The continuous maximal covering location problem in large-scale natural disaster rescue scenes

Yang, P., Xiao, Y., Zhang, Y., Zhou, S., Yang, J. & Xu, Y.

Abstract: This study proposes a continuous maximal covering location problem (C-MCLP) that is often confronted in the rescuing scenes of natural disasters such as earthquakes, floods, and storms. The aim of the research is to optimize (dynamically and rapidly) the continuous locations of the communication hub-centers (e.g., moving vehicles or boats) of the self-organizing mobile network that is quickly established in such signal-free fields. The proposed C-MCLP well represents the real emergency rescues, but it is more complex to solve than the traditional discrete MCLP models, where the hub facilities are typically immobile and placed only within a limited set of candidate sites. We develop two mixed-integer linear programming (MILP) models for the C-MCLP. The first model is the single-period C-MCLP model, which is applicable to a stochastic rescuing environment where the rescue teams (RTs) do not have planned movements and can move towards any direction. The second one is the multi-period C-MCLP model, which is for cases where RTs have planned movements in multiple periods/phases. We introduce a new linearization method for the non-linear Euclidean distance with a controllable approximation error allowance, by which the proposed models are linearized and can be solved optimally using commercial MIP solvers such as CPLEX and Lingo. To solve large-sized problems, we develop a MILP-based fix-and-optimize heuristic approach to obtain near-optimal solutions with high computational efficiency. Then we conduct simulation experiments to verify the proposed models and heuristic approach with an intended time-limit setting on small-sized and large-sized test problem instances, respectively, with up to 1000 nodes of rescue teams. Finally, experimental results are analyzed and compared with those obtained using the traditional *k*-means clustering algorithms, which confirm that the proposed models and approach are applicable for the C-MCLPs in emergency rescue scenes, and can yield rapid and good solutions.

Keywords：set covering; location problem; mobile communication network; disaster rescue; optimization

1. Introduction

Along with increasing intensification of human activities and global climate change, the occurrences of natural disasters in recent decades appeared to be much more frequent, posing significant and growing threats to human life and safety. According to the International Disaster Database (EMDAT, 2019), as shown in Fig.1, there was a significant increase of natural and geophysical disasters in recent years, and the estimated economic damage in 2017 was reported as 334 billion USD (Below and Wallemacq, 2018).
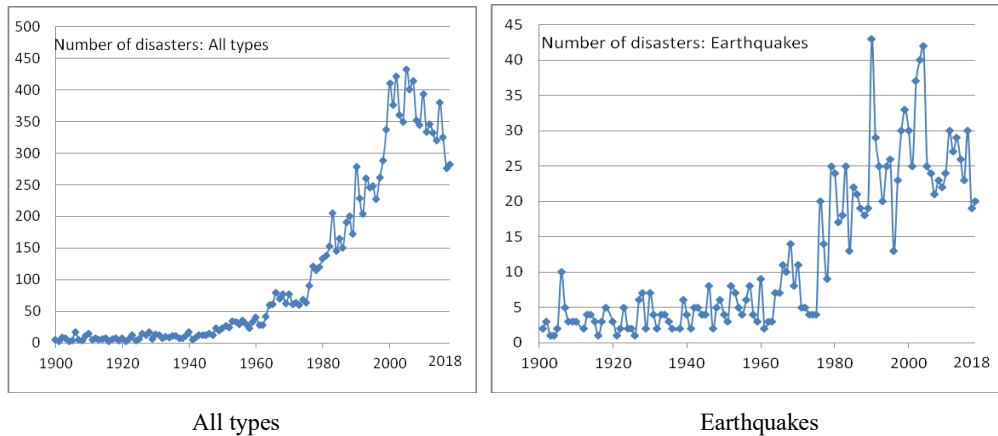
<div align="center">All types   Earthquakes</div>

Fig. 1. Natural Disasters: an upward trend
(Source: https://ourworldindata.org/grapher/number-of-natural-disaster-events)

     Most natural disasters are unpredictable with respect to when and where they occur; however, they always cause big damage to local infrastructure and significant losses in human lives and properties. A typical example presented in Fig. 2 is the well-known Wenchuan earthquake occurred in west China on May 12, 2008 (Wenchuan, 2008); In the incidents almost all the buildings in the red and yellow regions were reported as completely damaged. In addition, most part of the green region was exposed to an unrecoverable electrical power interruption. To minimize the post-disaster casualties, rescue operations should be carried out immediately within the so-called golden rescuing period. According to the statistical data, if disaster rescues are carried out within the first 72 hours, as shown in Fig. 3 for three time-varying survival rates, most survivors (more than 50%) could be rescued within the first 24 h, and the survival rate decreases dramatically as the golden time passes. However, the communication systems in the stricken areas are disrupted in most cases; thus, the rescue and relief operations are very inefficient and difficult to conduct, which leads to more casualties. In such cases, emergency communication vehicles (CVs) are typically deployed in the area to provide dedicated wireless communication coverage for rescue operations. The CVs, loaded with base transceiver station (BTS) antenna equipment, can transmit data, voice, picture/video, and other information in a timely manner, and provide onsite signal channels connecting the rescue teams (RTs) and the control centers. In addition, they help the command center and rescue teams to allocate rescue/relief resources more efficiently. Although there may be hundreds or thousands of RTs searching in a large-scale field of the incident, the number of available CVs are normally insufficient to cover the entire area, given that each individual CV has a limited capacity and effective covering radius (e.g., 1–3 km depending on the terrain). The dynamic and stochastic movements of the RTs makes it more difficult to be covered by communication signals all the time, it is therefore necessary to manage the CVs efficiently with continuously optimized locations, well-scheduled moving paths, and dynamic assignments of servicing nodes, to provide an overall maximal coverage of the RTs in the rescue process. This optimization problem hereby is referred to as the continuous maximal covering location problem (C-MCLP).
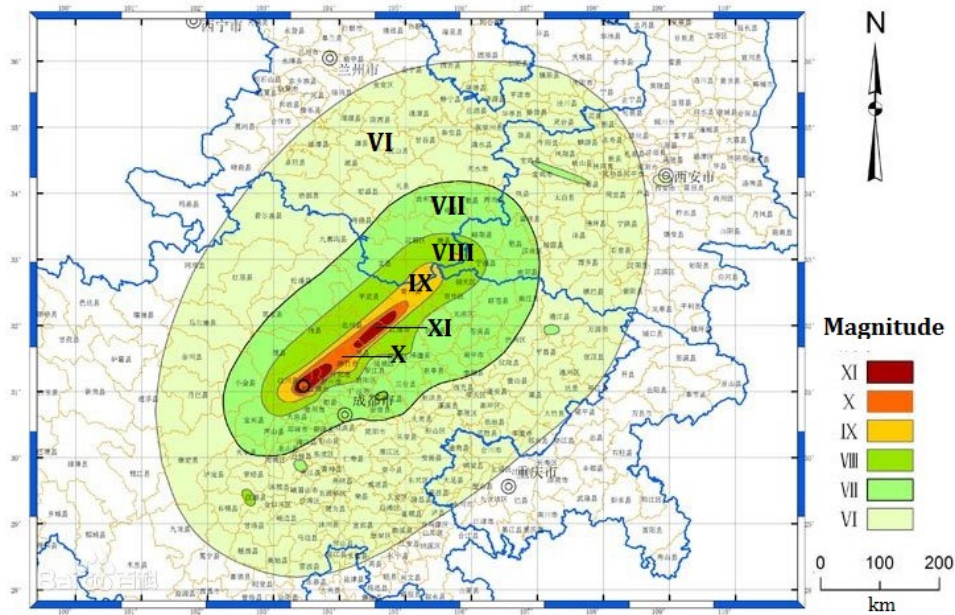
Fig. 2. Stricken region of the 512-Wenchuan-Earthquake (source: Wenchuan (2008))
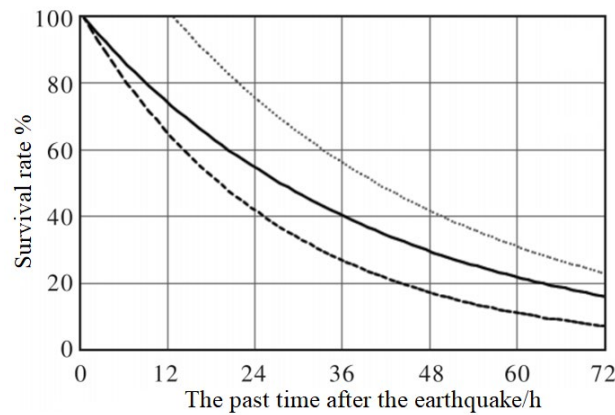

Fig. 3. General survival rates after an earthquake in three locations (source: Gao and Nie (2015))

The C-MCLP can be considered as a variant of the classical location problem, which has been extensively studied and has important application in many fields. In 1929, the industrial layout problem proposed by German scholar Alfred Weber was first recognized as a continuous site selection problem. Thereafter, it was referred to as the Weber problem (Friedrich, 1929). In geometry, the Weber problem is one of the most well-known problems in location theory, which has the objective to determine a point in the plane that minimizes the total transportation costs from the point to n destination points. Several variants of the site selection problem have subsequently been developed, such as the set covering problem, center problem, and p-median problem (Brandeau and Chiu, 1989; Wang, 2006).

The MCLP was first investigated by Church and ReVelle (1974) with respect to network design. Thereafter, various extensions to the original MCLP problem have been studied and modeled in many application fields, which were reviewed in their published time sequence as follows. Akella et al. (2007) proposed a mixed-integer programming (MIP) model that integrates the base station location problem, frequency channel assignment problem, and emergency notification problem. They developed a Lagrangean heuristic technique to improve the optimality gap further, based on the solution obtained by a greedy heuristic. As the extension of the application of

MCLP model, Erdemir et al. (2010) considered three coverage options of emergency services for motor vehicle crashes to establish a MCLP model considering both the response time and total service time. Their models also considered the uncertainties of vehicle crash distribution. Davari et al. (2011) extended the MCLP to a fuzzy maximal covering location problem (FMCLP), where the travel time between any pair of nodes was considered as a fuzzy variable, and they designed fuzzy expected value maximization model and a simulated annealing (SA) based solution approach. Yin and Mu (2011) extended the MCLP to take into account the capacity of the facilities, which is named as the modular capacitated maximal covering location problem (MCMCLP), where each of the potential sites (for building facilities) had several possible capacity levels to select. On the basis of single-period MCLP model, Dell'Olmo et al. (2013) presented a multi-period MCLP model for optimizing the periodical locations of the intersection safety cameras on an urban traffic network in a given planning horizon. Mohammadi et al. (2014) applied the MCLP model to optimize the location of emergency medical service centers considering a comprehensive objective including response time, cost of sites construction, and ambulance coverage, and with constraints on available ambulance number, limited budget, and minimum coverage level. Hochbaum and Pathria (2015) considered a general case of the maximum k-coverage problem and analyzed the quality of the k-stage covering algorithm. Taking the uncertain distribution of covered demand into consideration, Zhang et al. (2017) developed an (α, β)-maximal covering location model and an α-chance maximal covering location model under the uncertain distribution of covered demand. Paul et al. (2017) formulated a multi-objective hierarchical extension of the MCLP model, for the maximization of the population coverage, which requires rapid response windows and minimized modifications to the present structure. Bhattacharya and Nandy (2013) studied the k facilities maximal coverage problem, which is referred to as k-MaxCov, to maximize the total number of users served by new added facilities. They proposed an algorithm with a complexity time of $O(n \times \log(n))$ for the k-MaxCov problem.

