# A Primal-Dual Approximation Algorithm for the Minimum Soft Capacitated Power Cover Problem

Li Guan[1], Han Dai[1], Xiaofei Liu[2*]

[1] School of Mathematics and Statistics, Yunnan University, Kunming, China.
[2] School of Information Science and Engineering, Yunnan University, Kunming, China.

* Corresponding author. Email: lxfjl2016@163.com
Manuscript submitted January 15, 2022; accepted March 8, 2022.
doi: 10.17706/ijapm.2022.12.2.10-21

**Abstract:** In this paper, we study the minimum soft capacitated power cover problem: Given a set $V$ of $n$ client points, a set $\mathcal{S}$ of $m$ server points on a plane. Each sensor $s$ can be arranged by set $p_s$ of power ( $p_s$ ) may contain the same power) and the covering range of sensor $s$ with any power $p \in P_s$ is a disk $d(s, p)$ of radius $r(s)$ satisfying $p = cr(s)^{\alpha}$. Where $c > 0$ and $\alpha \geq 1$ are two constants. Any disk center at sensor $s$ has a capacity $k_s$. The minimum soft capacitated power cover problem is to find a power set for each sensor denoted as $\{p_s\}_{s\in\mathcal{S}}$ such that each client point is assigned to one disk supported by $\{p_s\}_{s\in\mathcal{S}}$ satisfying that the number of client points assigned to $d(s, p)$ is at most $k_s$ for any $s \in \mathcal{S}$ and $p \in P_s$. The objective is to minimize the value of $\{p_s\}_{s\in\mathcal{S}}$, i.e. the total power $\sum_{s:s\in\mathcal{S}} \sum_{p:p\in P_s} p$. Our main result is to present a primal-dual $f$-approximation algorithm for the MSCPCP, where $f = max_{v\in V} |\{D \in \mathcal{D} : v \in V(D)\}|$ and $\mathcal{D}$ is a disk set related to $V$ and $\mathcal{S}$.

**Key words:** Approximation algorithm, primal-dual, soft capacitated power cover.

## 1. Introduction

The minimum set cover problem (MSCP) is classic non-deterministic polynomial-time hardness ($NP$-hard) problem in combinatorial optimization and approximation algorithms, which is defined as follow. Given a ground element set $E = \{1, \cdots, n\}$, and a collection $\mathcal{S}$ of sets defined over $E$, Each $S \in \mathcal{S}$ has a nonnegative cost $w(S) \geq 0$. The MSCP is to find a subset $C$ of $\mathcal{S}$ such that each element $i$ is covered by a set $S \in C$, i.e., $i \in S$. The objective is to minimize the total cost of $C$. The MSCP has been studied extensively in the literature, and the best approximation factor achievable for it is $O(\log n)$ [1]-[4].

The minimum capacitated set cover problem (MCSCP) is a generalization of the MSCP, in which each set $S \in \mathcal{S}$ has a capacity $k_S$ associated with it, where the set $S$ can cover at most $k_S$ elements. Generally, the MCSCP can be divided into two categories: soft capacities and hard capacities. In the case of soft capacities, an unbounded number of copies of each set $S$ is available; In the case of hard capacities, set $S$ has an upper bound of copies, defined as $u_S$, and $C$ is a feasible cover of the MCSCP, if $C$ contains at most $u_S$ copies for each $S \in \mathcal{S}$, and each copy set covers at most $k_S$ elements.

The minimum vertex cover problem (MVCP) is an important special case of the MCSCP, defined as follows. Given a graph $G = (V, E)$, and each vertex $v \in V$ has a cost $w_v$. The objective is to cover all the edges by picking a subset with minimum weight from $V$. The MVCP is $NP$-hard [5], and Khot and Regev [6] improved that the MVC cannot be approximated with in $2-\varepsilon$ for any $\varepsilon > \mathbf{0}$ under the unique game conjecture (UGC). Based on the LP-rounding, Hochbaum [7] presented a 2-approximation algorithm with running time of $O(n^3)$. Based on the primal-dual method, Bar-Yehuda and Even [8] proposed a linear-time 2-approximation algorithm.

The minimum capacitated vertex cover problem (MCVCP) was first introduced by Guha *et al*. [9], which is a generalization of the MVCP, where each vertex $v \in V$ has a capacity $k_v$. They considered the MCVCP with soft capacities, and presented a 2-approximation algorithm. Gandhi *et al*. [10] provided further results for the MCVCP with soft capacities. Bar-Yehuda *et al*. [11] considered the partial MCVCP with soft capacities, which is to find a vertex set that covers at least $k$ edges, and presented a 3-approximation. A tight approximation for the partial MCVCP with soft capacities was given by Mestre [12]. Chuzhoy and Naor [13] considered the MCVCP with hard capacities and presented a 3-approximation algorithm if $w_v = 1$ for each $v \in V$ and an $O(f)$-approximations algorithm when the vertices are weighted, respectively, which is improved by Gandhi *et al*. [14]. More related results for the MCVCP can be found in [15]-[18].

The minimum power cover problem (MPCP) is another important special case of the MSCP, which comes from some practical problems, such as wireless sensor networks [19] and sea measurement floating sensors [20]. In the MPCP, we are given a plane with a point set $V$ and a sensor set $S$ on it. Each sensor $s \in S$ can adjust its power, where the power $p$ of each sensor is determined by the radius $r(s)$ of the sensor and the relationship between the power and the radius is as follows:

$p = c \cdot r(s)^{\alpha}$, where $c > 0$ and $\alpha \geq 1$ are two constants. The objective is to minimize the total power across all sensors such that each point $v$ in $V$ is covered by some sensor, where a point $v$ is covered by a sensor $s$ if the distance from $v$ to $s$ is no more than $r(s)$. The MPCP is $NP$-hard [21] and the best approximation algorithm is the PTAS designed by Biló et. al. [22]. More related results for the MPCP can be found in [23]-[27].

In the real world, each sensor has a service upper bound and multiple sensors can be placed in a location to serve more client points. Therefore, we consider a new MPCP, called the minimum soft capacitated power cover problem (MSCPCP), which is generated the MPCP to soft capacity constraints. In this paper, firstly, by analyzing the properties of the optimal solution, we use a disk set $D$ to redefine the MSCPCP. Then, we present a primal-dual $f$-approximation algorithm for the MSCPCP, where $f = max_{v \in V} |\{D \in \mathcal{D} : v \in V(D)\}|$.

