow \mathcal{S}_j$.
9: **else** (in this case $|\mathcal{C}(\mathcal{F}^{(j)})| > k$)
10: $\mathcal{H} \leftarrow$ CheapestBundle$(E^{(j-1)}, \mathcal{S}^{(j-1)}, c, r^{(j-1)}, k^{(j-1)})$.
11: **end if**
12: Output $\mathcal{F} \leftarrow \arg\min\{c(\mathcal{F}^{(j)}), c(\mathcal{F}^{(j-1)} \cup \mathcal{H})\}$.

Algorithm 3 (Function CheapestBundle $(E, \mathcal{S}, c, r, k)$)

1: For each $e \in E$, let $\mathcal{H}(e) \leftarrow \arg\min\{c(\mathcal{R}) : \mathcal{R} \subseteq \mathcal{S}_e, |\mathcal{R}| = r_e)\}$.
2: Order elements of E as $e_1, e_2, \ldots$ such that $c(\mathcal{H}(e_1)) \le c(\mathcal{H}(e_2)) \le \cdots$.
3: Return $\mathcal{H} \leftarrow \bigcup_{i=1}^{k} \mathcal{H}(e_i)$.

Theorem 2. *Making use of an α-approximation algorithm for MinDSC in line 4 of the algorithm, Algorithm 2 has approximation ratio at most $\alpha \ln k + \sqrt{\alpha n}$.*

Proof. Suppose the while loop is executed g times and $\mathcal{S}_1, \ldots, \mathcal{S}_g$ are the approximate densest subcollections picked by the algorithm. Let OPT be an optimal solution to MinPSMC and denote $opt = c(OPT)$. We use $\mathcal{C}^{(j)}(\mathcal{R})$ to denote the number of elements fully covered by subcollection $\mathcal{R}$ with respect to the $(j-1)$th residual instance. For each $j = 1, 2, \ldots, g$, because $\mathcal{S}_j$ is an α-approximate densest subcollection with respect to the $(j-1)$th residual instance and $OPT \setminus \mathcal{F}^{(j-1)}$ fully covers at least $k^{(j-1)} = k - |\mathcal{C}(\mathcal{F}^{(j-1)})|$ residual elements, we have

$$\frac{c(\mathcal{S}_j)}{\left|\mathcal{C}^{(j)}(\mathcal{S}_j)\right|} \le \alpha \cdot \frac{c\left(OPT \setminus \mathcal{F}^{(j-1)}\right)}{\left|\mathcal{C}^{(j)}\left(OPT \setminus \mathcal{F}^{(j-1)}\right)\right|} \le \alpha \cdot \frac{opt}{k^{(j-1)}}. \tag{12}$$

Notice that $|\mathcal{C}^{(j)}(\mathcal{S}_j)| = k^{(j-1)} - k^{(j)}$ holds for $j = 1, \ldots, g-1$. Combining this with (12),

$$\sum_{j=1}^{g-1} c(\mathcal{S}_j) \le \alpha \sum_{j=1}^{g-1} \frac{k^{(j-1)} - k^{(j)}}{k^{(j-1)}} opt. \tag{13}$$

Making use of the following fact, for two integers a, b with $b \ge a$,

$$\frac{b-a}{b} = \sum_{i=1}^{b-a} \frac{1}{b} \le \frac{1}{b} + \frac{1}{b-1} + \cdots \frac{1}{a+1} = H(b) - H(a),$$

where $H(b) = \sum_{i=1}^{b} 1/i$ is the bth harmonic number, we have

$$\sum_{j=1}^{g-1} \frac{k^{(j-1)} - k^{(j)}}{k^{(j-1)}} = \sum_{j=1}^{g-1} \left(H\left(k^{(j-1)}\right) - H\left(k^{(j)}\right)\right) = H\left(k^{(0)}\right) - H\left(k^{(g-1)}\right).$$

Combining this with (13) and the observation that $k^{(0)} = k$ and $k^{(g-1)} \ge 1$, we have

$$c\left(\mathcal{F}^{(g-1)}\right) = \sum_{j=1}^{g-1} c(\mathcal{S}_j) \le \alpha\left(H\left(k^{(0)}\right) - H\left(k^{(g-1)}\right)\right) opt \le \alpha(H(k) - 1) opt. \tag{14}$$