In addition to the modeling research on MCLPs, there are also a number of papers and technical reports in literature related to the methods of solving the MCLP. Pirkul and Schilling (1991) presented an efficient solution procedure that is particularly applicable to the assignment of uncovered demand to old facilities. Galinier and Hertz (2007) developed three exact algorithms for the large set covering problem that aims to find a minimum set of element subsets to cover the entire elements. Corrêa et al. (2009) suggested the use of column generation and covering graph approaches to obtain competitive solutions for the probabilistic MCLP instances up to 818 vertices in reasonable computational time. Guzmán et al. (2016) used a parametric approach to solve the fuzzy extension of the MCLP model, where they transformed the fuzzy model into several crisp problems using a decision parameter, and then solved the problems using classical optimization techniques.

Heuristic algorithms were often applied for solving large scale MCLP problems where exact algorithms are not applicable, such as genetic algorithm (GA) (Zarandi et al., 2011), simulated annealing (SA) (Rabieyan and Esfandiari, 2011), variable neighborhood search (VNS) (Davari et al., 2013), particle swarm optimization (PSO) (Takaci et al., 2012), and hybrid algorithms (Ma et al., 2012; Davari et al., 2013). Zarandi et al. (2011) developed a genetic algorithm (GA) based heuristic, referred to as the customized GA, for solving the problems up to 2500 nodes with fair accuracy. Rabieyan and Esfandiari (2011) proposed a SA algorithm to solve the proposed novel queuing location problem, which can be considered as a variant of the MCLP. Takaci et al. (2012) used a discrete particle swarm optimization (DPSO) algorithm to solve the fuzzy maximal covering location time problem. Ma et

al. (2012) established a mathematical model for the location problem of emergency service facilities, and then developed a solution approach referred to as the genetic-simulated annealing algorithm (GSAA). Davari et al. (2013) developed a combination of the variable neighborhood search (VNS) and fuzzy simulation to solve the large-scale MCLP instances on a plane with fuzzy coverage radii under the Hurwicz criterion. Mcgregor et al. (2018) studied the maximum k-set coverage problem using the data streaming algorithm. Other meta-heuristic algorithms include the variable neighborhood search algorithm (Xiao et al., 2012, 2014) and hybrid genetic algorithm (Xiao et al., 2017).

In this study, we considered a continuous and dynamic case of the MCLP in a large-scale disaster rescuing scene, where the locations of CVs were taken as continuous decision variables to optimize the routes of CVs in multiple periods and phases. The optimization goal is to achieve an overall maximal coverage of the RTs in the whole rescuing process. The contribution and novelty of this research are outlined as follows.

- We advanced the dynamic C-MCLP that involves multiple periods/phases of planned rescue operations in large-scale scenes, considering continuous arriving/joining of new RTs and CVs over time and two hierarchical objective functions on the maximal signal coverage (primary) and the minimized total moved distance (secondary) of the CVs.

- We presented two mixed-integer programming (MIP) models for the C-MCLP: the single-period C-MCLP model and the multi-period C-MCLP model to cater for different cases in rescuing.

- We developed a linearization method for the nonlinear Euclidean distance function, by which the proposed MIP models were transformed into mixed-integer linear programming (MILP) models that can be optimally and efficiently solved using commercial MIP solvers.

- A MILP-based heuristic algorithm, which is referred to as the MILP-based dynamic iterative partial optimization (MILP-DIPO), was developed to obtain near-optimal solutions for large-sized C-MCLPs rapidly.

- We tested the proposed models and algorithms in an AMPL/CPLEX environment with simulated large-scale problem instances with up to 1000 nodes of rescue teams.

The rest part of this paper is organized as follows. In Section 2, we first introduce and describe the problem, and then provide two MIP models for the single-period C-MCLP and multi-period C-MCLP in the two sub-sections. After that, a linearization method is provided for the nonlinear Euclidean distance function. In Section 3, discussions on solution approaches and algorithms are presented, which includes a two-stage optimization framework, the traditional k-mean clustering algorithm, and MILP-based heuristic algorithm. In Section 4, simulated experiments are conducted on different scales of disaster rescue scenarios and the experimental results are analyzed. Finally, the conclusions and future work are presented in Section 5.

2. Problem description and formulation

The C-MCLP can be described as follows. For a region subject to a natural disaster, e.g., earthquake, tsunami, floods, or storms; local infrastructure is exposed to significant damage, and communication and power supply interruptions. To search for and rescue the survivors injured or trapped beneath collapsed buildings, mountains, or buried by landslides, a group of rescue teams (RTs), denoted by set N, enter into this region to carry out emergency rescue operations, followed by a fleet of communication vehicles (CVs), denoted by set V, to provide the mobile

communication services for the RTs. Thus, the RTs, CVs, and links between them compose a dynamically self-organizing communication network. With a preliminary analysis of the disaster-stricken area, the command center is assumed to have developed a rough initial plan of action routes for each group of RTs, to guide their search over a set of pre-defined target areas that are represented by location coordinates (x_{it}, y_{it}), where $i \in N$, $t \in T$, and T is the set of periods (or phases) of the plan. To provide mobile communication services, the CVs are required to dynamically select a number of RTs to follow and to determine the optimal target locations in each period. Each CV is assumed to be subject to the following restrictions: (1) a maximum covering radius $D^{\max}$, only within which the RTs can be covered by wireless-signals; (2) a maximum channel capacity C that represents the maximum number of RTs a CV can service, and (3) a maximal distance $L_{\max}$ that a CV can travel between two consecutive periods. A binary variable E_{ijt} is used to indicate whether RT i is covered by CV j in period t ($E_{ijt} = 1$) or not ($E_{ijt} = 0$). Thus, $E_{ijt} = 1$ if the distance d_{ijt} between RT i and CV j in period t is within $D^{\max}$ while $\sum_{i \in N} E_{ijt} \leq C$; otherwise, $E_{ijt} = 0$, where $E_{ijt} \in \{0,1\}, i \in N, j \in V, t \in T$. Thus, an RT is considered to be connected to the self-organizing communication network if (and only if) it is serviced/covered by a minimum of one CV. In addition, more RTs and CVs are supposed to be continuously arriving at the field and start their rescue actions from the arrived periods. Each CV, j, is set with an initial point (X_{0j}, Y_{0j}) from which the vehicle enters the field. The objective of the C-MCLP is to maximize the overall coverage of all RTs over the entire planning horizon, and the decision variables include the binary assignments of RTs to CVs and the continuous route locations of CVs in all periods/phases.

The above described C-MCLP is a complex combination of the continuous site location problem with the task selection problem in multiple periods. It is difficult to obtain optimal solution for large-size problems. Further, given that the rescuing environments are time-varying with uncertainties, such as unexpected events and new arrivals of RTs/CVs, the C-MCLP should be rapidly solved within a short CPU time, otherwise the outcome solution may be no longer fit for the fast-changing situation. In practice, the problem size may be large, and it may involve the service of hundreds/thousands of RTs in such a signal-free area. The formulation of the C-MCLP is presented with two MIP models in Sections 2.1 and 2.2 for single and multiple periods, respectively.

2.1 The single-period C-MCLP model

The single-period C-MCLP considers a random search and rescue scene where the RTs do not have planned movements and are free to move at all directions whereas the CVs must constantly change their positions to follow the RTs to provide a maximum coverage. In the single-period C-MCLP mode, each RT i, $i \in N$, is associated with a current location (x_i, y_i) and a random deviation Δ_i of Euclidean distance from the current location. The CVs have current locations $(\overline{X}_j, \overline{Y}_j)$ that are supposed known, and need to find out a set of targeting locations (X_j, Y_j) they can move to for serving the RTs. The random deviation Δ_i is introduced because the RTs are also moving randomly while the CVs are moving to their optimized targeting locations. The CVs are subject to certain limitations such as maximum covering radius $D_j^{\max}$, maximum channel capacities C_j, and maximum moving length $L^{\max}$ during the time period a CV can travel. The $L^{\max}$ is determined normally by the travel speed of CV and the interval length of

re-optimized CV locations. The parameters and notations used in the single-period C-MCLP model are introduced as follows.

Parameters:

N	set of nodes that represents the RTs
V	set of nodes that represents the CVs
i	index of RTs, where $i \in N$
j	index of CVs, where $j \in V$
D_j^{max}	maximum covering radius of CV j, where $j \in V$
(x_i, y_i)	current location coordinates of RT i, where $i \in N$
Δ_i	a random deviation of RT i from its current location, where $i \in N$
$(\overline{X}_j, \overline{Y}_j)$	current location coordinates of CV j, where $j \in V$
L^{max}	distance that a CV can travel during the time interval of re-optimization
M	a large number
C_j	capacity of the CV j, where $j \in V$, representing the maximum number of RTs a CV can service

Decision variables:

(X_j, Y_j)	continuous variables that indicates the new/targeting location coordinates of CV j
$d_{ij}^x, d_{ij}^y, d_{ij}$	continuous variables that indicate the new/targeting distances between RT i and CV j in the x-axis, y-axis directions, and in Euclidean metric
d_{ij}	continuous variable that indicates the Euclidean distance between RT i and CV j
E_{ij}	binary variable that indicates whether RT i is served by CV j ($E_{ij} = 1$) or not ($E_{ij} = 0$)
e_i	binary variable that indicates whether RT i is served by a CV ($e_i = 1$) or uncovered ($e_i = 0$)
D_j^x, D_j^y, D_j	continuous variables indicating the distance travelled for CV j in the x-axis, y-axis directions, and in Euclidean metric

The objective function is the maximization of the signal coverage rate (SCR) of the RTs, i.e., the ratio of covered RTs to the total number of RTs. Thus, the single-period C-MCLP can be formulated using a mixed-integer nonlinear programming model as follows.

Problem *single-period C-MCLP*:

Max. $Signal\_Coverage\_Rate\ (SCR) = \sum_{i \in N} e_i / |N|$ (1)

Subject to:

$$\begin{cases} d_{ij}^x = \left| x_i - X_j \right| \\ d_{ij}^y = \left| y_i - Y_j \right| \\ d_{ij} = \sqrt{(d_{ij}^x)^2 + (d_{ij}^y)^2} \end{cases} \qquad \forall i \in N, \ j \in V \tag{2}$$

$$M \cdot (E_{ij} - 1) \le (D_j^{\max} - d_{ij} - \Delta_i) \qquad \forall i \in N, j \in V \tag{3}$$

$$C_j \ge \sum_{i \in N} E_{ij} \qquad \forall j \in V \tag{4}$$

$$e_i \le E_{ij} \qquad \forall i \in N, j \in V \tag{5}$$

$$\begin{cases} D_j^y = \left| \bar{X}_j - X_j \right| \\ D_j^y = \left| \bar{Y}_j - Y_j \right| \\ D_j = \sqrt{(D_j^x)^2 + (D_j^y)^2} \end{cases} \qquad \forall j \in V \tag{6}$$

$$D_j \le L^{\max} \qquad \forall j \in V \tag{7}$$

$$\begin{cases} X_j \ge 0, Y_j \ge 0, d_{ij}^x \ge 0, d_{ij}^y \ge 0, d_{ij} \ge 0, \\ D_j^x \ge 0, D_j^y \ge 0, D_j \ge 0, e_i \in \{0,1\}, E_{ij} \in \{0,1\} \end{cases} \qquad \forall i \in N, j \in V \tag{8}$$

In the above formulations, Constraints (2) calculate the Euclidean distances between the RTs and CVs. It should be noted that the Euclidean distances calculated in Constraints (2) are nonlinear; however, they can be linearized using the method introduced in the Section 2.3. Constraints (3) ensure that the variable E_{ij} takes a zero value when the distance plus deviation, $d_{ij} + \Delta i$, is greater than $D_j^{\max}$, such that only the RTs within the covering radius are counted for the coverage rate. Constraint (4) ensures that the capacities of the CVs are not exceeded. Constraint (5) indicates that an RT can be identified as covered when it is covered by a CV. Constraints (6) calculate the moved distances of CVs in the x-axis, y-axis directions, and in Euclidean metric. Constraints (7) restrict the moved distance of each CV within the maximum length. Constraint (8) defines the value domains of all the variables. The linearized single-period MCLP model was coded in AMPL, as shown in Fig. A3 in Appendix, which was also used in the computational experiments of Section 4.