The rest of this paper is organized as follows. In Section 2, we describe the definition of the MSCPCP and some preliminaries. In Section 3, we present the primal-dual approximation algorithm. In section 4, we give a specific example to help understand our algorithm. In Section 5, we present a brief conclusion and possible directions for future research.

## 2. Preliminaries

The minimum soft capacitated power cover problem (MSCPCP) is defined as follows: Given a set $V$ of $n$ client points, a set $S$ of $m$ server points on a plane. Each sensor $s$ can be arranged by set $P_s$ of power ($P_s$ may contain the same power) and the covering range of sensor $s$ with any power $p \in P_s$ is a disk $d(s, p)$ of radius $r(s)$ satisfying

$$p = c \cdot r(s)^{\alpha}$$

where $c > 0$ and $\alpha \geq 1$ are two constants. If $v \in d(s, p)$, the client point $v$ can be assigned to disk $d(s, p)$. Any disk center at sensor $s$ has a capacity $k_s$. The MSCPCP is to find a power set for each sensor, denoted as $\{P_s\}_{s \in S}$, such that each client point is assigned to one disk supported by $p \in \{P_s\}_{s \in S}$ satisfying that the number of client points assigned to $d(s,p)$ is at most $k_s$ for any $s \in S$ and $p \in P_s$. The objective is to minimize the value of $\{P_s\}_{s \in S}$, i.e. the total power

$$\sum_{s:s \in S} \sum_{p:p \in P_s} p.$$

Let $\{P_s*\}_{s \in S}$ be an optimal assignment for the MSCPCP, for any sensor $s$ with $p* \in P_s*$, there is at least one client point $v \in V$ on the boundary of the disk $d(s, p*)$; Otherwise, we can reduce $p*$ to cover the same number of client points and find a power assignment with a smaller value. Therefore for each sensor $s$, there are at most $n$ disks with different radius in optimal assignment and at most $mn$ disks need to be considered. We use $D$ to denote the set of all these disks. For any disk $D \in \mathcal{D}$, $p(D)$, $c(D)$ and $r(D)$ denote the power, center and radius of the disk $D$, respectively. Since any disk center at sensor $s$ has a capacity $k_s$, for any disk $D \in \mathcal{D}$ with $c(D) = s$, let $k_D = k_s$ be the capacity of disk $D$.

The MSCPCP can be redefined as follows: Given a client point set $V$ and a disk set $D$ on the plane. Each disk $D \in \mathcal{D}$ has a power $p(D)$, a capacity $k_D$ and a corresponding point set $V(D) \subseteq V$, where only the client point in $V(D)$ can be assigned to $D$ and $D$ can assign at most $k_D$ client points. The MSCPCP is to find a capacity assignment function $x : D \to N_{\geq 0}$ such that there exists an assignment of client points satisfying that the number of client points assigned to each disk is at most $k_D x(D)$ and minimum the total power

$$\sum_{D:D \in \mathcal{D}} p_D \cdot x(D).$$

For each client point $v \in V$ and each disk $D \in \mathcal{D}$, we introduce a binary variable $y_{vD}$, where

$$y_{vD} = \begin{cases} 1, \text{ if } v \in V(D) \text{ and } v \text{ is assigned to } D, \\ 0, \text{ otherwise.} \end{cases}$$

The integer linear programming of the MSCPCP is defined as follows:

$$\begin{aligned} &\min \sum_{D:D \in \mathcal{D}} p_D \cdot x(D) \\ s.t. \quad &\sum_{D:v \in V(D)} y_{vD} \geq \mathbf{1},\ \forall v \in V, \\ &k_D x(D) - \sum_{v:v \in V(D)} y_{vD} \geq \mathbf{0},\ \forall D \in \mathcal{D}, \\ &x(D) \geq y_{vD},\ \forall v \in V(D) \text{ and } D \in \mathcal{D}, \\ &y_{vD} \in \{\mathbf{0}, \mathbf{1}\},\ \forall\ v \in V \text{ and } D \in \mathcal{D}, \\ &x(D) \in N_{\geq \mathbf{0}},\ \forall D \in \mathcal{D}. \end{aligned} \tag{1}$$

The first set of constraints guarantees that each client point $v \in V$ is assigned to some disk in $D \in \mathcal{D}$ with $v \in V(D)$; the second set of constraint guarantees that the number of client points assigned to any disk $D$ is no more than $k_D x(D)$; In fact, we do not really need the third set of constraints, however this constraint will play an important role in the relaxation, i.e., without this constraint there is a large integrality gap between the best fractional and integral solutions. Relaxing the integrality constraints, we get a linear programming as follows:

$$
\begin{aligned}
\min & \sum_{D:D\in\mathcal{D}} p_D \cdot x(D) \\
s.t. & \sum_{D:v\in V(D)} y_{vD} \geq \mathbf{1},\ \forall v\in V, \\
& k_D x(D) - \sum_{v:v\in V(D)} y_{vD} \geq \mathbf{0},\ \forall D\in\mathcal{D}, \\
& x(D) \geq y_{vD},\ \forall v\in V(D)\ and\ D\in\mathcal{D}, \\
& y_{vD} \geq \mathbf{0},\ \forall\ v\in V\ and\ D\in\mathcal{D}, \\
& x(D) \geq \mathbf{0},\ \forall D\in\mathcal{D}.
\end{aligned} \tag{2}
$$

For any an optimal solution of (2), we have $y_{vD} \leq \mathbf{1}$. Thus, we deleted the constraints $y_{vD} \leq \mathbf{1}$ from (2). The corresponding dual program is

$$
\begin{aligned}
\max & \sum_{v:v\in V} \eta_v \\
s.t. &\ k_D\beta_D + \sum_{v:v\in V(D)} \gamma_{vD} \leq p_D,\ \forall D\in\mathcal{D}, \\
& \beta_D + \gamma_{vD} \geq \eta_v, \forall v\in V(D)\ and\ D\in\mathcal{D}, \\
& \eta_v \geq \mathbf{0},\ \forall v\in V, \\
& \beta_D \geq \mathbf{0},\ \forall D\in\mathcal{D}, \\
& \gamma_{vD} \geq \mathbf{0},\ \forall v\in V(D)\ and\ D\in\mathcal{D}.
\end{aligned} \tag{3}
$$

## 3. Primal-Dual Algorithm

In the section, we present a primal-dual $f$-approximation algorithm for the MSCPCP, where $f = max_{v\in V} |\{D\in\mathcal{D}: v\in V\ (D)\}|$.

