On Solving General Planar Maximum Coverage Location Problem with Partial Coverage

Manish Bansal

Department of Industrial and Systems Engineering

Virginia Tech, Blacksburg, VA 24060

Email: bansal@vt.edu

Abstract. In this paper, we introduce a new generalization of the classical planar maximum coverage location problem (PMCLP) in which demand zones and service zone of each facility are represented by two-dimensional spatial objects such as circles, polygons, etc. In addition, we allow *partial coverage* in the foregoing generalization, i.e., covering only part of a demand zone is allowed and the coverage accrued in the objective function as a result of this is proportional to the demand of the covered part only. We denote this problem by PMCLP-PC, and present a greedy algorithm and a pseudo-greedy algorithm for it. We showcase that the solution value corresponding to the greedy (or pseudo-greedy) solution is within a factor of $1 - 1/e$ (or $1 - 1/e^\eta$) of the optimal solution value where e is the base of natural logarithm and $\eta \leq 1$. We also discuss about applications of the PMCLP-PC beyond location analytics.

Keywords. maximum coverage location problem; greedy approach; pseudo-greedy algorithm; partial coverage; facility location

1. Introduction

Over the years, many classes of problems related to locating service facilities to cover the demand for service have been studied; refer to [8, 9, 13, 14, 29] for extensive reviews. The maximum coverage location problem (MCLP) is one of the well known facility location problems in which p service facilities with known service range are to be located such that the total covered demand is maximized [10]. The classical forms of MCLP and several succeeding studies consider a finite set of pre-specified candidate positions for the facilities; whereas its generalization, referred to as the planar MCLP, allows the facilities to be located anywhere in the continuous plane [5, 20, 34]. These problems have received considerable attention in the literature due to their widespread applicability ranging from locating emergency healthcare centers to fire fighting stations, and from policy making through geographical informative systems (GIS) to solving clustering problems which themselves find applications in data mining, machine learning, and bioinformatics [4, 15, 30]. Despite that in most of the literature on the (planar) MCLP, point or aggregated representation of demand has been used, which is obtained by dividing a region into demand zones and aggregating the demand

1

of each zone at a single representative point (e.g. its centroid). As a result, the concept of coverage in this representation is assumed to be "binary", i.e., the demand zone is assumed to be either totally covered (if its representative point is covered by a service zone) or not covered at all (if its representative point is not covered by any service zone). This type of coverage is referred to as "binary coverage." Furthermore, even when demand in the (planar) MCLP is represented by line segments or polygons in [26, 27, 33], binary coverage is still an assumption, i.e., a linear/polygonal demand zone is either completely covered or not covered at all.

There is a substantial body of research demonstrating the impacts of the approximation errors, caused by the binary coverage assumption, on the stability of location model solutions, and therefore, researchers have been consistently calling for better approaches to represent demand [1, 3, 11, 12, 17, 21, 22, 24, 25, 26, 27, 28, 32, 33, 35]. These analysis showcase that the modeling solutions are sensitive to how demand is represented in the (planar) MCLP [25], and there are evidences that the point representation introduces unintended measurement and interpretation errors [11, 12, 33] which lead to significant gaps in actual coverage. Note that, assuming the range of coverage of a facility is d, the service zone of the facility, using Euclidean distance, is a circle with a radius of d centered at the facility, and using the rectilinear distance, is a diamond (a square rotated 45 degrees) with a diagonal of $2d$ centered at the facility. Church [6, 7] and Murray [23] accentuated that in the true optimal solution to a real world application, service zone of a facility and demand zones are represented by different spatial objects in which the demand zones may only be partially covered/served by the service facilities. Additionally, in current digital environment of enhanced computing, the understanding of spatiotemporal patterns of data for many policy and planning settings has improvised drastically, thereby providing more accurate abstraction of geographic space. Moreover, researchers have been raising the following questions: "Is it necessary to provide total coverage to an entire region? Would it be reasonable to consider partial coverage of spatial entities? How to best spatially represent the region in order to carry out coverage analysis?" [24].
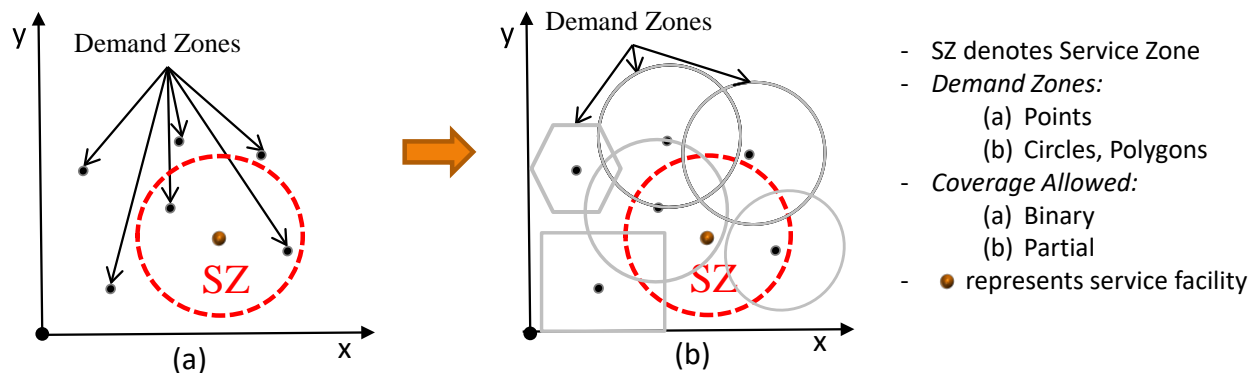