For $j = g$, because $|\mathcal{C}(\mathcal{S}^{(g)})| \le n - |C^{(g-1)}|$, inequality (12) implies

$$c(\mathcal{S}_g) \le \alpha \cdot \frac{n - \left|C^{(g-1)}\right|}{k^{(g-1)}} opt. \tag{15}$$

Notice that every bundle chosen in Algorithm 3 has cost upper bounded by opt, and through line 10 of Algorithm 2, $k^{(g-1)}$ bundles are chosen. Hence,

$$c(\mathcal{H}) \le k^{(g-1)} opt. \tag{16}$$

Combining (15) and (16), we have

$$\min\{c(\mathcal{S}_g), c(\mathcal{H})\} \le \min\left\{\frac{\alpha\left(n - \left|C^{(g-1)}\right|\right)}{k^{(g-1)}}, k^{(g-1)}\right\} opt.$$

Notice that the maximum value for $\min\{\alpha(n - |C^{(g-1)}|)/k^{(g-1)}, k^{(g-1)}\}$ is achieved when $\alpha(n - |C^{(g-1)}|)/k^{(g-1)} = k^{(g-1)}$, that is, when $k^{(g-1)} = \sqrt{\alpha(n - |C^{(g-1)}|)}$, and the maximum value is $\sqrt{\alpha(n - |C^{(g-1)}|)} \le \sqrt{\alpha n}$. So

$$\min\{c(\mathcal{S}_g), c(\mathcal{H})\} \le \sqrt{\alpha n} \cdot opt. \tag{17}$$

Combining (14) with (17) and the well-known fact that $H(k) \le \ln k + 1$, we have

$$c(\mathcal{F}) = c\left(\mathcal{F}^{(g-1)}\right) + \min\{c(\mathcal{S}_g), c(\mathcal{H})\} \le \left(\alpha(H(k)-1) + \sqrt{\alpha n}\right)opt \le \left(\alpha \ln k + \sqrt{\alpha n}\right)opt.$$

The claimed ratio is proved.

As a consequence of Theorem 1, we have the following corollary.

Corollary 1. *MinPSMC admits a* $(4\log nH(\Delta)\ln k + 2\log n\sqrt{n})$*-approximation.*

3. A Bicriteria Algorithm for MinPSMC

In this section, we give a bicriteria algorithm for MinPSMC. For this purpose, we first study the MinDSC$^{\ge k}$ problem, which is a MinDSC problem with the chosen subcollection fully covering at least k elements. A bicriteria approximation algorithm for MinDSC$^{\ge k}$ is presented (in which the total covering requirement is violated only by a little), based on which a bicriteria approximation guarantee for MinPSMC is obtained. In particular, when $k = \Omega(n)$, our ratio is better than that in Shi et al. (2019).

3.1. LP Formulation for MinDSC$^{\ge k}$

The following is an integer program for MinDSC$^{\ge k}$:

$$\min \frac{\sum_{S\in\mathcal{S}} c_S x_S}{\sum_{e\in E} y_e}$$
$$\text{s.t.} \begin{cases} \sum_{\mathcal{Q}:\mathcal{Q}\in\Omega_e} l_{\mathcal{Q}} \ge y_e, & \forall e \in E, \\ x_S \ge \max_{\mathcal{Q}:S\in\mathcal{Q}\in\Omega} l_{\mathcal{Q}}, & \forall S \in \mathcal{S}, \\ \sum_{e\in E} y_e \ge k, & \\ x_S \in \{0,1\}, & \forall S\in\mathcal{S}, \\ y_e \in \{0,1\}, & \forall e \in E, \\ l_{\mathcal{Q}} \in \{0,1\}, & \forall \mathcal{Q}\in\Omega. \end{cases} \tag{18}$$