2.2 The multi-period C-MCLP model

The multi-period C-MCLP considers an overall optimization of the continuous location coordinates (X_{jt}, Y_{jt}) of the CVs in multiple planned periods/phases. As an example shown in Fig. 4, the yellow circle represents the starting point of CVs, and three groups of rescue teams, represented by filled red, blue and green circles respectively, are covered by three CVs represented by hollow circles of the same color in rescuing process. The arrows show the planned rescue paths of CVs. This example assumed that all RTs have planned actions in rescuing, and the CVs are following the RTs with optimized moving paths to provide effective wireless communication service. The multi-period C-MCLP model needs to consider the increasing numbers of RTs and CVs along the periods, because more RTs and CVs may continuously arrive and join the rescue actions over time. In comparison to the single-period model, the multi-period model has several differences, as follows:

(1) The RTs have planned targeting locations in multiple periods/phases.

(2) Each CV has an initial location and a maximum distance it can traverse over one period.

(3) The numbers of RTs and CVs may increase as the time goes on, e.g., in the first period, there may be only 10 rescue teams, while in the second period there may be 15 rescue teams including 5 new joined ones.

(4) All CVs need to serve the RTs with a maximized SCR (primary objective) and at the same time to travel at the shortest total moved distance (secondary objective) in all periods/phases.
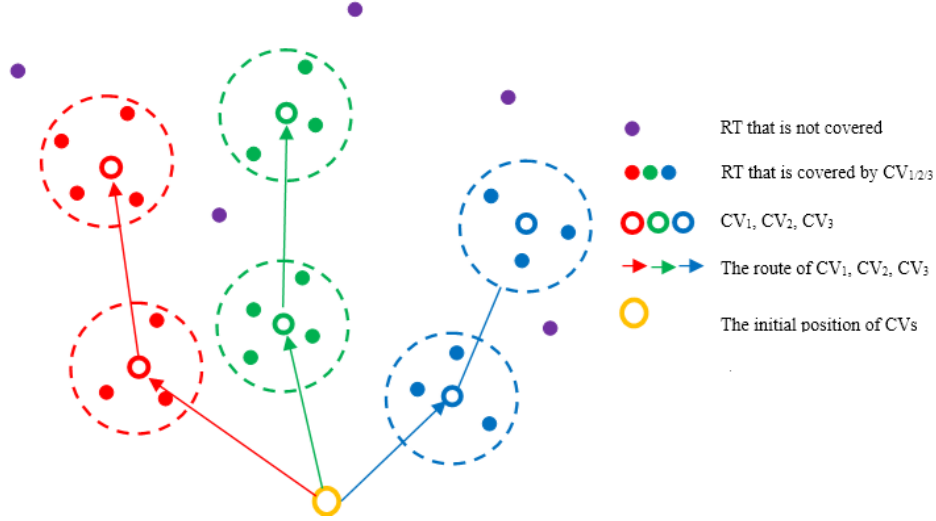
Fig. 4. Example of the two-period C-MCLP

The parameters and decision variables used to describe the multi-period C-MCLP model are listed below.

Parameters

T	set of periods/phases (hereafter, *period* is used for simplicity)
t	index of period, where $t \in T$
N_t	set of RTs that are available in period t, where $t \in T$
V_t	set of CVs that are available in period t, where $t \in T$
i	index of RTs, where $i \in N_t$
j	index of CVs, where $j \in V_t$
$D_j^{\max}$	maximum covering radius of CV j, where $j \in V_t$
$L^{\max}$	maximal distance that a CV can travel within one period
(x_{it}, y_{it})	coordinates of RT i in period t, where $i \in N_t$ and $t \in T$
(X_{j0}, Y_{j0})	initial location coordinates of CV j, where $j \in V$
C_j	capacity of the CV j, where $j \in V$
M	a large number

Decision variables

(X_{jt}, Y_{jt}) continuous variable indicating the coordinates of CV j in period t.

d_{ijt}^x, d_{ijt}^y continuous variables indicating the distances between RT i and CV j in period t in the x-axis and

             y-axis directions

d_{ijt} continuous variable indicating the Euclidean distance between RT i and CV j in period t.

D_{jt}^x, D_{jt}^y continuous variables indicating the moved distances of CV j in the x-axis and y-axis directions in

             period t

D_{jt} continuous variable indicating the moved distance of CV j in Euclidean metric in period t

E_{ijt} binary variable indicating whether RT i is served by CV j ($E_{ij} = 1$) or not ($E_{ij} = 0$)

e_{it} binary variable that indicates whether RT i is served by one CV ($e_i = 1$) or uncovered ($e_i = 0$)

The multi-period C-MCLP considers two hierarchical objective functions. The first function is to maximize the overall SCR of all the RTs in all periods. The second one is to minimize the total moved distance (TMD) of all the CVs, while keeping the maximized SCR unchanged. Thus, the multi-period C-MCLP can be formulated as a two-objective mixed-integer nonlinear programming model as follows.

Problem *multi-period C-MCLP*:

Max.   *Signal_Coverage_Rate (SCR)* $= \sum_{i \in N_t, t \in T} e_{it}' / |N_t|$ (9)

Min.   *Total_Moved_Distance (TMD)* $= \sum_{j \in V_t, t \in T} D_{jt}$ (10)

Subject to:

$$\begin{cases} d_{ijt}^x = \left| X_{jt} - x_{it} \right| \\ d_{ijt}^y = \left| Y_{jt} - y_{it} \right| \\ d_{ijt} = \sqrt{(d_{ijt}^x)^2 + (d_{ijt}^y)^2} \end{cases} \qquad \forall i \in N_t, j \in V_t, t \in T \qquad (11)$$

$$\begin{cases} D_{jt}^y = \left| X_{jt} - X_{j,t-1} \right| \\ D_{jt}^y = \left| Y_{jt} - Y_{j,t-1} \right| \\ D_{jt} = \sqrt{(D_{jt}^x)^2 + (D_{jt}^y)^2} \end{cases} \qquad \forall j \in V_t, t \in T \qquad (12)$$

$M \cdot \left(E_{ijt} - 1 \right) \le \left(D_j^{max} - d_{ijt} \right)$ $\forall i \in N_t, j \in V_t, t \in T$ (13)

$D_{jt} \le L^{max}$ $\forall j \in V_t, t \in T$ (14)

$C_j \ge \sum_{i \in N_t} E_{ijt}$ $\forall j \in V_t, t \in T$ (15)

$e_{it}' \le E_{ijt}$ $\forall j \in V_t, i \in N_t, t \in T$ (16)

$$\begin{cases} X_{jt} \geq 0, Y_{jt} \geq 0, d_{ijt}^x \geq 0, d_{ijt}^y \geq 0, \\ d_{ijt} \geq 0, D_{jt}^x \geq 0, D_{jt}^y \geq 0, D_{jt} \geq 0, \qquad \forall i \in N_t; j, t \in N \\ e_{it} \in \{0,1\}, E_{ijt} \in \{0,1\} \end{cases} \tag{17}$$

In the above formulations, Constraints (11) calculate the Euclidean distances between the RTs and the CVs in multiple periods, and Constraints (12) calculate the moved Euclidean distances of the CVs in multiple periods. Similarly, the nonlinear characteristics in these formulations can be linearized using the method introduced in Section 2.3 for the nonlinear Euclidean distance linearization. Constraints (13) ensure that the variable $E_{ijt} = 0$ when the distance $d_{ijt} > D_j^{\max}$. Constraint (14) guarantees that the moved distance of each CV in each period is less than or equal to the given maximum distance $L_{\max}$. Constraint (15) ensures that the capacity of each CV is not exceeded. Constraint (16) indicates that an RT can only be identified as covered when it is covered by a minimum of one CV. Constraint (17) defines the value domains of all the variables. The linearized version of the multi-period MCLP model written in AMPL code is shown in Fig. A4 in Appendix. Moreover, it was used in the computational experiments presented in Section 4.

2.3 A Euclidean distance linearization method

The Euclidean distance metrics used in Constraints (2), (6), (11), and (12) of the C-MCLP formulations are typically nonlinear expressions, which make the models non-optimally solvable to MIP solvers such as CPLEX and Lingo. Xie et al. (2018) used a set of linear constraints to surrogate the Euclidean distance metric within a negative controllable error range. For two points A (x_1, y_1) and B (x_2, y_2) on a two-dimension plane, the Euclidean distance function can expressed as shown in Eq. (18).

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{18}$$

The lower bound, d', of the Euclidean distance d can be restricted by the following linear constraints (Xie *et al.*, 2018).

$$d' \geq |x_1 - x_2| \cdot \sin(k\theta) + |y_1 - y_2| \cdot \cos(k\theta) \qquad 1 \leq k \leq n, \tag{19}$$

where θ is a constant angel calculated by $\theta = \arccos(1 - 4\varepsilon + 2\varepsilon^2)$, $n = \left\lceil \dfrac{\pi}{2\theta} \right\rceil$, $\lceil \Box \rceil$ represents the smallest integer that is greater than or equal to $\Box$, and ε is a given parameter to control the bounded d' deviating from the true Euclidean distance d within a given percentage ε, i.e., $|\dfrac{d' - d}{d}| \times 100\% \leq \varepsilon$.

However, the minimal d' bounded by Eq. (19) is always smaller than the true Euclidean distance d, though the gap can be as small as possible controlled by ε. This may cause the solution technically infeasible in some cases where distances smaller than the threshold are not allowed. For example, if the Euclidean distance is 10 and we let $\varepsilon = 0.1\%$, then the resulted bounded distance can be minimally 9.99 in an extreme case.

For above reason, this paper provides an alternative constraint, expressed in Eq. (20) as follows, that can be used to bound a minimum value, d'', that is no less than the true Euclidean distance d within a given maximum error

range ε.

$$d'' \geq |x_1 - x_2| \cdot \frac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} + |y_1 - y_2| \cdot \frac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} \qquad 1 \leq k \leq n, \qquad (20)$$

where θ is a constant angel calculated by $\theta = 2\arccos(\dfrac{1}{1+\varepsilon})$, $n = \left\lceil \dfrac{\pi}{2\theta} \right\rceil$, ε is the given parameter that guarantees

$|\dfrac{d'' - d}{d}| \times 100\% \leq \varepsilon$, and d is the true Euclidean distance. Detailed proof of Eq.(20) is provided in Appendix.

In Table 1, some example data of n, θ, and ε calculated by both Eqs.(19) and (20) are provided for readers of direct use. It can be observed that n and θ always have very close values corresponding to different ε under Eqs. (19) and (20), so they can be considered as the same level of complexity when used in model formulations.