The main idea of the primal-dual algorithm can be described as follows. Initially, no client points are assigned and all disks are closed. As the algorithm runs, we select certain disks to open. When a disk $D'$ is opened, all unassigned client points in $V(D')$ are assigned to it. However, later on, if another disk $D$ with $V(D')\cap V(D)\neq\varnothing$ is opened, client points in $V(D')\cap V(D)$ that was previously assigned to disk $D'$ may get reassigned to disk $D$. In the end, the algorithm construct the capacity assignment function $x: D\to N_{\geq 0}$, where $x(D)=\left\{\left\lceil \frac{|A_D|}{k_D}\right\rceil\right\}$ and $A_D$ is the set of client points assigned to $D$.

Before introducing the detail implementation method of the algorithm, we need the following definitions. For an instance $(V,\mathcal{D};p_D;k_D)$ of the MSCPCP, we defined $\mathcal{D}_{high}$ and $\mathcal{D}_{low}$ to be the set of high and low capacitated disks, i.e.,

$$\mathcal{D}_{high} = \{D\in\mathcal{D} | V(D) > k_D\} \text{ and } \mathcal{D}_{low} = \{D\in\mathcal{D} | V(D) \leq k_D\}.$$

Initially, let $V$ be the unassigned client point set, and let $D$ be the closed disk set. We begin with a trivial dual feasible solution zero of (3), i.e., $(\boldsymbol{\eta},\boldsymbol{\beta},\boldsymbol{\gamma})$=0. Dual variables $\{\eta_v\}_{\{v\in V\}}$ simultaneously increase. To maintain dual feasibility

$$\beta_D + \gamma_{vD} \geq \eta_v, \forall v\in V(D)\ and\ D\in\mathcal{D},$$

As we increase $\eta_v$, we have to increase $\beta_D$ or $\gamma_{vD}$. For the disk $D$ in $\mathcal{D}_{high}$, we increase $\beta_D$; for the disk $D$ in $\mathcal{D}_{low}$, we increase $\{\gamma_{vD}\}_{\{v\in V(D)\cap V\}}$, where $V$ is the unassigned client point set. For the first set of constraints of (3),

$$k_D\beta_D + \sum_{v:v\in V(D)} \gamma_{vD} \le p_D,\ \forall D \in \mathcal{D}, \tag{4}$$

initially the left-hand side is 0 and the right-hand side is the power of the disk. While increasing the dual variables $\{\eta_v\}_{\{v\in V\}}$, we stop as soon as an inequality of disk $D'$ in (4) is met with equality. Open disk $D'$.

If $D' \in \mathcal{D}_{high}$, add $D'$ to the set $\mathcal{D}^{cand}$ of candidate disks which may be used many a time and temporarily assign all client points in $V(D')\cap V$ to disk $D'$, for convenience, this client point set is defined as $A_{D'}^{temp}$. Otherwise, for $D' \in \mathcal{D}_{low}$, let $x(D') = \mathbf{1}$. Note that some temporarily assigned client points in $\{A_{D'}^{temp}\}_{D\in\mathcal{D}^{cand}}$ may be reassigned to disk $D'$. We design a reassigned step to find the reassigned client point set $A_{D'}$ satisfying

$$\sum_{v:v\in A_{D'}} \eta_v = p_{D'}$$

And the main idea of reassigned step is introduced later, where we propose the detailed reassigned step in Algorithm 2. For each disk $D \in \mathcal{D}_{high}$, if $|(V(D)\cap V)\setminus V(D')| \le k_D$, then move $D$ from $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$ and let $A_D = (V(D)\cap V)\setminus V(D')$ and $A_D^{temp} = V(D)\cap V$ be the set of client points certainly and temporarily assigned to $D$, where $A_D = (V(D)\cap V)\setminus V(D')$ and $A_D^{temp} = V(D)\cap V$ is used in the reassigned step when $D$ is opened. Remove $D'$ from $\mathcal{D}$ .All dual variables $\{\eta_v\}_{\{v\in V(D')\cap V\}}$ and their corresponding dual variables $\{\eta_{vD}\}_{\{v\in V(D)\cap V\}}$ and $\beta_{D'}$ will no longer increase, and we have

$$\beta_{D'} + \gamma_{vD'} = \eta_v, \forall v \in V(D')\cap V\ .$$

Remove all client points in $V(D')\cap V$ from $V$. The process is iterated until $V$=∅. For each disk $D \in \mathcal{D}^{cand}$, set $A_D = A_D^{temp}$ and set $x(D) = \left\{\left\lceil \frac{|A_D|}{k_D} \right\rceil\right\}$. Output the capacity assignment function $x(\cdot)$ and an auxiliary assignment $\{A_D\}_{\{D\in\mathcal{D}\}}$, where some client points $v$ may be assigned to different disks by $\{A_D\}_{\{D\in\mathcal{D}\}}$. We propose the detailed the primal-dual algorithm in Algorithm 1 below. Then, we introduce the main idea of the reassigned step (Algorithm 2). When a disk $D' \in \mathcal{D}_{low}$ is opened in Algorithm 1, we need use Algorithm 2. If $V(D') \le k_{D'}$, then we set $A_{D'} := V(D')$; Otherwise, for $V(D') > k_{D'}$, let $A_{D'}$ and $A_{D'}^{temp}$ be the set of client points certainly and temporarily assigned to $D'$ when $D'$ moves form $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$. For each $D \in \mathcal{D}^{cand}$, reassign all client point in $A_D^{temp} \cap A_{D'}$ from $D$ to $D'$. Then, reassign $k_{D'} - |A_{D'}|$ client points in $A_{D'}^{temp} \setminus A_{D'}$ from some opened disk to $D'$, where we prefer to choose the client points in $A_D^{temp}$ for $D \in \mathcal{D}^{cand}$. In Lemma 3.2, we prove that $\sum_{v:v\in A_{D'}} \eta_v = p_{D'}$. In Theorem 3.3, we prove that the approximation factor of this primal-dual algorithm is $f$, where $f = max_{v\in V} |\{D \in \mathcal{D} : v \in V(D)|$. To help understand Algorithm 1 and 2, we give a specific example in Section 4.

LEMMA 3.1 ($\boldsymbol{\eta},\boldsymbol{\beta},\boldsymbol{\gamma}$) is a feasible solution of Dual program (3).