We may relax (18) into a family of linear programs $\widetilde{LP}_k$ as follows:

$$\min \sum_{S\in\mathcal{S}} c_S x_S$$
$$\text{s.t.} \begin{cases} \sum_{e\in E} y_e = 1 & \\ \sum_{\mathcal{Q}:\mathcal{Q}\in\Omega_e} l_{\mathcal{Q}} \ge y_e, & \forall e\in E, \\ x_S \ge \sum_{\mathcal{Q}:S\in\mathcal{Q}\in\Omega_e} l_{\mathcal{Q}}, & \forall S\in\mathcal{S} \text{ and } \forall e \in E, \\ x_S \ge 0, & \forall S\in\mathcal{S}, \\ y_e \ge 0, & \forall e\in E, \\ y_e \le 1/k, & \forall e \in E, \\ l_{\mathcal{Q}} \ge 0, & \forall \mathcal{Q}\in\Omega. \end{cases} \tag{19}$$

The next lemma shows that the *minimum* optimal value of the preceding family of linear programs lower bounds that of the MinDSC$^{\ge k}$ problem.

Lemma 3. *Denote by* $opt_{\widetilde{LP}_k}$ *the optimal value of* $\widetilde{LP}_k$ *and* $opt_{DSC^{\ge k}}$ *the optimal value for MinDSC*$^{\ge k}$*. Then,* $opt_{\widetilde{LP}_k} \le opt_{DSC^{\ge k}}$.

Proof. Suppose $\mathcal{F}^*$ is an optimal solution to Min$DSC^{\ge k}$ and $|\mathcal{C}(\mathcal{F}^*)| = \ell^*$. Then, $\ell^* \ge k$. Let $x_S = 1/\ell^*$ for $S\in\mathcal{F}^*$. For each element $e\in\mathcal{C}(\mathcal{F}^*)$, let $y_e = 1/\ell^*$ and choose *exactly one* r_e-cover set $\mathcal{Q}_e \subseteq \mathcal{F}^*$ and let $l_{\mathcal{Q}_e} = 1/\ell^*$. Set all the other variables to be zeros. It can be verified that (x,y,l) is a feasible solution to linear program $\widetilde{LP}_k$. Hence,

$$opt_{\widetilde{LP}_k} \le \sum_{S\in\mathcal{F}^*} c_S x_S = \frac{\sum_{S\in\mathcal{F}^*} c_S}{\ell^*} = opt_{DSC^{\ge k}}.$$

The lemma is proved.

Similar to the proof of Lemma 2, we obtain the following lemma.

Lemma 4. *Program (19) is polynomial-time solvable.*

3.2. Algorithm and Analysis

Our algorithm is presented in Algorithm 4.

Algorithm 4 (Bicriteria Algorithm for Min DSC$^{\ge k}$)

Input: A MinDSC$^{\ge k}$ instance $(E,\mathcal{S},c,r)$ with $|E| = n$ and a real number $0 < \varepsilon \le 1$.
Output: A subcollection $\mathcal{S}'$ that fully covers at least $(1-\frac{\varepsilon}{2\log n})k$ elements.

```
p ← ⌈log log(n^2)⌉.
Find a minimum optimal solution (x^f, y^f, l^f) to LP~_k.
Let Y_i = {e ∈ E : 2^{-(i+1)} < k y_e^f ≤ 2^{-i}} for 0 ≤ i ≤ I − 1
    and Y_I = {e ∈ E : k y_e^f ≤ 2^{-I}}, where I = 2⌊log n⌋ − 1.
    Denote Y'_i = Y_0 ∪ Y_1 ∪ ··· ∪ Y_i for i = 0, 1, ..., I.
Let i_0 be an index such that |Y_{i_0}| ≥ 2^{i_0} k/(I + 1).
if i_0 ≥ p then
    Find an approximate solution S' to MinSMC
        on instance (Y_{i_0}, S, c, r).
else
    Find an approximate solution F' to MinSMC
        on instance (Y'_p, S, c, r).
    if |Y'_p| ≥ (1 − ε/(2 log n))k then
        S' ← F'
    else
        For {Z_j = Y_j \ Y'_p}_{j=0}^{I}, let j_0 be an index such
            that |Z_{j_0}| ≥ 2^{j_0} k(1 − Σ_{e∈Y'_p} y_e^f)/(I + 1).
        Find an approximate solution F'' to
            MinSMC on instance (Z_{j_0}, S, c, r).
        S' ← F' ∪ F''
    end if
end if
Output S'.
```

To analyze the approximation ratio of the algorithm, we explore the relation between $\widetilde{LP}_k$ and linear program formulation (4) of MinSMC on instance $(Y, \mathcal{S}, c, r)$.