Figure 1: (a) Classical Planar MCLP [5]; (b) Planar MCLP with partial coverage (PMCLP-PC). Observe that because of the binary coverage in 1(a), only two demand zones are completely covered by the service zones, whereas others are not at all covered. This causes errors because in reality, as shown in 1(b), five demand zones (represented by spatial objects) are partially covered by the service zones.

Though attempts have been made to address the last (or third) question. To our knowledge, except [2, 31], no other study has tackled the (planar) MCLP problem when "partial coverage" is allowed in its true sense, i.e., when covering only part of a demand zone is allowed and the coverage accrued in the objective function as a result of this is proportional to the demand of the covered part only. Bansal and Kianfar [2] and Song et al. [31] developed exact algorithms for the planar MCLP where partial coverage is allowed, service and demand zones are defined by axis-parallel rectangles, and $p \geq 1$ (multiple facilities) and $p = 1$ (single facility), respectively. In this paper, we introduce a new generalization of the planar MCLP where partial coverage is allowed, and service and demand zones can be represented by any two-dimensional spatial object such as line segment, circle, polygon, etc. We denote this generalization of the planar MCLP with partial coverage by PMCLP-PC, and a special case of the PMCLP-PC with axis-parallel rectangular service and demand zones by PMCLP-PCR (studied in [2]).

1.1 Challenges in solving general PMCLP-PC

The motivation for the binary coverage assumption in the (planar) MCLP (where demand is represented by points, line segments, or polygons) is to make the problem manageable by readily formulating them as a linear binary integer program (LBIP). This is easy to do for the MCLP because of the discrete nature of candidate locations for service facilities (as per the definition). Moreover, even in studies considering a planar setting, i.e., allowing the facilities to be located anywhere in the continuous plane, coverage is still assumed to be binary [27] because this helps to show that a finite number of potential facility locations, called the circle intersection point set (CIPS) [5] and polygon intersection point set (PIPS) [27], exist which contain an optimal solution to this problem. Thereby resulting in the following well-known LBIP for the (planar) MCLP:

$$\max \left\{ \sum_i v_i x_i : \sum_j a_{ij} y_j \geq x_i \text{ for } i = 1, \dots, n, \sum_j y_j = p, x_i, y_j \in \{0, 1\} \right\}, \qquad (1)$$

where $x_i = 1$ if demand zone i is covered, $y_j = 1$ if a service facility is sited at the candidate/PIP/CIP point j, v_i is the given total demand of demand zone i, and a_{ij} is the given binary value which is 1 if demand zone i is covered by locating a facility at candidate/CIP/PIP point j. In contrast, the use of LBIP to solve the PMCLP-PC is not straightforward or rather not even feasible. We suggest readers to see [2] for more details on challenges in solving PMCLP-PC and PMCLP-PCR.

1.2 Organization and contribution of this paper

In Section 2, we formally define the PMCLP-PC, and discuss about a few of its applications in the field of GIS for disaster management and telerobotics. We present a greedy algorithm and a pseudo-greedy algorithm for solving the PMCLP-PC (Section 3), and showcase that the solution
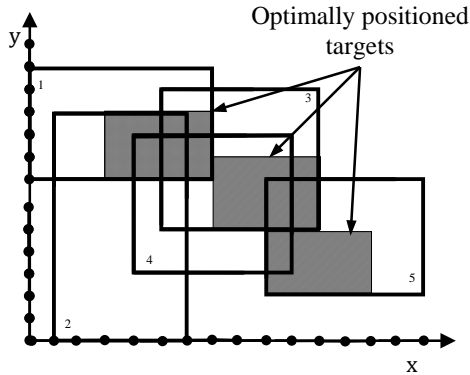
value corresponding to the greedy solution is within a factor of $1 - 1/e$ of the optimal solution value where e is the base of natural logarithm. It is important to note that the greedy (or pseudo-greedy) approach utilizes an exact (or an η-approximate where $\eta \leq 1$) algorithm for the PMCLP-PC with $p = 1$ (single facility), thereby providing approximate solutions for the general PMCLP-PC. Till date no such exact or η-approximate algorithm is known, except for PMCLP-PCR with $p = 1$ [2, 31]. Nevertheless, whenever an exact or η-approximate algorithm will be developed in future for the single facility PMCLP-PC or its special cases, it can be automatically embedded within the greedy (or pseudo-greedy) algorithm presented in this paper to provide solutions for the multiple (p) facility version of the corresponding problem. This demonstrates the significance of these two algorithms. More importantly, as we will prove in this paper, for a given $p \geq 1$, the greedy algorithm and the pseudo-greedy algorithm will provide solutions whose values will be at least $1 - [(p - 1)/p]^p$ and $1 - [(p - \eta)/p]^p$, respectively, times the optimal value for the PMCLP-PC instance.