Proof. For any client point $v \in V$ and $D \in \{D \mid v \in V(D)\}$, if $V(D) \le k_D$ ,dual variables $\gamma_{vD}$ and $\eta_v$ simultaneously increase until $v$ is assigned to some disk, i.e.,

$$\beta_D + \gamma_{vD} = \gamma_{vD} \ge \eta_v,$$

where $\beta_D = \mathbf{0}$ for any low capacitated disk. Otherwise, for $V(D) > k_D$, **Case 1**, if $D \in \mathcal{D}_{high}$ when $v$ is assigned to some disk, dual variables $\beta_D$ and $\eta_v$ simultaneously increase until $v$ is assigned to some disk, i.e.,

$$\beta_D + \gamma_{vD} = \beta_D \geq \eta_v,$$

where $\gamma_{vD} = \mathbf{0}$ for any high capacitated disk. **Case 2**, if $D \in \mathcal{D}_{low}$ when $v$ is assigned to some opened disk, let $D'$ be the disk, where $D$ changes from $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$ when $D'$ is opened. It keeps the dual variable $\gamma_{vD} = \mathbf{0}$ and increases dual variables $\beta_D$ and $\eta_v$ simultaneously until $D'$ is opened; it keeps the dual variable $\beta_D$ and increases dual variables $\eta_{vD}$ and $\eta_v$ simultaneously until $v$ is assigned to some disk, i.e.,

$$\beta_D + \gamma_{vD} \geq \eta_v.$$

Thus, we have $\beta_D + \gamma_{vD} \geq \eta_v$ for any $v \in V$. This statement and inequality (4) imply that $(\boldsymbol{\eta}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ is a feasible solution of the dual program (3).

Algorithm 1. The Primal-Dual Algorithm

**Input:** An instance $(V, D; p_D; k_D)$ of the MSCPCP.

**Output:** A capacity assignment function $x: D \to N_{\geq \mathbf{0}}$ and an auxiliary assignment $\{A_D\}_{\{D \in \mathcal{D}\}}$.

1. Initially, set $x(D) = \beta_D = \gamma_{vD} = \mathbf{0}$ for $v \in V$ and $D \in \mathcal{D}$, and set $A_D = A_D^{temp} = \mathcal{D}^{cand} = \varnothing$ for $D \in \mathcal{D}$. Let $\mathcal{D}_{high}$ and $\mathcal{D}_{low}$ be the high and low capacitated disk sets defined as above.
2. **while** $V \neq \varnothing$ **do**
3. $\Delta_\beta := \min_{D: D \in D_{high}} \dfrac{p_D - k_D\beta_D - \sum_{v: v \in V(D)} \gamma_{vD}}{k_D}$ and

   $\Delta_\gamma := \min_{D: D \in D_{low}} \dfrac{p_D - k_D\beta_D - \sum_{v: v \in V(D) \setminus V} \gamma_{vD}}{|V(D) \cap V|}$.

   $\Delta := \min\{\Delta_\beta, \Delta_\gamma\}$. Let $D'$ be the minimum disk among $\mathcal{D}_{high} \cup \mathcal{D}_{low}$ with $\Delta$.
4. **for** $D \in \mathcal{D}_{low}$ **do**
5. $\gamma_{vD} := \Delta$ for each $v \in V(D) \cap V$.
6. **for** $D \in \mathcal{D}_{high}$ **do**
7. $\beta_D := \Delta$.
8. **if** $|(V(D) \cap V \setminus V(D')| \leq k_D$ and $D \neq D'$ **then**
9. $A_D^{temp} := V(D) \cap V, A_D := (V(D) \cap V) \setminus V(D')$,

   $\mathcal{D}_{low} := \mathcal{D}_{low} \cup \{D\}$ and $\mathcal{D}_{high} := \mathcal{D}_{high} \setminus \{D\}$.
10. **if** $\Delta_\gamma = \Delta$ **then**

    $(\{A_D^{temp}\}_{\{D \in \mathcal{D}^{cand}\}}, A_{D'}) \leftarrow \mathrm{Re}\,assign(V, V(D'), A_{D'}, A_{D'}^{temp};$ $k_{D'}; \mathcal{D}^{cand}; \{A_D^{temp}\}_{\{D \in \mathcal{D}^{cand}\}})$. $x(D') = \mathbf{1}, \mathcal{D}_{low} := \mathcal{D}_{low} \setminus \{D'\}$ and $V := V \setminus V(D')$.
11. **else**
12. $\mathcal{D}_{high} := \mathcal{D}_{high} \setminus \{D'\}$. $\mathcal{D}^{cand} := \mathcal{D}^{cand} \setminus \{D'\}$,
13. $A_{D'}^{temp} := V(D') \cap V$ and $V := V \setminus V(D')$.
14. For each $D \in \mathcal{D}^{cand}$, set $A_D := A_D^{temp}$ and $x(D) := \left\lceil \dfrac{|A_D|}{k_D} \right\rceil$.
15. Output function $x(\cdot)$ and auxiliary assignment $\{A_D\}_{\{D \in \mathcal{D}\}}$.

LEMMA3.2. For any $D \in \mathcal{D}^{cand}$ and $v \in A_D^{temp}$, we have $k_D\eta_v = p_D$; for any $D \notin \mathcal{D}^{cand}$ with $x(D) = \mathbf{1}$, we have

$$\sum_{v: v \in A_D} \eta_v = p_D.$$

Proof. For any $D \in \mathcal{D}^{cand}$, let $v$ be a client point in $A_D^{temp}$, dual variables $\gamma_{vD}$ keeps 0, and dual variables $\eta_v$ and $\beta_D$ increase until $D$ is opened. When $D$ is opened, dual variables $\beta_D$ and $\eta_v$ will no longer increase, i.e.

$$\eta_v = \beta_D \text{ and } \gamma_{vD} = \mathbf{0}.$$

According to the conditions of the disk opening, we have

$$p_D = k_D\beta_D + \sum\nolimits_{v:v\in V(D)} \gamma_{vD} = k_D\beta_D = k_D\eta_v .$$

For any $D \notin \mathcal{D}^{cand}$ with $x(D) = \mathbf{1}$, if $V(D) \le k_D$, we have

$$A_D = V(D)$$

By the reassigned step (Algorithm 2). For any $v \in V(D)$, dual variables $\beta_D$ keeps 0, and dual variables $\eta_v$ and $\gamma_{vD}$ increase until $v$ is assigned to some disk. Thus, we have $\beta_D = \mathbf{0}$, $\eta_v = \gamma_{vD}$ and

$$p(D) = k_D\beta_D + \sum\nolimits_{v:v\in V(D)} \gamma_{vD} = \sum\nolimits_{v:v\in V(D)} \gamma_{vD} = \sum\nolimits_{v:v\in A_D} \eta_v .$$

Algorithm 2. Reassign

**Input:** An unassigned client point set $V$; the corresponding client point set $V(D')$; the certainly and temporarily assigned client point set $A_{D'}$ and $A_{D'}^{temp}$ the capacity $k_{D'}$; the candidate disk set $\mathcal{D}^{cand}$ and its corresponding temporarily assigned client point set $\{A_D^{temp}\}_{\{D\in\mathcal{D}^{cand}\}}$.