Theorem 3. *For $n \geq 32$, Algorithm 4 either outputs a solution fully covering at least $(1-\frac{\varepsilon}{2\log n})k$ elements with density at most $8(\log n)^2 H(\Delta)\cdot opt_{DSC^{\geq k}}$ or outputs a feasible solution to MinDSC$^{\geq k}$ with density at most $\frac{8(\log n)^2 H(\Delta)}{\varepsilon} opt_{DSC^{\geq k}}$.*

Proof. Similar to the proof of Theorem 1, we have the following two claims.

Claim 1. An index i_0 as in line 4 of Algorithm 4 exists, and $i_0 \leq I-1$.

Claim 2. For any index $i \leq I-1$, let $opt_{LP_2}(Y)$ be the optimal value of linear program (4) for MinSMC on instance $(Y, \mathcal{S}, c, r)$, where $Y \subseteq Y_i'$. Then, $opt_{LP_2}(Y) \leq 2^{i+1}k\cdot opt_{\widetilde{LP}_k}$.

In the case when $i_0 \geq p$, we have $|Y_{i_0}| \geq 2^p k/(I+1) \geq k$ by the choice of p. Hence, the subcollection $\mathcal{S}'$ calculated in line 6 of Algorithm 4 is a feasible solution to MinDSC$^{\geq k}$. Then, by a similar argument as in the proof of the last inequality of Theorem 1, combining claims 1 and 2 for $Y = Y_{i_0}$, Lemma 3, the fact that $\mathcal{S}'$ fully covers all the elements in Y_{i_0}, and the known integrality gap $H(\Delta)$ for (4), we have

$$\begin{aligned}\frac{c(\mathcal{S}')}{|\mathcal{C}(\mathcal{S}')|} &\leq \frac{c(\mathcal{S}')}{|Y_{i_0}|} \leq \frac{H(\Delta)\cdot opt_{LP_2}(Y_{i_0})}{2^{i_0}k/(I+1)} \\ &\leq \frac{2^{i_0+1}kH(\Delta)\cdot opt_{\widetilde{LP}_k}}{2^{i_0}k/(I+1)} \\ &\leq 4\log n H(\Delta) opt_{DSC^{\geq k}}.\end{aligned}$$

This finishes the proof for the first case.

In the following, suppose $i_0 < p$.

Claim 3. $den(\mathcal{F}') \leq 8(\log n)^2 H(\Delta) opt_{DSC^{\geq k}}$.

Proof. By claim 2, we have $opt_{LP_2}(Y_p') \leq 2^{p+1}k\cdot opt_{\widetilde{LP}_k}$. Because $i_0 < p$ implies $|Y_p'| \geq |Y_{i_0}| \geq 2^{i_0}k/(I+1)$, similar to the proof of claim 2, we have

$$\begin{aligned}\frac{c(\mathcal{F}')}{|\mathcal{C}(\mathcal{F}')|} &\leq \frac{c(\mathcal{F}')}{|Y_p'|} \leq \frac{H(\Delta)\cdot opt_{LP_2}\left(Y_p'\right)}{2^{i_0}k/(I+1)} \\ &\leq \frac{2^{p+1}kH(\Delta)\cdot opt_{\widetilde{LP}_k}}{2^{i_0}k/(I+1)} \\ &\leq 8\left(\log n\right)^2 H(\Delta) opt_{DSC^{\geq k}}.\end{aligned}$$

So, in the case when $\mathcal{S}'$ is determined by line 10 of Algorithm 4, $\mathcal{S}'$ is a bicriteria solution satisfying the claim of the theorem.

Finally, we consider the case when $\mathcal{S}'$ is determined in line 14 of Algorithm 4. Notice that, in this case, $|Y_p'| < (1-\frac{\varepsilon}{2\log n})k$.

Claim 4. An index j_0 as in line 12 of Algorithm 4 exists and $j_0 \leq I-1$.