2. Planar Maximum Coverage Location Problem with Partial Coverage: Definition and Applications

In this section, we formally define the PMCLP-PC as a new class of combinatorial optimization problem which also generalizes the maximum p-coverage problem, and discuss about its applications beyond facility location.

2.1 Problem Definition

We define the PMCLP-PC as follows. Let $\mathcal{R} = \{r_i, i = 1, \ldots, n\}$ be a set of n spatial objects, referred to as *requests* (or demand zones), on the two-dimensional plane such that dimensions, shape, and location of each request are known. Also, there is a weight (or demand) rate $w_i \in \mathbb{R}_+$ associated with each request r_i, i.e. weight per length/area-unit of r_i, $i \in \{1, \ldots, n\}$. Now, consider another set of geometric objects that provide coverage, referred to as *targets* (or service zones), and denote it by $\mathcal{T} := \{t_j, j = 1, \ldots, p\}$. We assume that the dimensions and shape of each target are known but their locations are not known. For $j = 1, \ldots, p$, we identify the location of t_j by the coordinates of either center or a corner (if exists) of the target and denote it by $\left(x_{t_j}, y_{t_j}\right) \in \mathbb{R}^2$. In addition, we denote the collective set of positions of $t_1, \ldots, t_p$ by $(x_t, y_t) = (x_{t_1}, \ldots, x_{t_p}, y_{t_1}, \ldots, y_{t_p}) \in \mathbb{R}^{2p}$, and define function $f_i(x_t, y_t) : \mathbb{R}^{2p} \to \mathbb{R}_+$ that returns the covered weight of r_i if targets are positioned at (x_t, y_t). More specifically, $f_i(x_t, y_t) = w_i A\left(r_i \cap \left(\cup_{j=1}^p t_j\right)\right)$ where function $A(.)$ returns the length/area of its argument, and $\cup$ and $\cap$ denote union and intersection operations on spatial objects. Therefore, the total weight covered by targets positioned at (x_t, y_t) is given by $f(x_t, y_t) = \sum_{i=1}^n f_i(x_t, y_t)$. The PMCLP-PC is the problem of finding the location of targets (or service facilities), i.e., (x_t, y_t), for which the total covered weight (or demand), i.e. $f(x_t, y_t)$, is maximized.

4

Request r_i	Lower-left corner (x_{r_i}, y_{r_i})	Length l_{r_i}	Width w_{r_i}	Weight rate w_i
1	(0,7)	5	7	4
2	(1,0)	10	5	6
3	(5,5)	6	6	7
4	(4,3)	6	6	3
5	(9,2)	5	6	10

Details of targets		Optimal positions (x_{t_j}, y_{t_j})
Length (l_s)	Width (w_s)	
		(3,7)
3	4	(7,5)
		(9,2)

Request r_i	Area covered $A\left(r_i \cap \left(\cup_{j=1}^3 t_j\right)\right)$	Weight covered
1	12	4×12
2	9	6×9
3	18	7×18
4	17	3×17
5	16	10×16
Optimal (max) weight covered		$= 439$

Figure 2: PMCLP-PC with 5 axis-parallel rectangular requests and 3 axis-parallel rectangular targets, and its optimal solution (x_{t_j}, y_{t_j}), i.e. coordinates of the lower-left corner of the targets

As an example, Figure 2 shows the optimal positioning of 3 axis-parallel rectangular targets for PMCLP-PC with 5 axis-parallel rectangular requests (the optimal solution is obtained using the exact algorithm presented in [2]). We note that although in this particular example the optimal positions (i.e. coordinates of the lower-left corner) of the targets are such that they do not overlap, it is quite possible in a PMCLP-PC instance that the targets overlap in the optimal solution. Also the numbers in the example of Figure 2 are all integers, just for simplicity of figure illustration. In general they do not have to be integers.

2.2 Applications of the PMCLP-PC beyond facility location

We present a few applications of the PMCLP-PC, and thereby demonstrate its significance.

Telerobotics. In space exploration [16], health-care [19], natural-environment observation [37], surveillance [36], and distance learning, the advent of network-based telerobotic camera systems enable multiple participants or researchers to interact with a remote physical environment using shared resources. This system of p networked robotic cameras receives rectangular subregions as requests from multiple users for monitoring [31, 37, 38]. Each subregion has an associated reward rate (weight per-unit area) that may depend on the priority of the user or the importance level associated with monitoring that subregion. The goal is to select the best view frame for the cameras (rectangular targets) to maximize the total reward from the captured parts of the requested subregions. Interestingly, this problem is same as the PMCLP-PCR. In the literature,

Song et al. [31] study the PMCLP-PCR for single camera, i.e. $p = 1$, and Xu et al. [37, 38] consider the PMCLP with rectangular requests and binary coverage where the rectangular targets are not allowed to overlap. The PMCLP-PC subsumes the aforementioned problems along with their variants with requested subregions of any shape, i.e. polygon, circle, etc.