**Output:** the reassigned client point set $\{A_D^{temp}\}_{\{D\in\mathcal{D}^{cand}\}}$ and $A_{D'}$ which is the set of client points assigned to $D'$.

1. **if** $A_{D'} = \varnothing$ **then**
2. $\quad A_{D'} := V(D')$.
3. $\quad$ **for** $D \in \mathcal{D}^{cand}$ **do**
4. $\qquad A_D^{temp} := A_D^{temp} \setminus A_{D'}$.
5. **else**
6. $\quad$ Set $R := A_{D'}^{temp} \setminus A_{D'}$ and $k := k_{D'} - |A_{D'}|$. Go to Step 15 if $k \ne \mathbf{0}$.
7. $\quad$ **for** $D \in \mathcal{D}^{cand}$ **do**
8. $\qquad A_D^{temp} := A_D^{temp} \setminus A_{D'}$
9. $\qquad$ **if** $R \cap A_D^{temp} \ne \varnothing$ **then**
10. $\qquad\quad$ **if** $|R \cap A_D^{temp}| < k$ **then**
11. $\qquad\qquad A_{D'} := A_{D'} \cup (R \cap A_D^{temp})$, $R := R \setminus A_D^{temp}$,
12. $\qquad\qquad k := k - |R \cap A_D^{temp}|$ and $A_D^{temp} := A_D^{temp} \setminus R$.
13. $\qquad\quad$ **else**
14. $\qquad\qquad$ Select a set $R' \subseteq R \cap A_D^{temp}$ satisfying $|R'| = k$. $A_{D'} := A_{D'} \cup R'$ and $A_D^{temp} := A_D^{temp} \setminus R'$. Go to Step 15**.**
15. Select a set $R' \subseteq R$ satisfying $|R'| = k$. $A_{D'} := A_{D'} \cup R'$ and go to Step 15.
16. Output $\{A_D^{temp}\}_{\{D\in\mathcal{D}^{cand}\}}$ and $A_{D'}$.

Otherwise, for $V(D) > k_D$, let $A'_D$ and $A_D^{temp}$ be the set of client points certainly and temporarily assigned to $D$ when $D$ moves form $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$. Thus, we have

$$A'_D \subseteq A_D \subseteq A_D^{temp} \text{ and } |A_D| = k_D.$$

By the reassigned step (Algorithm 2). For any $v \in A_D^{temp}$, dual variables $\gamma_{vD}$ keeps 0, and dual variables $\eta_v$ and $\beta_D$ increase until $D$ moves form $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$, and all client points in $A_D^{temp} \setminus A'_D$ are assigned to some open disk. Thus, we have that $\beta_D$ no longer increases and all client points in $V(D) \setminus A'_D$ is assigned after $D$ moves form $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$, i.e.,

$$\eta_v = \beta_D, \forall v \in A_D^{temp} \setminus A'_D \text{ and } \gamma_{vD} = \mathbf{0}, \forall v \in V(D) \setminus A'_D$$

Then, for any $v \in A'_D$, dual variables $\eta_v$ and $\gamma_{vD}$ increase until $v$ is assigned to some disk. Thus, we have

$$\eta_v = \beta_D + \gamma_{vD}, \forall v \in A'_D .$$

According to the conditions of the disk opening and $|A_D| = k_D$, we have

$$\begin{aligned} p_D &= k_D \beta_D + \sum_{v:v\in V(D)} \gamma_{vD} = k_D \beta_D + \sum_{v:v\in A'_D} \gamma_{vD} \\ &= |A_D \setminus A'_D| \beta_D + \sum_{v:v\in A'_D} (\beta_D + \gamma_{vD}) \\ &= \sum_{v:v\in A_D} \eta_v, \end{aligned}$$

where the third equality follows from $A_D \subseteq A_D^{temp}$ and $\eta_v = \beta_D$ for any $v \in A_D^{temp} \setminus A'_D$. Therefore, the Lemma holds.

Combining Lemma 3.1 and Lemma 3.2, we can obtain the following theorem.

THEOREM 3.3. Algorithm 1 achieves a worst-case guarantee of $f$ in polynomial time, where

$$f = max_{v\in V} |\{D \in \mathcal{D} : v \in V(D)| .$$

Proof. Consistent with the description above, $\{A_D\}_{\{D\in\mathcal{D}\}}$ and $x(\cdot)$ are the auxiliary assignment and assignment function generated by Algorithm 1, respectively. For any client point $v' \in V$, let $D'$ be the first disk to assign $v'$. If $D' \in \mathcal{D}_{low}$ when $v'$ is assigned to $D'$, then $v'$ is added to $A'_D$ and removes from the unassigned client point set. Meanwhile, for any $D \in \mathcal{D}^{cand}$ with $v' \cap A_D^{temp} \neq \varnothing$, remove $v'$ from $A_D^{temp}$ by the reassigned step (Algorithm 2). Thus, we have $v' \notin A_D$ for any $D \in \mathcal{D}^{cand}$, i.e.,

$$\begin{aligned} |\{D \in D \setminus \mathcal{D}^{cand} : v' \in A_D\}| &= |\{D \in \mathcal{D} : v' \in A_D\}| \\ &\le |\{D \in \mathcal{D} : v' \in V(D\}| \qquad (5) \\ &\le f, \end{aligned}$$

where $f = max_{v\in V} |\{D \in \mathcal{D} : v \in V(D)|$. The set consisting of such client points is defined as $V(\mathrm{D}low)$. Otherwise, for $D' \in \mathcal{D}_{high}$ when $v'$ is assigned to $D'$, this means $D' \in \mathcal{D}^{cand}$ and $v' \in V(\mathcal{D}_{high})$, where we define

$$V(\mathcal{D}_{high}) = V \setminus V(\mathcal{D}_{low}) .$$

Then $v'$ is added to $A_{D'}^{temp}$ and removes from the unassigned client point set. Thus, $v'$ will not be assigned to the other disk in $\mathcal{D}^{cand}$, i.e.,

$$v \notin A_D, \forall D \in D^{cand} \setminus \{D'\} .$$

**Case 1**. $v' \in A_{D'}$, then for any disk $D \in \mathcal{D} \setminus \mathcal{D}^{cand}$, $v' \cap A_D = \varnothing$, otherwise, $v'$ is removed from $A_{D'}^{temp}$ by the reassigned step (Algorithm 2). Since $v \notin A_D$ for any $D \in \mathcal{D}^{cand} \setminus \{D'\}$, we have

$$|\{D \in \mathcal{D} : v' \in A_D\}| = \mathbf{1}, \forall v' \in \bigcup_{D:D\in\mathcal{D}^{cand}} A_D \qquad (6)$$