Table 1. Example data of n, θ, and ε

	ε	10%	5%	2%	1%	0.5%	0.2%	0.1%	0.05%	0.02%	0.01%	0.005%	0.001%
Using	θ	0.902	0.635	0.401	0.283	0.2000	0.1265	0.0895	0.0633	0.0400	0.0283	0.0200	0.00894
Eq.(19)	n	2	3	4	6	8	13	18	25	40	56	79	176
Using	θ	0.859	0.620	0.397	0.282	0.1996	0.1264	0.0894	0.0632	0.0400	0.0283	0.0200	0.00894
Eq.(20)	n	2	3	4	6	8	13	18	25	40	56	79	176

Consequently, by using the surrogate constraints in Eq. (18) and a given accuracy requirement ε, the nonlinear Euclidean distances in Constraints (2), (6), (11), and (13) can be converted into the following linear Constraints (21), (22), (23), and (24), respectively, as follows.

$$\begin{cases} d_{ij}^x \geq x_i - X_j & \forall i \in N, j \in V \\ d_{ij}^x \leq X_j - x_i & \forall i \in N, j \in V \\ d_{ij}^y \geq y_i - Y_j & \forall i \in N, j \in V \\ d_{ij}^y \leq Y_j - y_i & \forall i \in N, j \in V \\ d_{ij} \geq d_{ij}^x \dfrac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} + d_{ij}^y \dfrac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} & \forall i \in N, j \in V, k = 1, 2..., n \end{cases} \qquad (21)$$

$$\begin{cases} D_j^x \geq X_j - X_j & \forall j \in V \\ D_j^x \leq X_j - X_j & \forall j \in V \\ D_j^y \geq Y_j - Y_j & \forall j \in V \\ D_j^y \leq Y_j - Y_j & \forall j \in V \\ D_j \geq D_j^x \dfrac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} + D_j^y \dfrac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} & \forall j \in V, k = 1, 2..., n \end{cases} \qquad (22)$$

$$
\begin{cases}
d_{ijt}^{x} \geq x_{it} - X_{jt} & \forall i \in N_t, j \in V_t, t \in T \\
d_{ijt}^{x} \leq X_j - x_i & \forall i \in N_t; j \in V_t, t \in T \\
d_{ijt}^{y} \geq y_{it} - Y_{jt} & \forall i \in N_t; j \in V_t, t \in T \\
d_{ijt}^{y} \leq Y_j - y_i & \forall i \in N_t, j \in V_t, t \in T \\
d_{ijt} \geq d_{ijt}^{x} \dfrac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} + d_{ijt}^{y} \dfrac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} & \forall i \in N_t, j \in V_t, t \in T, k = 1, 2..., n
\end{cases}
\tag{23}
$$

$$
\begin{cases}
D_{jt}^{x} \geq X_{jt} - X_{j,t-1} & \forall j \in V_t, t \in T \\
D_{jt}^{x} \leq X_{j,t-1} - X_{jt} & \forall j \in V_t, t \in T \\
D_{jt}^{y} \geq Y_{jt} - Y_{j,t-1} & \forall j \in V_t, t \in T \\
D_{jt}^{y} \leq Y_{j,t-1} - Y_{jt} & \forall j \in V_t, t \in T \\
D_{jt} \geq D_{jt}^{x} \dfrac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} + D_{jt}^{y} \dfrac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} & \forall j \in V_t, t \in T, k = 1, 2..., n
\end{cases}
\tag{24}
$$

3. Solution approaches

3.1 A two-stage optimization framework

We adopt the MILP-based hierarchical framework introduced by Xiao et al. (2015) to optimize the two hierarchical objective functions of the multi-period C-MCLP model with Pareto optimization. This is because the SCR is always considered as a more important objective than TMD so it should be maximized, while the TMD of CVs is being minimized based on the optimized SCR. The introduced hierarchical framework hereby includes the following two stages. The first step is to execute optimization on the SCR, i.e., the signal coverage rate, with no requirement on the TMD. After a maximized objective value on the *SCR*, which is denoted as *SCR**, is obtained, the following Constraint (25) is added to the MILP model to set a minimum requirement for the SCR objective.

$$
SCR^{*} \leq \sum_{i \in N_t, t \in T} e_{it}^{'} / |N_t|
\tag{25}
$$

Next, the TMD objective is set as the objective function to solve the MILP model with a re-optimized solution. Hence, the finally obtained solution is a Pareto optimal solution that was optimized on both SCR and TMD objectives. The two-stage optimization framework is outlined in Fig.5 as follows.

---

1) Set SCR as the objective function without objective function TMD

2) Solve the C-MCLP model to obtain an optimized SCR value denoted as SCR*

3) Set TMD as the objective function without objective function SCR

4) Add Constraint (25) to the model

5) Solve the model to obtain the Pareto optimal solution on the two objectives SCR and TMD

Fig. 5. A two-stage optimization framework

The hierarchical setting of two objective function in Fig.5 is special case of multi-objective optimization

searching for the Pareto optimal solutions. More information on the methods and applications of multi-objective optimization can be seen in recent reviews carried out by Cui et al. (2017), Konak et al. (2006), and Zitzler et al. (2000). It should also be noted that the first stage of optimizing the SCR is much more time-consuming than the second stage of optimizing the TMD. Our pilot experiments indicated that the first stage might require several hours of CPU time whereas the second stage required only a few seconds for large-sized instances. Therefore, as introduced in the following sub-sections, the algorithms were designed mainly for the first stage.

3.2 The k-means algorithms

The k-means algorithm (Hartigan A. J., 1979) is an efficient algorithm for rapid clustering of data objects into a given number of clusters, to minimize the inner-distances of the clusters. The k-means algorithm (or its variants) is typically used as efficient solution approach for various facility location problems (Xiao et al., 2017, You et al., 2019). The k-means algorithm was adopted to partition the RTs into the same number of CVs. The RTs with distance similarities were grouped into the same cluster and therefore served by one CV. Thus, the cluster centers were set as the target locations of the CVs.

The k-means algorithm starts from a random initialization of CVs' locations as the seed centers. Then, repeat to partition each RT to the closest center . After that, the cluster center is replaced by the mean center of the members in that cluster. These two steps are repeated until the objective function no longer changes. We consider two types of objective function, i.e., (1) the total inner-distances of all clusters and (2) the total number of covered RTs, to form two types of k-means algorithm denoted as k-means-I and k-means-II, respectively. The framework of the k-means-I/II algorithms for the C-MCLP is outlined as follows:

The k-means-I/II algorithm:

(1) Set the cluster number k as the same number of CVs

(2) Randomly generate k locations as the initial locations of CVs

(3) Calculate the distances between the RTs and CVs

(4) Assign each RT to its closest CV

(5) Calculate the mean centers of the clustered RTs as the new location for CVs

(6) Calculate the objective function: (I) the total inner-distances or (II) the total number of covered RTs

(7) Repeat steps (3), (4), and (5) until there is no further change to the objective function

(8) For each CV, rank its assigned RTs with descending distance order

(9) For each CV, pick its member RTs as the covered nodes from the nearest to furthest, until the capacity is reached or the maximum covering radius is exceeded

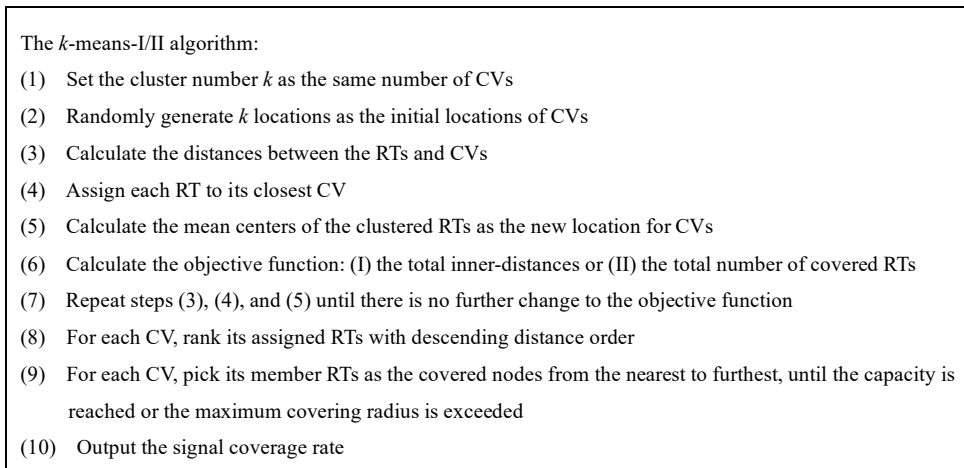(10) Output the signal coverage rate

---

Fig. 6. Framework of the k-means algorithm

3.3 A MILP-based fix-and-optimize heuristic approach

For large-sized C-MCLP instances, an efficient heuristic approach was developed to obtain near-optimal solutions and achieve high computational efficiencies. The proposed approach is referred to as the MILP-based dynamic iterative partial optimization (MILP-DIPO for short), and it is based on the MILP-based neighborhood searching algorithms developed by Xiao et al. (2016, 2019a, 2019b). The underlying principle is as follows. For a large-sized complex problem with multiple decision variables, if a complete optimization over all variables cannot be achieved within an acceptable CPU time, then partial optimizations (POs) can only be applied to a small portion

of the selected decision variables, while the most other variables are set with given values. Thus, the selected decision variables can be efficiently optimized within a very short CPU time. Then, POs are applied repeatedly to different portions of randomly selected variables, until no further improvement can be made to the objective function after a given number of continuous attempts. The main framework of the MILP-DIPO algorithm is outlined in Fig.7 as follows:
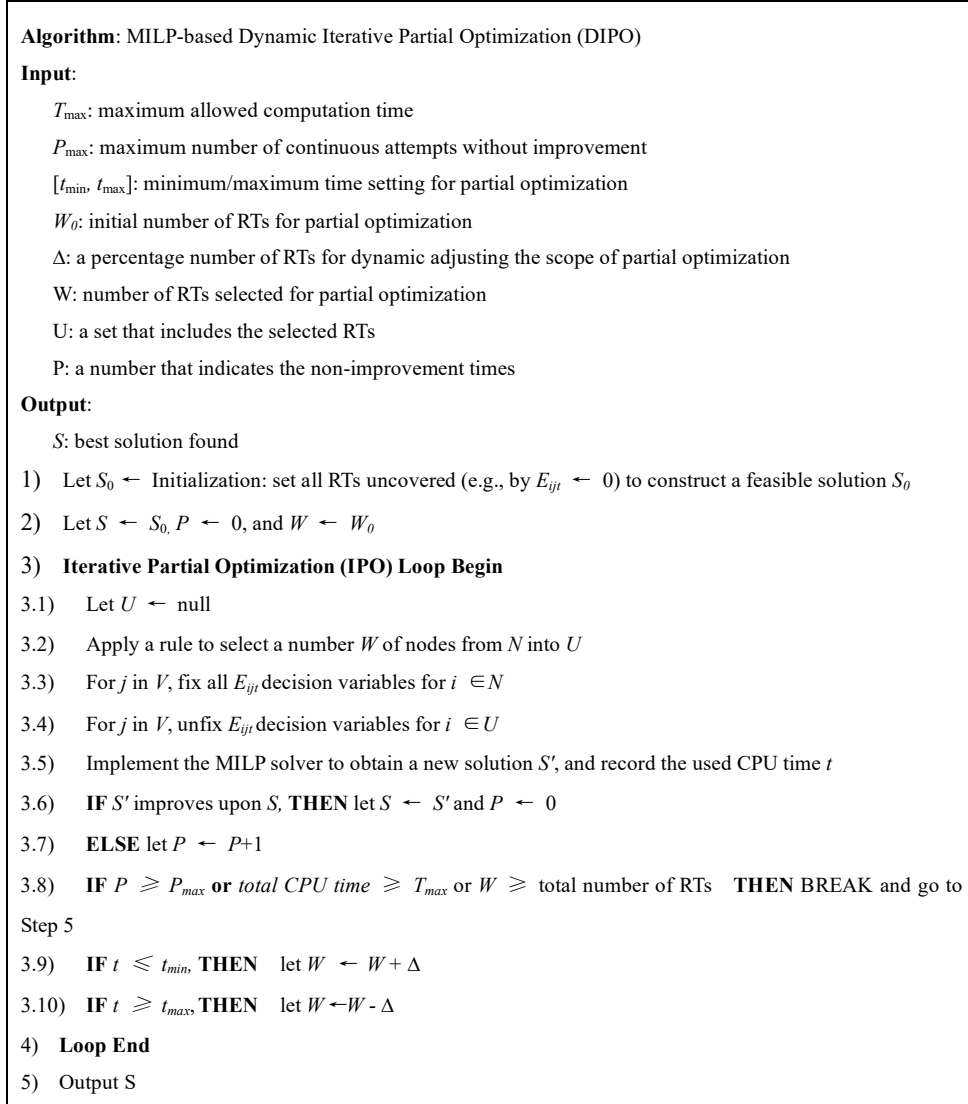