Disaster Management. In the events of natural disasters such as earthquake and hurricane, and human-made disasters such as oil spill in the oceans, satellite imaging is utilized to gather information about the highly impacted regions for the purpose of the disaster emergency planning [18, 27, 31]. In this spatial optimization problem, the disaster regions (or requests) are represented by polygons with per area-unit intensity of destruction (weights) associated to them, and the objective is to select the position of the best view frame of cameras (rectangular targets) to identify the most destructed regions. The PMCLP-PC subsumes the foregoing problem along with its variants which utilizes varieties of satellite sensors, i.e. targets of different shapes.

3. Greedy and Pseudo-Greedy Algorithms for PMCLP-PC

In this section, we first present greedy and pseudo-greedy algorithms for solving the PMCLP-PC. Thereafter, we prove that the greedy approach and pseudo-greedy approach provide approximate solutions whose values are within a factor of $1 - 1/e$ and $1 - 1/e^\eta$, respectively, of the optimal solution value, where $\eta \leq 1$.

3.1 Greedy Algorithm

Assuming that there exists an exact algorithm for solving the PMCLP-PC with $p = 1$, referred to as Single Target Problem (STP), we solve multiple STPs in our greedy-based approximation algorithm for the PMCLP-PC. The pseudocode is presented in Algorithm 1, where for a set of requests $\mathcal{R}_j^g$, the function $\texttt{SingleTargetProblem}\left(\mathcal{R}_j^g, t_j\right)$, $j \in \{1, \ldots, p\}$, returns the maximum weight, ψ_g^j, covered by the target t_j along with its optimal position $(x_{t_j}^g, y_{t_j}^g)$ (Line 4). We initialize the algorithm in Line 2 by setting $\mathcal{R}_1^g = \mathcal{R}$ (the original set of all given requests). We also use the function $\texttt{TrimOut}\left(r, t_j, x_{t_j}^g, y_{t_j}^g\right)$ to eliminate the parts of request r that are covered by target t_j positioned at $(x_{t_j}^g, y_{t_j}^g)$. In Lines 6-8, we create set $\mathcal{R}_{j+1}^g$ for the next iteration by replacing each request r in the set $\mathcal{R}_j^g$ with trimmed requests. We denote the set of trimmed requests replacing request r by T_r. See Figure 3 for different output possibilities after every application of $\texttt{TrimOut}$ when the request r and target t_j are axis-parallel rectangles. The summation of the maximum covered weight by calling $\texttt{SingleTargetProblem}\left(\mathcal{R}_j^g, t_j\right)$ over $j \in \{1, \ldots, p\}$ gives a feasible solution and a lower bound on the optimal objective value of the PMCLP-PC.

Algorithm 1 Greedy Algorithm for PMCLP-PC

1: **function GreedyAlgorithm**$(\mathcal{R}, \mathcal{T})$

2:     $\mathcal{R}_1^g := \mathcal{R}; \psi_g := 0;$

3:     **for** $j \in P$ **do**

4:         $(\psi_g^j, x_{t_j}^g, y_{t_j}^g) := \texttt{SingleTargetProblem}(\mathcal{R}_j^g, t_j);$ ▷ Returns the maximum weight ψ_g^j covered by
     the target t_j along with its optimal position $(x_{t_j}^g, y_{t_j}^g)$ for the set of requests $\mathcal{R}_j^g$

5:         $\psi_g \leftarrow \psi_g + \psi_g^j;$

6:         **for** $r \in \mathcal{R}_j^g$ **do**

7:             $\mathcal{R}_{j+1}^g \leftarrow \{\mathcal{R}_j^g \setminus r\} \cup \texttt{TrimOut}(r, t_j, x_{t_j}^g, y_{t_j}^g);$ ▷ Remove the parts of request r that are covered
     by target t_j positioned at $(x_{t_j}^g, y_{t_j}^g)$

8:         **end for**

9:     **end for**

10:     **return** $\left(\psi_g, x_{t_1}^g, \ldots, x_{t_p}^g, y_{t_1}^g, \ldots, y_{t_p}^g\right)$
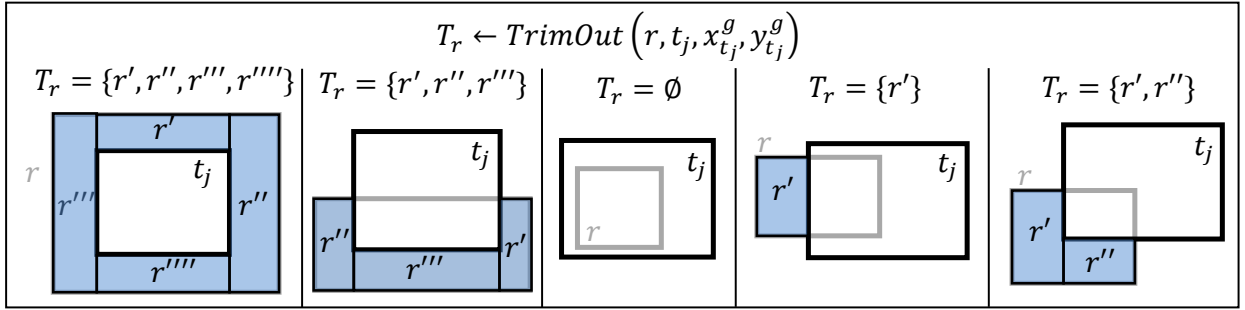
11: **end function**