**Case 2**. $v' \notin A_{D'}$, i.e. $v' \in A_{D'}^{temp} \setminus A_{D'}$. By the reassigned step (Algorithm 2), $v'$ is removed from $A_{D'}^{temp}$ when some disk in $\mathcal{D} \setminus \mathcal{D}^{cand}$ is opened. Since $A_D \subseteq V(D)$ for any $D \in \mathcal{D}$, we have

$$\begin{aligned} |\{D \in \mathcal{D} : v' \in A_D\}| &\le |\{D \in \mathcal{D} : v \in V(D)\}| - \mathbf{1} \\ &\le f - \mathbf{1}, \forall v' \in V(\mathcal{D}_{high}) \setminus \bigcup_{D:D\in\mathcal{D}^{cand}} A_D \qquad (7) \end{aligned}$$

where the first inequality follows from $v' \notin A_{D'}$ and $v' \in A_{D'}^{temp} \setminus A_{D'} \subseteq V(D')$.

We define

$$\begin{cases} \mathcal{D}_{=1}^{cand} = \{D \in \mathcal{D}^{cand} : x(D) = 1\}; \\ \mathcal{D}_{>1}^{cand} = \{D \in \mathcal{D}^{cand} : x(D) > 1\}; \end{cases}$$

where $\mathcal{D}^{cand}$ is the candidate disk set. For any $D' \in \mathcal{D}^{cand} = 1$, we have $|A_{D'}| \le k_{D'}$ and $|A_{D'}^{temp}| > k$. Thus, we have

$$x(D')p_{D'} = p_{D'} = k_{D'}\eta_v \le \sum_{v:v\in A_{D'}^{temp}} \eta_{v'} \tag{8}$$

where $v$ is a client point in $A_{D'}^{temp}$ and the second equality follows from Lemma 3.2. For any $D' \in \mathcal{D}_{>1}^{cand}$, we have $|A_{D'}| > k_{D'}$ and

$$\begin{aligned} x(D')p_{D'} &= \left\lceil \frac{|A_{D'}|}{k_{D'}} \right\rceil p_{D'} \le \frac{|A_{D'}| + k_{D'}}{k_{D'}} p_{D'} \\ &= \frac{|A_{D'}| + k_{D'}}{k_{D'}} k_{D'}\eta_v < 2|A_{D'}|\eta_v \\ &= \sum_{v:v\in A_{D'}} 2\eta_v \end{aligned} \tag{9}$$

where $v$ is a client point in $A_{D'}$ and the second equality follows from Lemma 3.2.

The total power of the capacity assignment function $x(\cdot)$ generated by Algorithm 1 is

$$\begin{aligned} \sum_{D:D\in\mathcal{D}} x(D)p_D &= \sum_{D:D\in D\setminus D^{cand}} x(D)p_D + \sum_{D:D\in\mathcal{D}^{cand}} x(D)p_D \\ &= \sum_{D:D\in D\setminus D^{cand} \text{and } x(D)=1} p_D + \sum_{D:D\in\mathcal{D}_{=1}^{cand}} p_D + \sum_{D:D\in\mathcal{D}_{>1}^{cand}} x(D)p_D \\ &\le \sum_{D:D\in D\setminus D^{cand} \text{and } x(D)=1} \sum_{v:v\in V(D)} \eta_v \\ &+ \sum_{D:D\in\mathcal{D}_{=1}^{cand}} \sum_{v:v\in A_D^{temp}} \eta_v + \sum_{D:D\in\mathcal{D}_{>1}^{cand}} \sum_{v:v\in A_D} 2\eta_v \\ &= \sum_{v:v\in V(D_{low})} |\{D\in D : v\in A_D\}|\eta_v \\ &+ \sum_{v:v\in V(D_{high})\setminus \bigcup_{D:D\in D^{cand}} A_D} |\{D\in D : v\in A_D\}|\eta_v \\ &+ \sum_{v:v\in \bigcup_{D:D\in D_{=1}^{cand}} A_D^{temp}} \eta_v + \sum_{v:v\in \bigcup_{D:D\in D_{>1}^{cand}} A_D} 2\eta_v \\ &\le \sum_{v:v\in V(D_{low})} f\eta_v + \sum_{v:v\in V(D_{high})\setminus \bigcup_{D:D\in D^{cand}} A_D} (f-1)\eta_v \\ &+ \sum_{v:v\in \bigcup_{D:D\in D_{=1}^{cand}} (A_D^{temp}\setminus A_D)} \eta_v + \sum_{v:v\in \bigcup_{D:D\in D^{cand}} A_D} 2\eta_v \\ &\le f\sum_{v\in V}\eta_v \le f \cdot OPT, \end{aligned}$$

where the first inequality follows from inequalities (8) and (9); the second inequality follows from inequalities (5), (6) and (7); the third in equality follows from $f \ge 2$; the last inequality follows from Lemma 3.1 and $OPT$ is the total power of the optimal capacity assignment function.