Algorithm: MILP-based Dynamic Iterative Partial Optimization (DIPO)

Input:

    T_{max}: maximum allowed computation time

    P_{max}: maximum number of continuous attempts without improvement

    $[t_{min}, t_{max}]$: minimum/maximum time setting for partial optimization

    W_0: initial number of RTs for partial optimization

    Δ: a percentage number of RTs for dynamic adjusting the scope of partial optimization

    W: number of RTs selected for partial optimization

    U: a set that includes the selected RTs

    P: a number that indicates the non-improvement times

Output:

    S: best solution found

1)   Let S_0 ← Initialization: set all RTs uncovered (e.g., by E_{ijt} ← 0) to construct a feasible solution S_0

2)   Let S ← S_0, P ← 0, and W ← W_0

3)   **Iterative Partial Optimization (IPO) Loop Begin**

3.1)   Let U ← null

3.2)   Apply a rule to select a number W of nodes from N into U

3.3)   For j in V, fix all E_{ijt} decision variables for $i \in N$

3.4)   For j in V, unfix E_{ijt} decision variables for $i \in U$

3.5)   Implement the MILP solver to obtain a new solution S', and record the used CPU time t

3.6)   **IF** S' improves upon S, **THEN** let S ← S' and P ← 0

3.7)   **ELSE** let P ← $P+1$

3.8)   **IF** $P \geqslant P_{max}$ **or** *total CPU time* $\geqslant T_{max}$ or $W \geqslant$ total number of RTs   **THEN** BREAK and go to Step 5

3.9)   **IF** $t \leqslant t_{min}$, **THEN**   let W ← $W + \Delta$

3.10)   **IF** $t \geqslant t_{max}$, **THEN**   let W ← $W - \Delta$

4)   **Loop End**

5)   Output S

Fig. 7. Main framework of the MILP-DIPO algorithm

As shown in Fig. 7, the MILP-DIPO algorithm starts with an initial solution S_0, a simple feasible solution that covers no RTs, and then S_0 is set as the incumbent solution C. The non-improvement counter P and the number of nodes to be selected W are initialized in Step 2. Thereafter, a loop is launched in Step 3 to implement the iterative Partial Optimization (IPO), which includes Steps 3.1–3.10. In Step 3.2, a set of RTs is selected according to the time frequency priority (TFP) rule (Xiao et al., 2019a, 2019b) under which the nodes having been selected fewer times are always assigned higher possibility to be selected. A frequency recorder, e.g., r_i, is used to record the times of being selected for each node i. Thus, the probability of selecting node i under the TFP rule is calculated as

$p_i = \dfrac{1}{(r_i + 1)R}$, where $R = \sum\limits_{j \in N} \dfrac{1}{(r_j + 1)}$. For example, if there are three nodes 1, 2, and 3, and they were selected for 2,

0, and 3 times respectively, then their probability of being selected are calculated as 0.21, 0.63, and 0.16 respectively. Notably, for solving multi-period C-MCLP, the selected RTs should come from all the periods. In Steps 3.3 and 3.4, the selected nodes (RTs) are unfixed for their assignments, whereas the remainder are fixed. The CPLEX solver is then applied to optimize the selected nodes (Step 3.5). In Steps 3.6 and 3.7, the new obtained solution is judged for if it has been improved upon incumbent solution. If yes, then it is accepted as the new incumbent solution, and the non-improvement counter P is reset to zero; otherwise, P is incremented by one. The loop is stopped on one of the three conditions: (1) no further improvement can be made on the incumbent solution after a number P_{max} of continuous attempts, (2) the total elapsed CPU time is greater than a given limited time T_{max}, or (3) all the nodes were simultaneously selected for optimization (Step 3.8). In Steps 3.9–3.10, the number W is adjusted dynamically to control the solving time used by CPLEX between t_{min} and t_{max}. The number W is increased by ∇ percentage of the total RTs when the previous CPU time is shorter than t_{min}, or decreased by ∇ percentage when the CPU time is longer than t_{max}. Finally, the resulted incumbent solution S is the best solution found; thus, it is the output of the algorithm. It should be noted that when applying the MILP-DIPO algorithm to the solution of large-scale multi-period C-MCLP, the partial optimizations should be implemented on RTs selected from all the periods (Step 3.2).

In the IPO loop, the MILP solver (CPLEX) consumes most of the CPU time (Step 3.5); thus, the computational complexity of the algorithm is dependent on how many times the IPO is implemented by the MILP solver, which is controlled by an input parameter P_{max}. The average CPU time, denoted as R, used by CPLEX for each round of partial optimization and restricted between t_{min} and t_{max}, is determined by the number of selected nodes W. Thus, the algorithm's complexity level can be estimated as $O(P_{max} \times W \times R)$.

4. Simulating experiments

The simulation experiments were conducted on a Linux PC server with two 2.90 GHz Intel Xeon (R) CPUs (32 threads) and 128-GB RAM. The MIP solver AMPL/CPLEX (version 12.6.0.1) was used to solve the tested instances.

4.1 The experimental data

Several groups of datasets were generated to simulate the search and rescue operations in natural disaster stricken areas of different scales. The first group contained three instances with rescue team numbers 20, 50, and 100, respectively; and their actions were conducted in a specified square region with dimensions of 100×100. Here, one unit of distance was set to be 100 m. The second group contained four instances with rescue team numbers 200, 400, 600, and 1000, respectively. The first three instances had a specified square region with dimensions of 200×200, and the final instance (with 1000 rescue teams) had a large region with dimensions of 300×300. The targeting coordinates of the rescue teams were randomly generated within the regions, with a random deviation Δ_i in [0, 5] for each RT $i \in N$. The number of CVs was set as 3, 3, 6, 12, 23, 34, and 56 with respect to team number of 20, 50, 100, 200, 400, 600, and 1000, respectively. The covering radius of the CVs was set as 15 (1.5 km), the capacity of each CV was set as 18, and the maximal moved distance within one period was set as $L^{max}=30$.

For the multi-period C-MCLP, five periods/phases of rescuing operations were considered with an increasing numbers of rescue teams that were assumed to be continuously arriving and joining the rescue operations. Two test instances were generated to test the performances of the proposed model and heuristic algorithm. The first test instance simulated a five-period case with 20, 25, 30, 35, and 40 available RTs in five periods, to search for survivors in a 100x100 region, and the number of CVs was set as 3, with parameters C=18, D^{max}=15, and L^{max}=30. The second test instance was a larger multi-period C-MCLP instances with 160, 180, 200, 220, and 240 available RTs in five periods, to search for survivors in a 300x300 region. The number of available CVs was set as 15, with parameters C=18, D^{max}=25, and L^{max}=50. In these two instances, the targeting coordinates of RTs in each period were generated by the simulation of a planned search actions, wherein all the teams started from one entry side and gradually searched across the whole region.

In all the calculations, the Euclidean distance linearization parameter ε was set as 0.1%, which indicates that the approximate error rate for the Euclidean distance linearization was within 0.1%.

4.2 Experiments on single-period C-MCLP

For the small-sized instances (the number of rescue teams was 20, 50, 100, and 200), the CPLEX solver was directly used to obtain optimal solutions. Moreover, we also applied the k-means-I, k-means-II, and MILP-DIPO algorithms to obtain 10 heuristic solutions for each instance. The parameters settings for DIPO were T_{max} = 300, P_{max} = 50, t_{min} = 1, t_{max} = 1, W_0 = 10, and Δ = 1%. The computational results are listed in Table 2, which reveal the differences between these methods with respect to the solution quality (SCR) and computational efficiency (solution time). Columns NRT and NCV represent the number of rescue teams and number of CVs, respectively. As can be seen in Table 2, the resulted SCR values by CPLEX are optimal. The MILP-DIPO algorithm could also obtain the optimal solutions in 10 runs for all test instances. Columns *MILP-DIPO best* and *MILP-DIPO avg* indicate the best solution and average solution found in 10 runs, respectively. The k-means algorithms delivered quite worse solutions, without finding any optimal solution. The solution time required by the CPLEX, k-means, and MILP-DIPO could be considered negligible when the number of rescue teams was less than 100. When the number of rescue teams increased, CPLEX required significantly more time to obtain the optimal solution. This can be attributed to the combinatorial nature of the C-MCLP, which leads to an exponentially growth of time in accordance with an increase in the problem size. The MILP-DIPO showed both good solution qualities and relatively high computational efficiencies. The k-means algorithms have much higher computational efficiency but lower solution quality than other two.

Table 2. Comparison of solutions obtained by different methods.

NRT	NCV	SCR					Time (s)		
		CPLEX	k-means-I	k-means-II	MILP-DIPO best	MILP-DIPO avg	CPLEX	k-means	DIPO
20	3	**20/20**	17.7/20	17.2/20	**20/20**	**20/20**	<1	<1	<1
50	3	**21/50**	11.0/50	10.3/50	**21/50**	19.5/50	<1	<1	1.4
100	6	**51/100**	39.5/100	35.9/100	**51/100**	49.9/100	3.0	<1	16.2
100	12	**57/100**	37.7/100	36.2/100	**57/57**	**57/57**	7.9	<1	21.5
120	12	**66/120**	44.0/120	43.9/120	**66/120**	65.5/120	14.9	<1	32.1

140	12	**74/140**	47.8/140	49.2/140	**74/140**	73.3/140	17.3	<1	33.7
160	12	**83/160**	56.1/160	55.5/160	**83/160**	83/160	33.3	<1	38.5
180	12	**93/180**	62.5/180	61.4/180	**93/180**	92.4/180	42.2	<1	38.5
200	10	**84/200**	67.9/200	62.5/200	**84/200**	83.5/200	49.1	<1	35.3
200	12	**100/200**	52.5/200	50.4/200	**100/200**	98.9/200	77.1	<1	46.6
200	14	**113/200**	79.3/200	75.7/200	**113/200**	111.8/200	182.1	<1	52.3
200	16	**130/200**	90.1/200	90.4/200	**130/200**	128.2/200	998.8	<1	58.4

Note: **bold-face** indicates the optimal values.