Figure 3: Possible outputs of `TrimOut` for axis-parallel rectangular request and target

Algorithm 1 generalizes the greedy-based polynomial-time heuristic of Bansal and Kianfar [2] for solving the PMCLP-PCR, i.e. PMCLP-PC where requests r_i, $i = 1, \ldots, n$, and targets t_j, $j = 1, \ldots, p$, are axis-parallel rectangles. Interestingly, for the PMCLP-PCR (a special case of the PMCLP-PC), the function $\texttt{SingleTargetProblem}(\mathcal{R}_j^g, t_j)$, $j \in \{1, \ldots, p\}$, takes $O(n_j^2)$ time where n_j is the number of rectangular requests in the set $\mathcal{R}_j^g$, and returns the optimal weight covered by the rectangular target t_j along with its optimal position $(x_{t_j}^g, y_{t_j}^g)$, i.e. the coordinates of its lower left corner [2, 31]. Moreover, for PMCLP-PCR, each application of `TrimOut` replaces the rectangular request r with at most 4 trimmed rectangular requests, thereby adding at most 3 rectangular requests in $\mathcal{R}_{j+1}^g$. Figure 3 shows that when one, two, or four corners of a rectangular target lie(s) within a request, one, two, or three, respectively, new rectangular (trimmed) requests are created. Notice that since the trimmed requests in T_r do not overlap, any corner of another rectangular target cannot lie in more than one trimmed request. As a result, in the next iteration of the outer for loop, the application of the function `TrimOut` on the trimmed rectangular requests in T_r will add at most 4 rectangular requests in $\mathcal{R}_{j+2}^g$. The same argument applies to all following iterations and all $r \in \mathcal{R}$. This implies that in each iteration of outer loop at most $4n_j$ number

of rectangular requests are added in $\mathcal{R}_j^g$, making $n_{j+1} \leq n(4j - 3)$ for $j \geq 1$. Since the function TrimOut is a constant time operation for this problem, the complexity of the greedy algorithm for PMCLP-PCR is $O\left(\sum_{j=1}^p \left((4j-3)^2 n^2 + (4j-3)n\right)\right) = O(n^2 p^3)$.

Observation 1. *In iteration $j \in \{1, \ldots, p\}$ of the greedy algorithm (Algorithm 1), we exactly solve a STP for $\mathcal{R}_j^g$ set of requests. Observe that summing the weight covered in the iteration, i.e., ψ_g^j, for p times provides an upper bound on the optimal solution value of the PMCLP-PC for $\mathcal{R}_j^g$ set of requests. This is because the summation does not consider overlapping of the targets.*

3.2 Pseudo-Greedy Algorithm

In the above discussed greedy algorithm, we assume that an exact algorithm for solving the STP, i.e. PMCLP-PC with $p = 1$, is known. However, in case this assumption fails and only an η-approximate algorithm for solving the STP is known, then we utilize a pseudo-greedy algorithm for solving the PMCLP-PC. More specifically, the pseudo-greedy algorithm is same as the greedy algorithm (Algorithm 1) except that the function SingleTargetProblem is replaced by function η-ApproxSTP which returns an approximate weight, κ_s^j, covered by the target t_j that is within a factor of η (≤ 1) of the optimal solution value returned by SingleTargetProblem. The pseudocode is given in Algorithm 2. Note that for $\eta = 1$, the pseudo-greedy algorithm is exactly same as the greedy algorithm. Furthermore, for $\eta < 1$, the sets of requests $\mathcal{R}_j^g$ and $\mathcal{R}_j^s$, $j = 2, \ldots, p$, in the greedy algorithm (Algorithm 1) and pseudo-greedy algorithm (Algorithm 2), respectively, are different.

Algorithm 2 Pseudo-Greedy Method for PMCLP-PC

1: **function PseudoGreedyAlgorithm**($\mathcal{R}, \mathcal{T}$)
2:     $\mathcal{R}_1^s := \mathcal{R}; \kappa_s := 0$;
3:     **for** $j \in P$ **do**
4:         $(\kappa_s^j, x_{t_j}^s, y_{t_j}^s) := \eta\text{-ApproxSTP}(\mathcal{R}_j^s, t_j)$;   ▷ Returns an approximate weight, κ_s^j, covered by the target t_j with its optimal position $(x_{t_j}^g, y_{t_j}^g)$ for the set of requests $\mathcal{R}_j^s$
5:         $\kappa_s \leftarrow \kappa_s + \kappa_s^j$;
6:         **for** $r \in \mathcal{R}_j^s$ **do**
7:             $\mathcal{R}_{j+1}^s \leftarrow \{\mathcal{R}_j^s \setminus r\} \cup \text{TrimOut}\left(r, t_j, x_{t_j}^s, y_{t_j}^s\right)$;
8:         **end for**
9:     **end for**
10:    **return** $\left(\kappa_s, x_{t_1}^s, \ldots, x_{t_p}^s, y_{t_1}^s, \ldots, y_{t_p}^s\right)$
11: **end function**