## 4. A Special Example for the MSCPCP

In this section, we give a specific example for the MSCPCP to help understand the primal-dual algorithm as follows: Given a client points set $V = \{v_i\}_{i=1,\cdots,9}$ and a set of disks $D = \{D_1, D_2, D_3\}$ in Fig. 1. a, where $k_{D_1} = 2, k_{D_2} = 5$ and $k_{D_3} = 3$; $p(D_1) = 2, p(D_2) = 6$ and $p(D_3) = 9$. In Algorithm 1, initially, $x(D) = \beta_D = \gamma_{vD} = 0$ for $v \in V$

and $D\in\mathcal{D}$, and $A_D = A_D^{temp} = \mathcal{D}^{cand} = \varnothing$ for $D\in\mathcal{D}$. $\mathcal{D}_{high}=\{D_1,D_3\}$ and $\mathcal{D}_{low}=.\ \{D_2\}$. For the first iteration, we have $\Delta_\beta=\mathbf{1}$ and $\Delta_\gamma=\frac{\mathbf{3}}{\mathbf{2}}$ and $D'=D_1$ is the minimum disk with $\Delta=\mathbf{1}$. By $D_2\in\mathcal{D}_{low}$ and $V(D_2)\cap V=\{v_3,v_4\}$, $\gamma_{v_3D_2}=\gamma_{v_4D_2}=\mathbf{1}$; since $D_1,D_3\in\mathcal{D}_{high}$, $\beta_{D_1}=\beta_{D_3}=\mathbf{1}$. Especially, $|(V(D_3)\cap V)\setminus V(D')|=\mathbf{3}\le k_{D_3}$, $A_{D_3}^{temp}=(V(D_3)\cap V)\setminus V(D')=\{v_6,v_8,v_9\}$. Then, $D_3$ is moved from $\mathcal{D}_{high}$ to $\mathcal{D}_{low}$ and $\mathcal{D}_{high}=\{D_1\}$. Since $\Delta_\beta=\Delta$, $D_1$ is removed from $\mathcal{D}_{high}$ and $\mathcal{D}_{high}=\varnothing$. Then, $D_1$ is added to the candidate disk set $\mathcal{D}^{cand}$; and $A_{D_1}^{temp}=\{v_1,v_2,v_3,v_4,v_5\}$; and the unassigned client point set $V=\{v_6,v_7,v_8,v_9\}$, in Fig. 1. b.

For the second iteration, we have $\Delta_\beta=\infty$ by $\mathcal{D}_{high}=\varnothing$ and $\Delta_\gamma=\mathbf{2}$ and $D'=D_2$ is the minimum disk with $\Delta=\mathbf{2}$. Since $D_2$, $D_3\in\mathcal{D}_{low}$, $V(D_2)\cap V=\{v_6,v_7\}$ and $V(D_3)\cap V=\{v_6\}$, $\gamma_{v_6D_2}=\ \gamma_{v_7D_2}=\gamma_{v_6D_3}=\mathbf{2}$. Since $\Delta_\gamma=\Delta$, we use the reassigned step (Algorithm2). Since $A_{D_2}=\varnothing$, $D^{cand}=\{D_1\}$ and $A_{D_1}^{temp}=\{v_1,v_2,\ v_3,v_4,v_5\}$ are input, Algorithm 2 outputs $A_{D_2}=V(D_2)=\{v_3,v_4,\ \ v_6,v_7\}$ and $A_{D_1}^{temp}=A_{D_1}^{temp}\setminus A_{D_2}=\{v_1,v_2,v_5\}$. Then, $x(D_2)=\mathbf{1}$, $\mathcal{D}_{low}\ =\{D_3\}$ and the unassigned client point set $V=\{v_8,v_9\}$, in Fig. 1. c. For the third iteration, we have $\Delta_\beta=\infty$ by $\mathcal{D}_{high}=\varnothing$ and $\Delta_\gamma=\mathbf{3}$ and $D'=D_3$ is the minimum disk with $\Delta=\mathbf{3}$. Since $D_3\in\mathcal{D}_{low}$, $\gamma_{v_8D_3}=\gamma_{v_9D_3}=\mathbf{3}$. Since $\Delta_\gamma=\Delta$, we use the reassigned step (Algorithm 2). Since $A_{D_3}=\{v_6,v_8,v_9\}\neq\varnothing$ and $A_{D_1}^{temp}=\{v_1,v_2,v_5\}$,we have $k=\mathbf{0}$ and Algorithm 2 outputs $A_{D_3}=\{v_6,v_8,v_9\}$ and $A_{D_1}^{temp}=\{v_1,v_2,v_5\}$. Then, $x(D_3)=\mathbf{1}$, $\mathcal{D}_{low}\ =\varnothing$ and the unassigned client point set $V=\varnothing$. This means, the iteration stops, in Fig. 1. d. Since $\mathcal{D}^{cand}=\{D_1\}$ and $A_{D_1}^{temp}=\{v_1,v_2,v_5\}$, we have

$$A_{D_1}=\{v_1,v_2,v_5\}\ \text{and}\ x(D_1)=\left\lceil\frac{|A_{D_1}|}{k_{D_1}}\right\rceil=\mathbf{2}.$$

The primal-dual algorithm outputs $x(D_1)=\mathbf{2}$, $x(D_2)=\mathbf{1}$ and $x(D_3)=\mathbf{1}$; $A_{D_1}=\{v_1,v_2,v_5\}$, $A_{D_2}=\ \{v_3,v_4,v_6,v_7\}$ and $A_{D_3}=\{v_6,v_8,v_9\}$. The value of $x(\cdot)$ is 19. It is obvious that the optimal assignment function is $x^*(D_1)=\mathbf{1}$, $x^*(D_2)=\mathbf{1}$ and $x^*(D_3)=\mathbf{1}$; $A^*_{D_1}=\{v_1,v_2\}$, $A^*_{D_2}=\{v_3,v_4,v_6,v_7\}$ and $A^*_{D_3}=\ \{v_5,v_8,v_9\}$. The value of $x^*(\cdot)$ is 17.

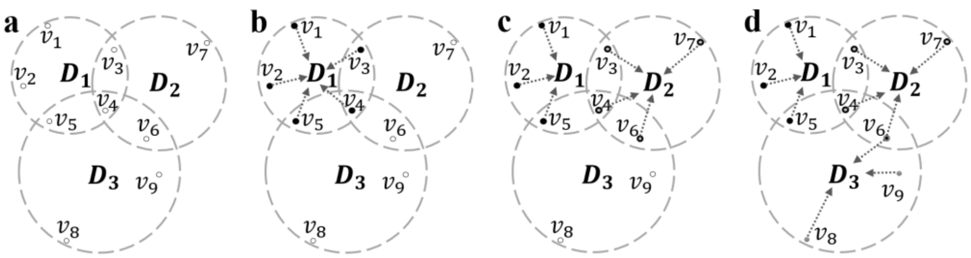



Fig. 1. A specific example for the MSCPCP.

## 5. Conclusion

In this paper, we introduce minimum soft capacitated power cover problem (MSCPCP), which is generated the MPCP to soft capacity constraints. We propose a primal-dual $f$-approximation algorithm for the MSCPCP, where $f=max_{v\in V}|\{D\in\mathcal{D}:v\in V(D)|$. The minimum power multi-cover problem (MPMP) is a generation of the MPCP, in which every client points $v$ has a covering requirement $cr_v$. The goal of the MPMC is to select a disk set such that each point $v$ is covered at least $cr_v$ times. Thus, the minimum soft capacitated power multi-cover problem (MSCPMP), which can be viewed as a generalization of the MSCPCP, deserves to be explored. It is possible to design an approximation algorithm with an approximation ratio of $f$, but it is a challenge.