Fig. 8 presents three optimal solutions obtained by CPLEX for the instances wherein NRT = 20, 50, and 100, respectively, to provide readers a graphic view of the problem structures and solutions. As shown in the figure, CVs move from an initial location to the optimized targeting locations, the center of the circles, to maximize the coverage of RTs. In the first image, three CVs were deployed to serve 20 rescue teams. Different colors were used to plot the rescue teams that were covered by different CVs; wherein the red, green, and blue colors represent the rescue teams covered by CV1, CV2, and CV3, respectively. The pink color represents the rescue teams that were not covered by CVs. In the second and third images, the detailed results were illustrated for the instances wherein NRT = 50 and NRT = 100, respectively.
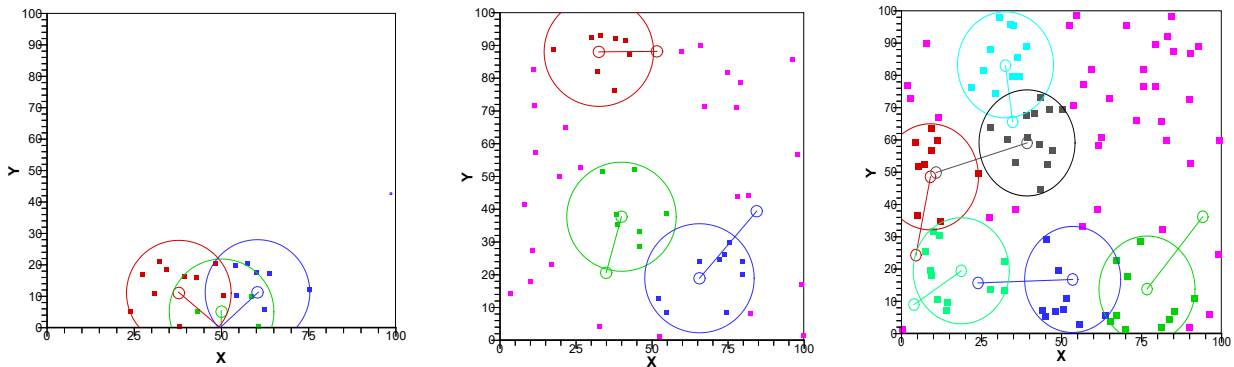

Fig. 8. Detailed solutions for NRT = 20, 50, and 100

Next, we tested the performance of our model with respect to different values for parameters ε and L^{max}. First, we used CPLEX to solve two instances, NRT/NCV = 100/12 and NRT/NCV = 200/10, and obtained the optimal solutions with respect to different values of ε from 5% to 0.01% and with fixed L^{max}=30. The computational results are shown in Table 3. The results reveal that as ε varied, both the SCR objective values and the solving time varied. In general, more computational time is required to increase the accuracy of the parameter ε. In practice, control of the error range within 0.1% (ε = 0.1%) should meet the requirements of most practical applications. Next, we used CPLEX to solve these two instances with fixed ε = 0.1% and with respect to different values of L^{max} increasing from 20 to 50. Table 4 presents the performance of the model. As can be seen from the table, the SCR and TMD increased steadily as the L^{max} increased, and more computational time was required for a larger L^{max}.

Table 3. Solution efficiencies under different values of ε

E	Θ	n	NRT=100, NCV=12		NRT=200, NCV=10	
			SCR	Time (s)	SCR	Time (s)
5%	0.620	3	55/100	2.7	82/200	15.1

1%	0.282	6	56/100	2.5	83/200	28.1
0.5%	0.1996	8	57/100	2.0	83/200	30.6
0.1%	0.0894	18	57/100	8.8	84/200	49.8
0.05%	0.0632	25	57/100	4.1	84/200	54.2
0.01%	0.0283	56	57/100	6.1	84/200	107.9

Table 4. Solution efficiencies under different values of L^{max}

| L^{max} | NRT=100, NCV=12 | | | NRT=200, NCV=10 | | |
	SCR	TMD	Time (s)	SCR	TMD	Time (s)
20	59	160.3	0.9	94	140.4	1.5
25	61	211.6	2.8	96	181.0	30.5
30	65	235.8	21.7	102	222.4	51.9
35	65	232.1	46.2	104	251.8	180.6
40	67	264.5	929.2	107	262.5	554.0
45	70	307.1	1852.3	107	262.5	3600
50	70	307.1	3600	109	285.9	3600

Thereafter, we tested the proposed MILP-DIPO heuristic algorithm with the large-sized problem instances of NRT = 400, 600, and 1000. Given that the CPLEX solver was unable to deliver the optimal solutions for these instances within an acceptable CPU time-range, e.g., 300 s, in emergency situations, we used the DIPO algorithm, as shown in Fig. 9, to obtain heuristic solutions. The parameters settings for MILP-DIPO were $T_{max} = 300$, $P_{max} = 50$, $t_{min} = 1$, $t_{max} = 1$, $W_0 = 100$, and $\Delta = 1\%$. The CPLEX (with a time limit of 300 s) and k-means algorithm were also used to solve these instances and obtain comparative solutions. We repeatedly ran the DIPO algorithm for 10 times, and the average/best/worst solutions of the 10 runs were obtained for each instance. The obtained solutions were compared, as shown in Table 5.

In Table 5, the number of RTs and CVs are listed in the first two columns, the objective values are compared in the third, fourth, and fifth columns, and the computational times are compared in the last column. With respect to the SCR objective, the MILP-DIPO algorithm was proved to be the best. In particular, even the worst result yielded by MILP-DIPO was better than or close to the best results yielded by the other two methods. The best SCR optimized by MILP-DIPO are higher than the solutions of the CPLEX by 2.3%, 12.7%, and 6.5% for the instances of NRT = 400, 600, and 1000, respectively. The solutions are also higher than the solutions of the k-means algorithm by 55.2%, 30.5%, and 38.1% for the instances of NRT = 400, 600, and 1000, respectively. With respect to the computational efficiency, the k-means was the best, as it required less than 1s to compute. However, given that the primary objective is the maximization of the SCR, the MILP-DIPO algorithm can be considered as more practical than the other two algorithms. Fig. 9 presents a comparison of the SCR objective values of the solutions obtained using the three methods (CPLEX, k-means-I, and MILP-IPDO) with respect to the three tested instances.

Table 5. Comparison of solutions obtained using MILP-DIPO, CPLEX, and k-means algorithms

| NRT | NCV | CPLEX | k-means | | MILP-DIPO | | | Time (s) | | |
			I	II	Best	Worst	AVG	CPLEX	k-means	MILP-DIPO
400	23	220/400	145.0/400	144.3/400	**225/400**	221/400	222.8/400	300	< 1	124.9
600	34	370/600	319.6/600	314.6/600	**417/600**	398/600	405/600	300	< 1	130.0
1000	56	496/1000	382.3/1000	380.5/1000	**528/1000**	493/1000	504.5/1000	300	3.2	198.2

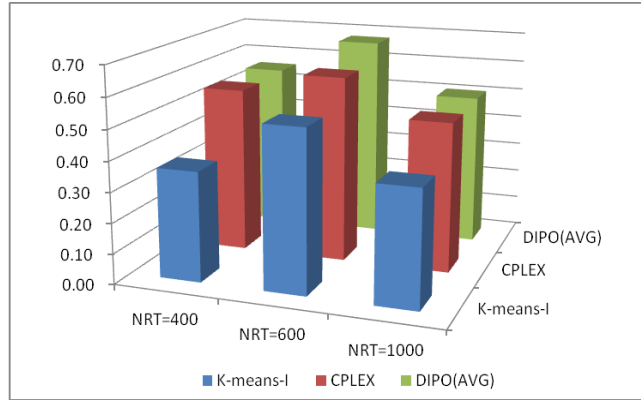Note: bold fact indicates the best values.

Fig. 9. Comparison of the solutions with respect to different methods

4.3 Experiments on multi-period C-MCLP

The multi-period C-MCLP model is applicable to a more complex disaster rescue case, and it allows for RTs to join the rescue actions continuously as previous RTs move forward. All RTs were assigned different targeting areas to carry out rescue actions in different periods. First, we applied CPLEX to solve the small multi-period instance with RT numbers 20, 25, 30, 35, and 40 in five periods. The optimal solution was obtained in 43 seconds and shown in Table 6, where the columns SCR and TMD represent the objective function values achieved by the multi-period C-MCLP model. Table 7 presents the optimal CV coordinates in different periods.

Table 6. Solution of the small multi-period instance

t	NCV	SCR	TMD	Time (s)
1	3	20/20	36.83	--
2	3	23/25	67.07	--
3	3	21/30	50.31	--
4	3	28/35	60.23	--
5	3	33/40	71.54	--
Total		125/150	285.97	43

Table 7. Optimized coordinates of CVs in multiple periods

	Initial	Period = 1	Period = 2	Period = 3	Period = 4	Period = 5
	(50, 0)	(37.78, 11.04)	(24.68, 23.03)	(21.97, 50.89)	(21.93, 64.81)	(15.20, 89.62)
(X_j, Y_j)	(50, 0)	(50.03, 5.09)	(50.03, 35.21)	(51.37, 45.29)	(48.65, 70.90)	(46.56, 86.61)
	(50, 0)	(60.33, 11.25)	(73.30, 25.39)	(81.51, 34.34)	(82.64, 54.86)	(76.93, 84.3)

The movements of the RTs and CVs are plotted in Fig. 10, where squares represent the RTs, circles represent the CVs, and nodes in same color represent a CV's coverage. The filled squares indicate the RTs covered by a CV whereas the empty squares indicate the RTs not covered by any CVs. It can be observed that at the beginning all RTs were covered by CVs as they were gathering around the initial position. However, as the search actions went on, more and more RTs became uncovered as the RTs start to disperse in the whole search region and new RTs were continuing to join in. The CVs were scheduled with the shortest moved distance to provide as a maximized SCR as possible. The assignments of RTs to CVs were dynamically determined based on the locations of the RTs and CVs. That means a RT could be connected to one CV in one period and it might be connected to another one in the next period. When a RT is covered by multiple CVs, it just connects to one of them and occupies only one channel. This

20

figure presents a planned search and rescue process starting from the initial point and crossing the entire area, with more RTs joining over time. It simulates the real need from the rescuing scene that seeks primarily to cover as more as possible the number of RTs in all periods (as the primary objective), and secondarily to shorten the total moved distance of the CVs (as the secondary objective).
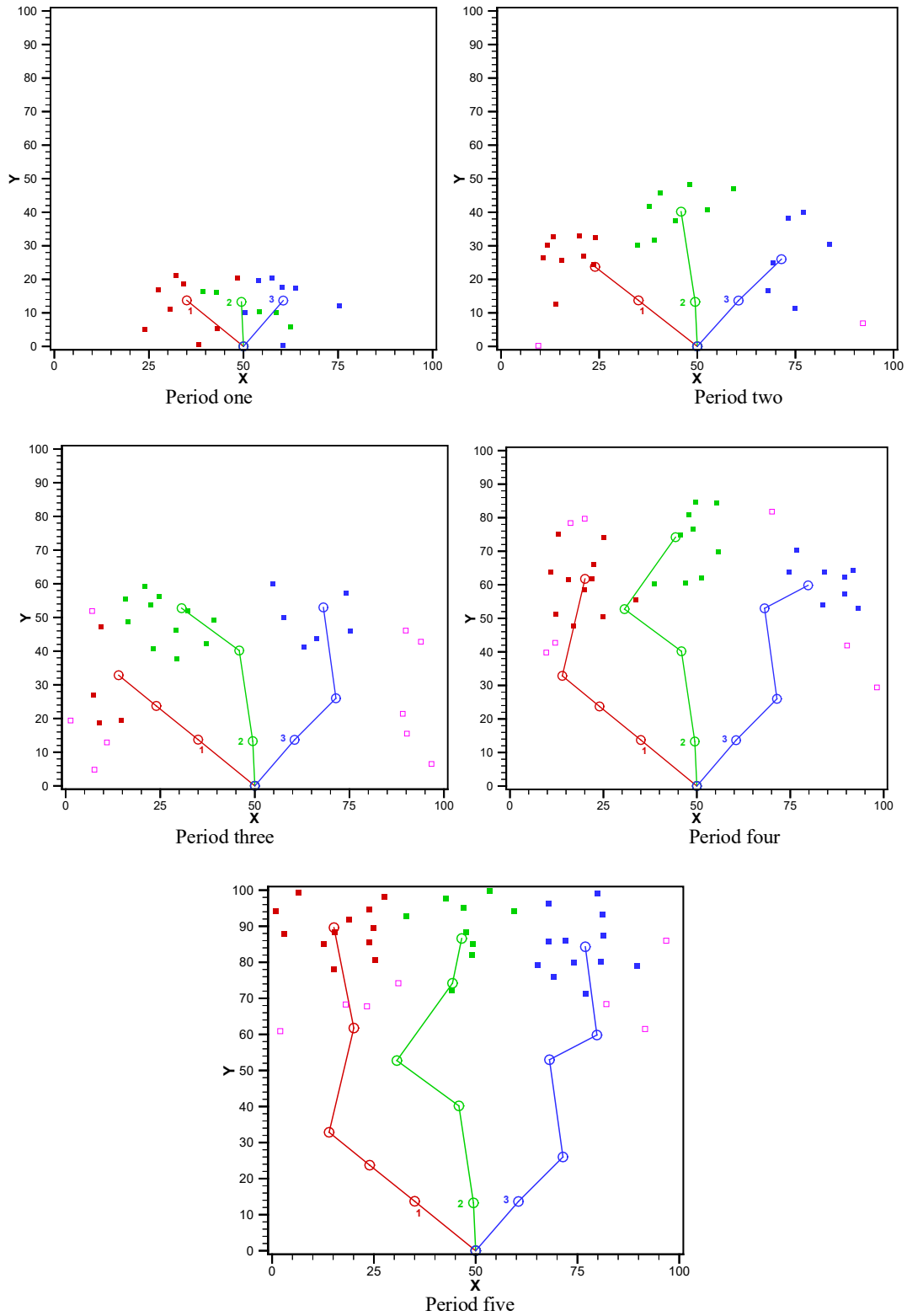