Remark 1. It is important to note that the complexity of the greedy (or pseudo-greedy) algorithm depends on the complexity of the functions SingleTargetProblem (or η-ApproxSTP) and TrimOut, and the number of trimmed requests generated after each iteration. As mentioned before, till date these functions are not known, except for PMCLP-PCR with $p = 1$ [2, 31]. However, whenever they

will be developed in future, we can embed them within our greedy (or pseudo-greedy) algorithm to provide solutions for the PMCLP-PC.

3.3 Approximation Ratios

In the following theorem, we provide approximation ratios associated with the greedy and pseudo-greedy algorithms for the PMCLP-PC. Let $(x_t^*, y_t^*) \in \mathbb{R}^{2p}$ be the optimal solution and $(x_t^a, y_t^a) \in \mathbb{R}^{2p}$ be an approximate solution for the PMCLP-PC. Then the approximation ratio corresponding to the approximate solution (or algorithm) is defined by

$$\gamma_a = \frac{f(x_t^a, y_t^a)}{f(x_t^*, y_t^*)}. \tag{2}$$

Theorem 2. *Let the approximation ratios for the greedy algorithm (Algorithm 1) and the pseudo-greedy algorithm (Algorithm 2) be denoted by γ_g and γ_s, respectively. Then*

$$\gamma_g > 1 - \frac{1}{e}, \text{ and } \gamma_s > 1 - \frac{1}{e^\eta}, \tag{3}$$

where e is the base of natural logarithm.

Proof. Recall that the notations ψ_g^j in Algorithm 1 and κ_s^j in Algorithm 2 denote the weights returned by the functions $\texttt{SingleTargetProblem}\left(\mathcal{R}_j^g, t_j\right)$, and $\eta\text{-}\texttt{ApproxSTP}\left(\mathcal{R}_j^s, t_j\right)$, respectively, for $j \in \{1, \ldots, p\}$. In other words, ψ_g^j is the maximum weight and κ_s^j is the η-approximate weight covered by the target t_j for the given set of requests $\mathcal{R}_j^g$ and $\mathcal{R}_j^s$, respectively. For the sake of convenience, in this proof, we denote the optimal solution value for the PMCLP-PC instance, i.e., $f(x_t^*, y_t^*)$, by f^*. Therefore, the approximation ratio for the greedy algorithm (Algorithm 1) and the pseudo-greedy algorithm (Algorithm 2) are given by

$$\gamma_g = \frac{1}{f^*}\left(\sum_{l=1}^{p} \psi_g^l\right), \text{ and } \gamma_s = \frac{1}{f^*}\left(\sum_{l=1}^{p} \kappa_s^l\right), \tag{4}$$

respectively. Let the optimal solution value of the PMCLP-PC for $\mathcal{R}_j^g$ and $\mathcal{R}_j^s$ set of requests be denoted by ζ_j^g, and ζ_j^s, respectively. Then, based on Observation 1 and the definition of the approximation ratio (2),

$$p\psi_g^j \geq \zeta_j^g, \text{ and } p\kappa_s^j \geq p\eta\psi_s^j \geq \eta\zeta_j^s \tag{5}$$

for $j = 1, \ldots, p$. Note that for $j = 1$, $\mathcal{R}_1^g = \mathcal{R}_1^s = \mathcal{R}$, and hence $\zeta_1^g = \zeta_1^s = f^*$. Also, since after each iteration $l \in \{1, \ldots, j-1\}$ of the greedy algorithm and pseudo-greedy algorithm, we "trim-out"

9

requests of total weight ψ_g^l and κ_s^l, respectively, we get

$$p\psi_g^j \geq \zeta_j^g \geq f^* - \sum_{l=1}^{j-1} \psi_g^l, \text{ and } p\kappa_s^j \geq \eta\zeta_j^s \geq \eta\left(f^* - \sum_{l=1}^{j-1} \kappa_s^l\right), \tag{6}$$

where $\psi_g^0 = \kappa_s^0 = 0$. This implies that

$$\begin{aligned}
p\sum_{l=1}^{j} \psi_g^l &\geq f^* + (p-1)\sum_{l=1}^{j-1} \psi_g^l \\
&\geq f^* + \left(\frac{p-1}{p}\right)\left(f^* + (p-1)\sum_{l=1}^{j-2} \psi_g^l\right) \\
&\geq f^*\left(1 + \frac{p-1}{p} + \left(\frac{p-1}{p}\right)^2 + \ldots + \left(\frac{p-1}{p}\right)^{j-1}\right),
\end{aligned} \tag{7}$$

and

$$\begin{aligned}
p\sum_{l=1}^{j} \kappa_s^l &\geq \eta f^* + (p-\eta)\sum_{l=1}^{j-1} \psi_g^l \\
&\geq \eta f^* + \left(\frac{p-\eta}{p}\right)\left(\eta f^* + (p-\eta)\sum_{l=1}^{j-2} \psi_g^l\right) \\
&\geq \eta f^*\left(1 + \frac{p-\eta}{p} + \left(\frac{p-\eta}{p}\right)^2 + \ldots + \left(\frac{p-\eta}{p}\right)^{j-1}\right).
\end{aligned} \tag{8}$$