Proof. Denote $E' = E \setminus Y_p'$. By $\sum_{e\in E'} ky_e^f = k(1-\sum_{e\in Y_p'} y_e^f)$, there exists an index j_0 such that $\sum_{e\in Z_{j_0}} ky_e^f \geq k(1-\sum_{e\in Y_p'} y_e^f)/(I+1)$. Because $ky_e^f \leq 2^{-j_0}$ for every $e \in Z_{j_0} \subseteq Y_{j_0}$, we have $|Z_{j_0}| \geq 2^{j_0}k(1-\sum_{e\in Y_p'} y_e^f)/(I+1)$. Because $|Y_p'| < (1-\frac{\varepsilon}{2\log n})k$, we have $|Y_p'| \leq k-1$. Combining this with $ky_e^f \leq 1$, we have

$$k\left(1-\sum_{e\in Y_p'} y_e^f\right) \geq k - \left|Y_p'\right| \geq 1. \quad (20)$$

So, if $j_0 = I$, then $n \geq |Z_{j_0}| \geq 2^I/(I+1) \geq 2^{(2\lfloor\log n\rfloor-1)}/2\lfloor\log n\rfloor$, which is impossible when $n \geq 32$. So $j_0 \leq I-1$.

Claim 5. The subcollection $\mathcal{S}'$ determined in line 14 of Algorithm 4 fully covers at least k elements and, thus, is a feasible solution to MinDSC$^{\geq k}$.

Proof. Because $Z_{j_0} \subseteq E \setminus Y_p'$ by line 12 of Algorithm 4, we have $j_0 \geq p+1$. Combining this with the definition of p in line 1 of Algorithm 4, we have $2^{j_0}/(I+1) \geq 2^{p+1}/(I+1) \geq 2$. Then, by the choice of j_0 in line 12 of Algorithm 4 and (20),

$$\left|Z_{j_0}\right| \geq \frac{2^{j_0}k\left(1-\sum_{e\in Y_p'} y_e^f\right)}{I+1} \geq k\left(1-\sum_{e\in Y_p'} y_e^f\right) \geq k - \left|Y_p'\right|. \quad (21)$$

Hence, $|\mathcal{C}(\mathcal{S}')| \geq |Y_p'| + |Z_{j_0}| \geq k$.

Claim 6. $den(\mathcal{F}'') \leq \frac{8}{\varepsilon}(\log n)^2 H(\Delta) opt_{DSC^{\geq k}}$.

Proof. By claim 2, $opt_{LP_2}(Z_{j_0}) \leq 2^{j_0+1}k opt_{\widetilde{LP}_k}$. Because $|Y_p'| < (1-\frac{\varepsilon}{2\log n})k$, we have $k(1-\sum_{e\in Y_p'} y_e^f) \geq k - |Y_p'| > \frac{\varepsilon k}{2\log n}$. By a similar argument as in the proof of claim 5,

$$\begin{aligned}\frac{c(\mathcal{F}'')}{|\mathcal{C}(\mathcal{F})|} \leq \frac{c(\mathcal{F}'')}{|Z_{j_0}|} &\leq \frac{H(\Delta)\cdot opt_{LP_2}(Z_{j_0})}{2^{j_0}k\left(1-\sum_{e\in Y_p'} y_e^f\right)/(I+1)} \\ &< \frac{2^{j_0+1}kH(\Delta)\cdot opt_{\widetilde{LP}_k}}{2^{j_0}\varepsilon k/\left(2\log n(I+1)\right)} \\ &\leq \frac{8}{\varepsilon}(\log n)^2 H(\Delta) opt_{DSC^{\geq k}}.\end{aligned}$$

By claim 5, the subcollection $\mathcal{S}'$ determined in line 14 of Algorithm 4 is a feasible solution to MinDSC$^{\geq k}$. Combining claim 3 with claim 6, $\mathcal{S}'$ has density

$$\frac{c(\mathcal{S}')}{|\mathcal{C}(\mathcal{S}')|} \leq \frac{c(\mathcal{F}') + c(\mathcal{F}'')}{|Y'_p| + |Z_{j_0}|} \leq \max\left\{\frac{c(\mathcal{F}')}{|Y'_p|}, \frac{c(\mathcal{F}'')}{|Z_{j_0}|}\right\}$$
$$\leq \frac{8}{\varepsilon}(\log n)^2 H(\Delta) opt_{DSC^{\geq k}}.$$

The theorem is proved.