Fig. 10. Movements of RTs and CVs in five periods

It should be noted that the CVs are always assumed to be able to move faster than RTs, which is determined by the setting of parameter L^{max}; otherwise, the period coverage rates of the obtained solution will show a gradually decreasing trend to zero as the time period goes on. When L^{max} is large enough, the optimized SCR in a period will be mainly determined by the distribution of RTs in that period, and equally affected by the distributions of RTs before and after that period. Without considering the moved distance of CVs in the first period from their initial locations, the multi-period solution would be the same if the RTs are supposed to move reversely from the last period to the first one.

Next, we tested the MILP model and the proposed MILP-DIPO algorithm on the large multi-period C-MCLP instance. This instance has 160, 180, 200, 220, and 240 available RTs, totally 1000, in five periods. Four methods, i.e., CPLEX, k-means-I, k-means-II, and DIPO, were applied to solve this test instance and compare the results. The CPLEX solver was set with a 3600 second time limit, and the MILP-DIPO algorithm was set with parameters $T_{max} = 300$, $P_{max} = 50$, $t_{min} = 1$, $t_{max} = 1$, $W_0 = 100$, and $\Delta = 1\%$. All methods (except CPLEX) repeated 10 runs to obtain 10 random solutions. The results are compared in Table 8 as follow.

Table 8. solution comparison of the large multi-period C-MCLP instance

Method	SCR				TMD			
	Best	Worst	AVG	Time (s)	Best	Worst	AVG	Time (s)
k-means-I	546	489	523.5	<1	--	--	--	<1
k-means-II	536	488	513.6	<1	--	--	--	<1
CPLEX	496	--	--	3600	3426.6	--	--	2.5
MILP-DIPO	615	587	597.9	179.7	3316.1	3160.2	3253.3	1.2

As seen in Table 8, in the first stage for maximizing the SCR, the MILP-DIPO algorithm used averagely 179.7 second and obtained the best average solution 597.9, as well as the best solution 615 among all methods under comparison. The k-means-I and II ranked the second and third ones, respectively, while the CPLEX solver is the last. In the second stage (for CPLEX and MILP-DIPO only), the TMD were obtained efficiently when the primary objective was fixed to a known value, with 2.5 and 1.2 seconds for CPLEX and MILP-DIPO, respectively. This indicated that computational complexity of the C-MCLP lies mainly on the first stage for maximizing the SCR. Moreover, in some particular cases where CVs are relatively rare resources, the number of CVs can be also an secondary objective function to be minimized, together with the SCR and TMD, forming a multi-objective optimization problem for studying.

5. Conclusion

The C-MCLP plays a significant role in efficient searching and rescuing survivors in natural-disaster scenes, such as earthquakes, floods, and storms, where communication system is often disrupted. In this study, we formulated the C-MCLP with two mixed-integer mathematical linear programming models and two hierarchical objective functions, e.g., the maximal signal-coverage and the shortest moved distance, to provide high-quality solutions at high computational efficiencies. The proposed models can be efficiently and optimally solved for small- and medium-sized problem instances by directly using commercial MILP solvers such as CPLEX and Lingo. For the large-scale problems and the multi-period cases, we developed an efficient heuristic algorithm, referred to

as MILP-DIPO, to solve the problem with near-optimal solutions rapidly. The experimental results and comparison with the traditional *k*-means algorithm revealed that the proposed models and algorithms work efficiently with better solutions obtained in a limited time setting, e.g., within 300 s, which is a suitable time limit for such emergency situations.

Besides natural-disaster rescues, the proposed models and solution method can also be extended to other applications such as warehouse/factory location optimization in logistic distribution system, distribution coverage of emergency rescue resources in urban areas, and orbit optimization of a group of communication satellites. The current models are mainly tested on academic case studies, but yet to be be tested in an real situation. Future study and improvement on the theoretical models and solution approach of the C-MCLP include: (1) considering dynamic connection reliability affected by environmental terrain differences, (2) considering robust optimization of senarios with uncertain movements of rescue teams, or (3) developing a more efficient heuristic algorithm for large-sized problem instances.

References

[1]   Akella R. M., Batta R., Delmelle M. E., Rogerson A. P., Blatt A., Wilson G., 2007. Base station location and channel allocation in a cellular network with emergency coverage requirements. European Journal of Operational Research 164(2), 301-323.

[2]   Below R., Wallemacq P., 2018. Annual Disater Statistical Review 2017, CRED https://www.emdat.be/publications [accessed 2019/2/25]

[3]   Brandeau L. M., Chiu S. S., 1989. An overview of representative problems in location research. Management Science, 35(6), 645-674.

[4]   Bhattacharya B. B., Nandy S. C., 2013. New variations of the maximum coverage facility location problem. European Journal of Operational Research 224(3), 477-485.

[5]   Church L. R., Revelle C., 1974. The maximal covering location problem. Papers of the Regional Science Association, 32(1), 101-118.

[6]   Corrêa A. de. F., Lorena N. A. L., Ribeiro M. G., 2009. A decomposition approach for the probabilistic maximal covering location-allocation problem. Computers & Operations Research 36(10), 2729-2739.

[7]   Cui Y., Geng Z., Zhu Q., Han Y., 2017. Review: Multi-objective optimization methods and application in energy saving. Energy 125, 681-704.

[8]   Davari S., Zarandi F. H. M., Hemmati A., 2011. Maximal covering location problem (MCLP) with fuzzy travel times. Expert Systems with Applications 38(12), 14535-14541.

[9]   Davari S., Zarandi F. H. M., Turksen B. I., 2013. A greedy variable neighborhood search heuristic for the maximal covering location problem with fuzzy coverage radii. Knowledge-Based Systems 41(41), 68-76.

[10]   Dell'Olmo P., Ricciardi N., Sgalambro A., 2014. A Multiperiod Maximal Covering Location Model for the Optimal Location of Intersection Safety Cameras on an Urban Traffic Network. Procedia - Social and Behavioral Sciences 108,106-117.

[11]   EMDAT, 2019. OFDA/CRED International Disaster Database, Université catholique de Louvain – Brussels – Belgium.

[12]   Erdemir T. E., Batta R., Rogerson A. P., Blatt A., Flanigan M., 2010. Joint ground and air emergency medical services coverage models: A greedy heuristic solution approach. European Journal of Operational Research 207(2), 736-749.

[13]   Friedrich, J. C., 1929. Alfred Weber's theory of the location of industries. American Journal of Sociology 35(5), 853.

[14]   Galinier P., Hertz A., 2007. Solution techniques for the Large Set Covering Problem. Discrete Applied Mathematics 155(3), 312-326.

[15]   Gao N., Nie G., 2015. Study on Earthquake Emergency Relief Efficiency. Journal of Catastrophology 30(2), 158-161.

[16]   Guzmán C. V., Pelta A. D., Verdegay, L. J., 2016. An approach for solving maximal covering location problems with fuzzy constraints. International Journal of Computational Intelligence Systems 9(4), 734-744.

[17]   Hartigan A. J., Wong A. M., 1979. Algorithm AS 136: A K-means clustering algorithm. Journal of the Royal Statistical Society 28(1), 100-108.

[18]   Hochbaum S. D., Pathria A., 2015. Analysis of the greedy approach in problems of maximum *k*-coverage. Naval Research Logistics 45(6), 615-627.

[19] Konak A., Coit W. D., Smith E. A., 2006. Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering & System Safety 91(9), 992-1007.

[20] Llc, M. G., 2014. Providing emergency communication services using a vehicle telematics unit and a mobile device. USPTO Patent Application 20130130639.

[21] Lin Z., 2016. Discussion on Application of Long-Distance Directional Coverage Technology in Emergency Communication. Mobile Communications 40(14), 29-33.

[22] Ma C., Li Y., He R., Qi B., Diao A., 2012. Research on location problem of emergency service facilities based on genetic-simulated annealing algorithm. International Journal of Wireless & Mobile Computing 5(2), 206-211.

[23] Mcgregor A., Vu T. H., 2016. Better Streaming Algorithms for the Maximum Coverage Problem. Theory of Computing Systems, Special Issue on Database Theory, 1-25.

[24] Mohammadi M., khotbesara Z. D., Mirzazadeh A., 2014. MCLP and SQM Models for the emergency vehicle districting and location problem. Decision Science Letters 3(4), 479-490.

[25] Paul R. N., Lunday J. B., Nurre G. S., 2017. A Multiobjective, Maximal Conditional Covering Location Problem Applied to the Relocation of Hierarchical Emergency Response Facilities. Omega 66,147-158.

[26] Pirkul H., Schilling A. D., 1991. The maximal covering location problem with capacities on total workload. Management Science 37(2), 233-248.

[27] Rabieyan R., Esfandiari N., 2011. Optimizing unprecedented Maximal Covering Location Problem with simulated annealing Algorithm. IEEE International Conference on Business, Engineering and Industrial Applications, Kuala Lumpur, Malaysia. 5-7 June, 2011.

[28] Takaci A., Maric M., Drakulic D., 2012. The role of fuzzy sets in improving maximal covering location problem (MCLP). IEEE Jubilee International Symposium on Intelligent Systems and Informatics, Subotica, Serbia. 20-22 Sept. 2012, pp.103-106.

[29] Wang F., Xu Y., Li Y., 2006. A Review of the Discrete Facility Location Problem. International Journal of Plant Engineering & Management 11(1), 40-50.

[30] Wenchuan Earthquake, 2008. https://baike.baidu.com/item/ 5%C2%B712%E6%B1%B6%E5%B7%9D%E5%9C%B0%E9%9C%87/11042644?fr=aladdin, accessed in [4/10/2019]

[31] Xiao Y., Zuo X., Kaku I., Zhou S., Pan X., 2019a. Development of energy consumption optimization model for the electric vehicle routing problem with time windows. Journal of Cleaner Production 225, 647-663.

[32] Xiao Y., Huang C., Huang J., Kaku I., Xu Y., 2019b. Optimal mathematical programming and variable neighborhood search for k-modes categorical data clustering. Pattern Recognition 90, 183–195.