For $j = p$, Inequalities (7) and (8) reduce to

$$\gamma_g = \frac{1}{f^*}\left(\sum_{l=1}^{p} \psi_g^l\right) \geq \left(1 - \left(\frac{p-1}{p}\right)^p\right), \text{ and } \gamma_s = \frac{1}{f^*}\left(\sum_{l=1}^{p} \kappa_s^l\right) \geq \left(1 - \left(\frac{p-\eta}{p}\right)^p\right), \tag{9}$$

respectively. Now because the functions in the right-hand sides of the last two inequalities are decreasing in p, we compute limit of these functions as p approaches infinity and get

$$\gamma_g > 1 - \frac{1}{e}, \text{ and } \gamma_s > 1 - \frac{1}{e^\eta}. \tag{10}$$

This completes the proof. $\qquad\square$

Corollary 3. *For a given $p \geq 1$, the greedy algorithm (Algorithm 1) and the pseudo-greedy algorithm (Algorithm 2) provide solutions whose values are at least*

$$1 - \left(\frac{p-1}{p}\right)^p \quad \text{and} \quad 1 - \left(\frac{p-\eta}{p}\right)^p,$$

respectively, times the optimal value for the PMCLP-PC instance.

4.   Conclusion

We introduced a new generalization of the classical planar maximum coverage location problem (PMCLP) where demand zones and service zone of facilities are represented by two-dimensional spatial objects such as line segments, polygons, circles, etc., and the partial coverage is allowed in its true sense. The main objective of this generalization, denoted by PMCLP-PC, is to provide a novel framework for facility location which will mitigate significant errors and uncertainty that arise due to the binary coverage assumption (including demand aggregation) in the PMCLP. Additionally, we presented a greedy algorithm and a pseudo-greedy algorithm to provide approximate solutions for the general PMCLP-PC by assuming that an exact algorithm and an η-approximate algorithm, respectively, for the PMCLP-PC with $p = 1$ (single facility or target) exist. We also proved that the solution value corresponding to the greedy (or pseudo-greedy) solution is within a factor of $1 - 1/e$ (or $1 - 1/e^\eta$) of the optimal solution value where e is the base of natural logarithm and $\eta \leq 1$. It is important note that till date no exact or η-approximate algorithm is known for PMCLP-PC with $p = 1$, except the so-called plateau vertex algorithm (PVT) [31] and improved PVT [2] exact algorithms for PMCLP-PCR with $p = 1$. However, whenever such algorithms will be developed in future for other special cases of the STP, they can be automatically embedded within the greedy (or pseudo-greedy) algorithm presented in this paper to provide solutions for the multiple facility (or target) version of the corresponding problem.

References

[1] Adel A. Aly and John A. White. Probabilistic Formulation of the Emergency Service Location Problem. *Journal of the Operational Research Society*, 29(12):1167–1179, December 1978.

[2] Manish Bansal and Kiavash Kianfar. Planar maximum coverage location problem with partial coverage and rectangular demand and service zones. *INFORMS Journal on Computing*, 29(1):152–169, 2017.

[3] Regina Benveniste. A Note on the Set Covering Problem. *Journal of the Operational Research Society*, 33(3):261–265, March 1982.

[4] Chen-Hua Chung. Recent applications of the maximal covering location planning (M.C.L.P.) model. *The Journal of the Operational Research Society*, 37:735–746, 1986.

[5] Richard L. Church. The planar maximal covering location problem. *Journal of Regional Science*, 24:185, May 1984.

[6] Richard L. Church. Location modelling and GIS. In P. Longley, M. Goodchild, D. Maguire, and D. Rhind, editors, *Geographical Information Systems*. Wiley, New York, 2 edition, 1999.

[7] Richard L. Church. Geographical information systems and location science. *Computers & Operations Research*, 29:541–562, May 2002.

[8] Richard L. Church and Alan T. Murray. *Business Site Selection, Location Analysis and GIS*. John Wiley & Sons, Inc., 2008.

[9] Richard L. Church and Alan T. Murray. *Location covering models history applications and advancements*. Springer, 2013.

[10] Richard L. Church and C. ReVelle. The maximal covering location problem. *Papers of the Regional Science Association*, 32:101–118, 1974.

[11] John R Current and David A Schilling. Analysis of errors due to demand data aggregation in the set covering and maximal covering location problems. *Geographical Analysis*, 22:116–126, April 1990.

[12] Mark S. Daskin, Ali E. Haghani, Mandar Khanal, and Chryssi Malandraki. Aggregation effects in maximum covering models. *Annals of Operations Research*, 18:115–139, April 1989.