Corollary 2. *Let opt be the optimal value of MinPSMC. For $n \geq 32$, Algorithm 4 outputs a bicriteria solution fully covering at least $(1 - \frac{\varepsilon}{2\log n})k$ elements with cost at most $\frac{8n}{k\varepsilon}(\log n)^2 H(\Delta) \cdot opt$. In particular, when $k = pn$, where $0 < p < 1$ is a constant, Algorithm 4 outputs a bicriteria solution fully covering at least $(1 - \frac{\varepsilon}{2\log n})k$ elements with cost at most $O(\frac{1}{\varepsilon}(\log n)^2 H(\Delta))opt$.*

Proof. Let OPT be an optimal solution to MinPSMC. Because OPT is a feasible solution of MinDSC$^{\geq k}$, we have

$$opt_{DSC^{\geq k}} \leq \frac{opt}{|\mathcal{C}(OPT)|} \leq \frac{opt}{k}.$$

Combining this with Theorem 3, Algorithm 4 either outputs a solution fully covering at least $(1 - \frac{\varepsilon}{2\log n})k$ elements with cost at most $\frac{8n}{k}(\log n)^2 H(\Delta) \cdot opt$ or outputs a feasible solution to MinPSMC with cost at most $\frac{8n(\log n)^2 H(\Delta)}{k\varepsilon} opt$. Thus, the corollary is proved.

4. Conclusion and Discussion

In this paper, we give two approximation algorithms for the minimum partial set multicover problem. The first algorithm achieves approximation ratio $4\log n H(\Delta)\ln k + 2\sqrt{n}\log n$. Unlike previous work, the ratio is independent of the maximum covering requirement $r_{\max}$, and, thus, can be applied, say, on the MinPPDS problem in which the maximum degree of the graph is not upper bounded by a constant. The second is a bicriteria algorithm, which performs better in the case when the total covering requirement $k = \Omega(n)$, that is, when $k = pn$, where $0 < p < 1$ is a constant, the solution fully covers at least $(1 - \frac{\varepsilon}{2\log n})k$ elements and achieves approximation guarantee $O(\frac{1}{\varepsilon}(\log n)^2 H(\delta))$, where δ is the maximum degree in the graph. The ratio is obtained by studying the MinDSC$^{\geq k}$ problem, which might be of interest by itself. Whether MinDSC$^{\geq k}$ admits a good approximation instead of a bicriteria one is an interesting question. Another question is whether MinPPDS admits a better approximation by exploring the graph structural properties of the problem.

Because Min$k$U is a special case of MinPSMC, our algorithm can also be applied to Min$k$U. In fact, by showing that the MinDSC problem corresponding to Min$k$U can be solved in polynomial time, we can obtain a $(\ln k + \sqrt{m})$-approximation for Min$k$U, where m is the number of hyperedges. However, this ratio is not as good as the best known ratio $O(m^{1/4+\varepsilon})$ for Min$k$U. It is interesting to ask whether the techniques in Chlamtáč et al. (2017) can be generalized to give a better approximation for MinPSMC.