## Conflict of Interest

The authors declare that they have no known competing financial interests.

## Autor Contributions

Li Guan put forward the problems we want to study and the research ideas of the problems. Han Dai mainly gives the idea and relevant proof of the algorithm, and is responsible for drafting the paper. Xiaofei Liu is mainly responsible for reviewing and revising the paper, giving a specific example, and is responsible for the final version of the paper.

## References

[1] Chvatal, V. (1979). A greedy heuristic for the set-covering problem. *Mathematics of Operations Research, 4*, 233-235.

[2] Feige, U. (1999). A threshold of $\ln n$ for approximating set cover. *Journal of the ACM, 45*, 314-318.

[3] Johnson, D. (1974). Approximation algorithms for combinatorial problem. *Journal of Computer and System Sciences, 9(3)*, 256-278.

[4] Vazirani, V. V. (2001). *Approximation algorithms.* Berlin Heidelberg / New York: Springer.

[5] Karp, R. M. (1972). Reducibility among combinatorial problems. In: *Complexity of Computer Computations* (pp. 85-103). Springer.

[6] Khot, S., & Regev, O. (2008). Vertex cover might be hard to approximate to within $2-\varepsilon$. *Journal of Computer and System Sciences, 74(3)*, 335-349.

[7] Hochbaum, D. S. (1982). Approximation algorithms for the set covering and vertex cover problems. *SIAM Journal on Computing, 11(3)*, 555-556.

[8] Bar-Yehuda, R., & Even, S. (1981). A linear time approximation algorithm for the weighted vertex cover problem. *Journal of Algorithms, 2*, 198-203.

[9] Guha, S., Hassin, R., Khuller, S., & Or, E. (2003). Capacitated vertex covering. *Journal of Algorithms, 48(1)*, 257-270.

[10] Gandhi, R., Halperin, E., Khuller, S., Kortsarz, G., & Srinivasan, A. (2006). An improved approximation algorithm for vertex cover with hard capacities. *Journal of Computer and System Sciences, 72(1)*, 16-33.

[11] Bar-Yehuda, R., Flysher, G., Mestre, J., & Rawitz, D. (2010). Approximation of partial capacitated vertex cover. *SIAM Journal on Discrete Mathematics, 24(4)*, 1441-1469.

[12] Mestre, J. (2009). A primal-dual approximation algorithm for partial vertex cover: Making educated guesses. *Algorithmica, 55(1)*, 227-239.

[13] Chuzhoy, J., & Naor, J. (2006). Covering problems with hard capacities. *SIAM Journal on Computing, 36(2)*, 498-515.

[14] Gandhi, R., Halperin, E., Khuller, S., Kortsarz, G., & Srinivasan, A. (2006). An improved approximation algorithm for vertex cover with hard capacities. *Journal of Computer and System Sciences, 72*, 16-33.

[15] Cheung, W., Goemans, M., & Wong, S. (2014). Improved algorithms for vertex cover with hard capacities on multi graphs and hypergraphs. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms* (pp. 2626-2637).

[16] Kao, M. (2017). Iterative partial rounding for vertex cover with hard capacities. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms* (pp. 2638-2653).

[17] Saha, B., & Khuller, S. (2012). Set cover revisited: hypergraph cover with hard capacities. *Proceedings of the International Colloquium on Automata, Languages and Programming* (pp. 762-773).

[18] Wong, S. (2017). Tight algorithms for vertex cover with hard capacities on multigraphs and hypergraphs. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms* (pp. 2626-2637).

[19] Xing, G., Lu, C., Zhang, Y., Huang, Q., & Pless R. (2005). Minimum power configuration in wireless sensor networks. *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing* (pp. 390-401).

[20] Li, M., Yang, Z., & Liu, Y. (2013). Sea depth measurement with restricted floating sensors. *ACM Transactions on Embedded Computing Systems, 13(1)*, 1-21.

[21] Alt, H., Arkin, E. M., Brönnimann, H., Erickson, J., Fekete, S. P., Knauer, C., Lenchner, J., Mitchell, J. S. B., & Whittlesey, K. (2006). Minimum-cost coverage of point sets by disks. *Proceedings of the ACM Symposium on Computational Geometry* (pp. 449-458).

[22] Biló, V., Caragiannis, I., Kaklamanis, C., & Kanellopoulos, P. (2005). Geometric clustering to minimize the sum of cluster sizes. *Proceedings of the European Symposium on Algorithm* (pp. 460-471).

[23] Bhowmick, S., Inamdar, T., & Varadarajan, K. (2017). On metric multi-covering problems. Retrieved from http://arxiv.org/abs/1602.04152

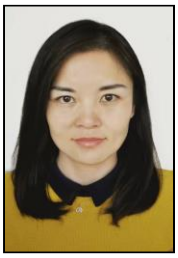
[24] Bhowmick, S., Varadarajan, K., & Xue, S. (2015). A constant-factor approximation for multi-covering with disks. *Journal of Computational Geometry, 6(1)*, 220-234.

[25] Huang, Z., Feng, Q., Wang, J., & Xu, J. (2021). PTAS for minimum cost multi-covering with disks. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms* (pp. 840-859).

[26] Liu, X., Li, W., & Xie, R. (2021). A primal-dual approximation algorithm for the *k*-prize-collecting minimum power cover problem. *Optimization Letters.*

[27] Shi, Y., Ran, Y., Zhang, Z., Willson, J., Tong, G., & Du, D. (2019). Approximation algorithm for the partial set multi-cover problem. *Journal of Global Optimization, 75*, 1133-1146.


Copyright © 2022 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).



**Li Guan** is a PhD graduate from Yunnan University and currently is a lecturer at the School of Mathematics and Statistics of Yunnan University. Her main research interests are combinatorial optimization and algorithmic game theory.

**Han Dai** is currently a master student in the School of Mathematics and Statistics of Yunnan University. Her main research interest is combinatorial optimization.

**Xiaofei Liu** received PhD degree in operational research and cybernetics from Yunnan University in 2018. He was a postdoctor at Peking University. He is currently a lecturer with Yunnan University. His research interests include combinational optimization, set function, power cover, load balance and edge-cloud computing.