[13] Zvi Drezner and Horst W. Hamacher, editors. *Facility Location: Applications and Theory*. Springer, New York, June 2002.

[14] H. A. Eiselt and Vladimir Marianov. Pioneering Developments in Location Analysis. In *Foundations of Location Analysis*, International Series in Operations Research & Management Science, pages 3–22. Springer, Boston, MA, 2011.

[15] Reza Zanjirani Farahani, Nasrin Asgari, Nooshin Heidari, Mahtab Hosseininia, and Mark Goh. Covering problems in facility location: A review. *Computers & Industrial Engineering*, 62(1):368–407, February 2012.

[16] G. Hirzinger, B. Brunner, J. Dietrich, and J. Heindl. Sensor-based space robotics-ROTEX and its telerobotic features. *IEEE Transactions on Robotics and Automation*, 9(5):649–663, 1993.

[17] Kamyoung Kim and Alan T. Murray. Enhancing spatial representation in primary and secondary coverage location modeling. *Journal of Regional Science*, 48(4):745–768, October 2008.

[18] Shufan Liu and Michael E. Hodgson. Optimizing large area coverage from multiple satellite-sensors. 50(6):652–666, 2013.

[19] Jacques Marescaux, Joel Leroy, Francesco Rubino, Michelle Smith, Michel Vix, Michele Simone, and Didier Mutter. Transcontinental Robot-Assisted Remote Telesurgery: Feasibility and Potential Applications. *Annals of Surgery*, 235(4):487–492, 2002.

[20] Abraham Mehrez and Alan Stulman. The maximal covering location problem with facility placement on the entire pane. *Journal of Regional Science*, 22:361, 1982.

[21] Harvey J. Miller. GIS and geometric representation in facility location problems. *International Journal of Geographical Information Systems*, 10:791, 1996.

[22] Alan Murray. Geography in coverage modeling: Exploiting spatial structure to address complementary partial service of areas. *Annals of the Association of American Geographers*, 95:761–772, December 2005.

[23] Alan T. Murray. Advances in location modeling: GIS linkages and contributions. *Journal of Geographical Systems*, 12(3):335–354, September 2010.

[24] Alan T. Murray. Maximal Coverage Location Problem: Impacts, Significance, and Evolution. *International Regional Science Review*, 39(1):5–27, January 2016.

[25] Alan T. Murray and Morton E. O'Kelly. Assessing representation error in point-based coverage modeling. *Journal of Geographical Systems*, 4:171–191, June 2002.

[26] Alan T. Murray, Morton E. O'Kelly, and Richard L. Church. Regional service coverage modeling. *Computers & Operations Research*, 35:339–355, February 2008.

[27] Alan T. Murray and Daoqin Tong. Coverage optimization in continuous space facility siting. *International Journal of Geographical Information Science*, 21:757, 2007.

[28] Alan T. Murray, Daoqin Tong, and Kamyoung Kim. Enhancing classic coverage location models. *International Regional Science Review*, 33:115 –133, April 2010.

[29] C. S. ReVelle and H. A. Eiselt. Location analysis: A synthesis and survey. *European Journal of Operational Research*, 165(1):1–19, August 2005.

[30] David A. Schilling, Vaidyanathan Jayaraman, and Reza Barkhi. A review of covering problems in facility location. *Location Science*, 1:25–55, 1993.

[31] Dezhen Song, A.F. van der Stappen, and K. Goldberg. Exact algorithms for single frame selection on multiaxis satellites. *IEEE Transactions on Automation Science and Engineering*, 3:16–28, 2006.

[32] Daoqin Tong and Richard L. Church. Aggregation in continuous space coverage modeling. *International Journal of Geographical Information Science*, 26(5):795–816, May 2012.

[33] Daoqin Tong and Alan T. Murray. Maximising coverage of spatial demand for service. *Papers in Regional Science*, 88:85–97, March 2009.

[34] C. D. T. Watson-Gandy. Heuristic procedures for the *m*-partial cover problem on a plane. *European Journal of Operational Research*, 11:149–157, October 1982.

[35] Hu Wei, Alan T. Murray, and Ningchuan Xiao. Solving the continuous space p-centre problem: planning application issues. *IMA Journal of Management Mathematics*, 17(4):413–425, October 2006.

[36] Yiliang Xu and Dezhen Song. Systems and algorithms for autonomously simultaneous observation of multiple objects using robotic PTZ cameras assisted by a wide-angle camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3802–3807, 2009.

[37] Yiliang Xu, Dezhen Song, and Jingang Yi. Exact algorithms for non-overlapping 2-frame problem with non-partial coverage for networked robotic cameras. In *IEEE Conference on Automation Science and Engineering (CASE)*, pages 503–508, 2010.

[38] Yiliang Xu, Dezhen Song, Jingang Yi, and A. F. van der Stappen. An approximation algorithm for the least overlapping p-Frame problem with non-partial coverage for networked robotic cameras. In *2008 IEEE International Conference on Robotics and Automation*, pages 1011–1016, May 2008.