References

Applebaum B, Barak B, Wigderson A (2010) Public-key cryptography from different assumptions. *Proc. 42nd ACM Sympos. Theory Comput.* (ACM, Cambridge, MA), 171–180.
Arora S, Barak B, Brunnermeier M, Ge R (2010) Computational complexity and information asymmetry in financial products. *1st Sympos. Innovations Comput. Sci. – ICS* (Tsinghua University Press, Beijing), 49–65.
Asahiro Y, Hassin R, Iwama K (2002) Complexity of finding dense subgraphs. *Discrete Appl. Math.* 121(1–3):15–26.
Bar-Yehuda R (2001) Using homogeneous weights for approximating the partial cover problem. *J. Algorithms* 39(2):137–144.
Bhaskara A, Charikar M, Chlamtáč E, Feige U, Vijayaraghavan A (2010) Detecting high log-densities: An *o* (*n* 1/4) approximation for densest *k*-subgraph. *Proc. 42nd ACM Sympos. Theory Comput.* (ACM), 201–210.
Chlamtáč E, Dinitz M, Krauthgamer R (2012) Everywhere-sparse spanners via dense subgraphs. *53rd Annual Sympos. Foundations Comput Sci.* (IEEE, Piscataway, NJ), 758–767.
Chlamtáč E, Dinitz M, Makarychev Y (2017) Minimizing the union: Tight approximations for small set bipartite vertex expansion. *Proc. 28th Annual ACM-SIAM Sympos. Discrete Algorithms* (SIAM, Philadelphia), 881–899.
Chlamtáč E, Dinitz M, Konrad C, Kortsarz G, Rabanca G (2018) The densest *k*-subhypergraph problem. *SIAM J. Discrete Math.* 32:1458–1477.
Dinh TN, Shen Y, Nguyen DT, Thai MT (2014) On the approximability of positive influence dominating set in social networks. *J. Combin. Optim.* 27:487–503.
Feige U, Peleg D, Kortsarz G (2001) The dense *k*-subgraph problem. *Algorithmica* 29:410–421.
Fujishige S (2005) *Submodular Functions and Optimization* (Elsevier, Amsterdam).
Gandhi R, Khuller S, Srinivasan A (2004) Approximation algorithms for partial covering problems. *J. Algorithms* 53(1):55–84.
Hajiaghayi MT, Jain K (2006) The prize-collecting generalized Steiner tree problem via a new approach of primal-dual schema. *SODA* (ACM-SIAM, Miami), 631–640.
Kearns MJ (1990) *The Computational Complexity of Machine Learning* (MIT Press, Cambridge, MA).
Korte B, Vygen J (2008) *Combinatorial Optimization Theory and Algorithms* (Springer, Berlin, Heidelberg).
Kortsarz G, Peleg D (1993) On choosing a dense subgraph. *Proc. 34th Annual Sympos. Foundations Comput. Sci.* (IEEE, Piscataway, NJ), 692–701.
Liu P, Huang X (2018) Approximation algorithm for partial set multicover vs. full set multicover. *Discrete Math. Algorithms Appl.* 10(2):1850026.
Louis A, Makarychev Y (2014) Approximation algorithms for hypergraph small set expansion and small set vertex expansion. Working paper, Georgia Tech, Atlanta.

Nutov Z (2010) Approximating Steiner networks with node-weights. *SIAM J. Comput.* 39(7):3001–3022.

Rajagopalan S, Vazirani VV (1998) Primal-dual RNC approximation algorithms for set cover and covering integer programs. *SIAM J. Comput.* 28(2):525–540.

Ran Y, Shi Y, Zhang Z (2017a) Local ratio method on partial set multicover. *J. Combin. Optim.* 34:302–313.

Ran Y, Shi Y, Tang C, Zhang Z (2020) A primal-dual algorithm for the minimum partial set multi-cover problem. *J. Combin. Optim.* 39:725–746.

Ran Y, Zhang Z, Du H, Zhu Y (2017b) Approximation algorithm for partial positive influence problem in social network. *J. Combin. Optim.* 33:791–802.

Shi Y, Ran Y, Zhang Z, Du D-Z (2020) A bicriteria algorithm for the minimum submodular cost partial set multi-cover problem. *Theoret. Comput. Sci.* 803:1–9.

Shi Y, Ran Y, Zhang Z, Willson J, Tong G, Du D-Z (2019) Approximation algorithm for the partial set multi-cover problem. *J. Global Optim.* 75:1133–1146.

Slavík P (1997) Improved performance of the greedy algorithm for partial cover. *Inform. Processing Lett.* 64(5):251–254.

Vazirani VV (2013) *Approximation Algorithms* (Springer Science & Business Media, Berlin, Heidelberg).

Wang F, Camacho E, Xu K (2009) Positive influence dominating set in online social networks. *Internat. Conf. Combin. Optim. Appl.* (Springer), 313–321.

Wang F, Du H, Camacho E, Xu K, Lee W, Shi Y, Shan S (2011) On positive influence dominating sets in social networks. *Theoretical Comput. Sci.* 412(3):265–269.

Zhang Z, Shi Y, Willson J, Du D-Z, Tong G (2017) Viral marketing with positive influence. *INFOCOM 2017-IEEE Conf. Comput. Comm.* (IEEE, Piscataway, NJ), 1–8.