[33] Xiao Y., Konak A., 2017. A genetic algorithm with exact dynamic programming for the green vehicle routing & scheduling problem. Journal of Cleaner Production. 167, 1450-1463.

[34] XiaoY., Xie Y., Kulturel-Konak S., Konak A., 2017. A problem evolution algorithm with linear programming for the dynamic facility layout problem-A general layout formulation. Computers & Operations Research 88,187-207.

[35] Xiao Y., Konak A., 2015. A simulating annealing algorithm to solve the green vehicle routing & scheduling problem with hierarchical objectives and weighted tardiness. Applied Soft Computing 34, 372-388.

[36] Xiao Y., Zhang R., Zhao Q., Kaku I., Xu Y., 2014. A variable neighborhood search with an effective local search for uncapacitated multilevel lot-sizing problems. European Journal of Operational Research 235(1), 102-114.

[37] Xiao Y., Kaku I., Zhao Q., Zhang R., 2012. Neighborhood search techniques for solving uncapacitated multilevel lot-sizing problems. Computers & Operations Research. 39(3), 647-658.

[38] Xiao Y., Konak A., 2016. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. Transportation Research Part E Logistics and Transportation Review 88, 146-166.

[39] Xie Y., Zhou S., Xiao Y., Kulturel-Konak S., Konak A., 2018 A β-accurate linearization method of Euclidean distance for the facility layout problem with heterogeneous distance metrics. European Journal of Operational Research 265(1), 26-38.

[40] Yin P., Mu L., 2012. Modular capacitated maximal covering location problem for the optimal siting of emergency vehicles. Applied Geography

34, 247-254.

[41] You M., Xiao Y., Zhang S., Yang Y., Zhou S., 2019. Optimal mathematical programming for the warehouse location problem with Euclidean distance linearization. Computers & Industrial Engineering 136, 70–79.

[42] Zarandi F. H. M., Davari S., Sisakht H. A. S., 2011. The large scale maximal covering location problem. Scientia Iranica 18(6), 1564-1570.

[43] Zhang B., Peng J., Li S., 2017. Covering Location Problem of Emergency Service Facilities in an Uncertain Environment. Applied Mathematical Modelling 51, 429-447.

[44] Zitzler E., Deb K., Thiele L., 2000. Comparison of multiobjective evolutionary algorithms: empirical results. Evolutionary Computing 8(2), 173–95.

Appendix

Equation $z = \sqrt{x^2 + y^2}$ expresses a cone surface in a x-y-z coordinate system. Fig. A1 shows the intersecting circle by the plane $z = r$ and $z = \sqrt{x^2 + y^2}$, where r is positive number. Let A and B be two points on the circle that pose angles $k\theta$-θ and and $k\theta$ with the x-axis, where θ is a given angle and k is an integer. Then, the coordinates of points O, A, and B are expressed as $O(0, 0, 0)$, $A(r\cos(k\theta-\theta), r\sin(k\theta-\theta), r)$, $B(r\cos(k\theta), r\sin(k\theta), r)$, respectively. Let $z = ax + by$ represent the equation of the plane passing points O, A, and B. It can be deduced that

$$a = \frac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} \text{ and } b = \frac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)}.$$
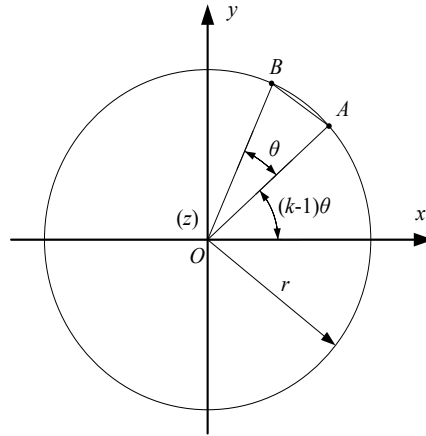

Fig. A1. A linear surrogate of the Euclidean distance

Thus, the Euclidean distance equation, $z = \sqrt{x^2 + y^2}$, can be bounded of its minimum value by a set of planes with different k by the constraints as follows.

$$z \geq \frac{\sin(k\theta) - \sin(k\theta - \theta)}{\sin(\theta)} x + \frac{\cos(k\theta - \theta) - \cos(k\theta)}{\sin(\theta)} y, \quad \forall k = 1, 2, ..., \eta, \tag{A1}$$

where θ is a given angle and η represents the number of planes calculated by $\eta = \left\lceil \dfrac{\pi}{2\theta} \right\rceil$.

Let's see the first plane, i.e., k=1, which has the equation $z = x + y\tan(\theta/2)$. The maximum approximate

error by this plane for replacing the equation $z = \sqrt{x^2 + y^2}$ with Eq. (A1) is at the middle of arc (A, B), noted by point C as shown in Fig. A2. The coordinate of point C is $(z\cos(\theta/2), z\sin(\theta/2), z)$. Suppose the correspondent point C' on the plane has a coordinate of $(z\cos(\theta/2), z\sin(\theta/2), z')$, then $z' = z\cos(\theta/2) + z\sin(\theta/2)\tan(\theta/2)$. Thus, the error rate ε can be calculated by

$$\varepsilon = \frac{z'-z}{z} \times 100\% = \cos(\theta/2) + \sin(\theta/2)\tan(\theta/2) - 1 \tag{A2}$$

Thus, the minimum angle for guaranteeing a given maximum error rate ε is $\theta = 2\arccos\dfrac{1}{(\varepsilon+1)}$.
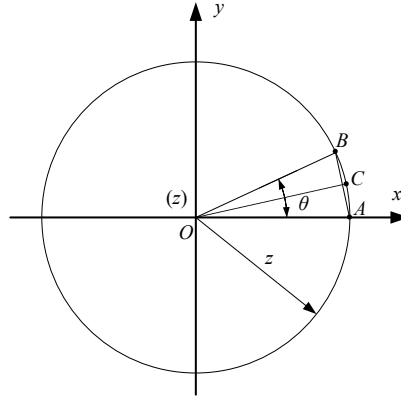

Fig. A2. The maximum approximate error estimation

```
#Parameters
set RT;                          #set of rescue teams, RT
set CV;                          #set of communication vehicles, CV
param RT_X{RT};                  #coordinate X of RTs
param RT_Y{RT};                  #coordinate Y of RTs
param delta{RT};                 #maximum deviations of the RTs from their current locations
param CV_X0{CV};                 #initial coordinate X of CVs
param CV_Y0{CV};                 #initial coordinate Y of CVs
param q:=18;                     #given parameter for Euclidean distance linearization
param cita:=0.0872;              #given parameter for Euclidean distance linearization
param Dmax{CV};                  #maximum coverage range of CVs
param C{CV};                     #capacity of CVs
param Lmax;
param M:=999;                    #a large number

#Decision variables
var CV_X{CV}>=0;                 #coordinate X of CV
var CV_Y{CV}>=0;                 #coordinate Y of CV
var dx{RT,CV} >= 0;
var dy{RT,CV} >= 0;
var d{RT,CV} >= 0;               #Euclidean distances between CVs and RTs
var E{RT,CV} binary;
var E1{RT} binary;
var DX{CV} >= 0;
var DY{CV} >= 0;
var D{CV} >= 0;                  #moved distances of CVs

maximize Signal_Coverage:    sum{i in RT} E1[i];

subject to Con1a{i in RT, j in CV}: dx[i,j] >= RT_X[i] - CV_X[j];
```

```
subject to Con1b{i in RT, j in CV}: dx[i,j] >= CV_X[j] - RT_X[i];
subject to Con2a{i in RT, j in CV}: dy[i,j] >= RT_Y[i] - CV_Y[j];
subject to Con2b{i in RT, j in CV}: dy[i,j] >= CV_Y[j] - RT_Y[i];
subject to Con3{i in RT,j in CV, p in 1..q}:
          d[i,j] >= dx[i,j]*(sin(p*cita)-sin(p*cita-cita))/sin(cita) + dy[i,j]*(cos(p*cita-cita)-cos(p*cita))/sin(cita);
subject to Con4{i in RT, j in CV}: M*(E[i,j]-1) <= Dmax[j]-d[i,j]-delta[i];
subject to Con5{i in RT}: E1[i] <= sum{j in CV}E[i,j];
subject to Con6{j in CV}: C[j] >= sum{i in RT}E[i,j];
subject to Con7a{j in CV}: DX[j] >= CV_X0[j] - CV_X[j];
subject to Con7b{j in CV}: DX[j] >= CV_X[j] - CV_X0[j];
subject to Con8a{j in CV}: DY[j] >= CV_Y0[j] - CV_Y[j];
subject to Con8b{j in CV}: DY[j] >= CV_Y[j] - CV_Y0[j];
subject to Con9{j in CV, p in 1..q}:
          D[j] >= DX[j]*(sin(p*cita)-sin(p*cita-cita))/sin(cita) + DY[j]*(cos(p*cita-cita)-cos(p*cita))/sin(cita);
subject to Con10{j in CV}: D[j] <= Lmax;
```

Fig. A3 The single-period C-MCLP model coded by AMPL

```
#parameters
set T;                          #set of periods
set RT{T};                      #set of rescue teams, RT
param RT_X{t in T, RT[t]};      #coordinate X of rescue teams in all periods
param RT_Y{t in T, RT[t]};      #coordinate Y of rescue teams in all periods
param delta{t in T, RT[t]};
set CV;                         #set of communication vehicles, CV
param CV_X0{CV};                #initial coordinate X of CVs
param CV_Y0{CV};                #initial coordinate Y of CVs
param q:=18;                    #given parameter for Euclidean distance linearization
param cita:=0.0872;             #given parameter for Euclidean distance linearization
param Dmax{CV};                 #maximum coverage range of CVs
param C{CV};                    #capacity of CVs
param Lmax;
param M:=999;                   #a large number representing the maximum range of the to-be rescue area

#decision variables:
var CV_X{T,CV}>=0;              #coordinate X of CVs
var CV_Y{T,CV}>=0;              #coordinate Y of CVs
var dx{t in T,RT[t],CV} >= 0;
var dy{t in T,RT[t],CV} >= 0;
var d{t in T,RT[t],CV} >= 0;    #distances between CVs and RTs in periods
var E{t in T,RT[t],CV} binary;
var E1{t in T,RT[t]} binary;
var DX{T,CV} >= 0;
var DY{T,CV} >= 0;
var D{T,CV} >= 0;               #moved distances of CVs in periods

maximize Total_Signal_Coverage: sum{t in T,i in RT[t]}E1[t,i];
minimize Total_Moved_Distance: sum{t in T,j in CV}D[t,j];

subject to Con1a{t in T,i in RT[t], j in CV}: dx[t,i,j] >= RT_X[t,i] - CV_X[t,j];
subject to Con1b{t in T,i in RT[t], j in CV}: dx[t,i,j] >= CV_X[t,j] - RT_X[t,i];
subject to Con2a{t in T,i in RT[t], j in CV}: dy[t,i,j] >= RT_Y[t,i] - CV_Y[t,j];
subject to Con2b{t in T,i in RT[t], j in CV}: dy[t,i,j] >= CV_Y[t,j] - RT_Y[t,i];
subject to Con3{t in T,i in RT[t],j in CV, p in 1..q}:
    d[t,i,j] >= dx[t,i,j]*(sin(p*cita)-sin(p*cita-cita))/sin(cita) + dy[t,i,j]*(cos(p*cita-cita)-cos(p*cita))/sin(cita);
subject to Con4{t in T, i in RT[t], j in CV}: M*(E[t,i,j]-1) <= Dmax[j]-d[t,i,j];
subject to Con5{t in T, i in RT[t]}: E1[t,i] <= sum{j in CV}E[t,i,j];
subject to Con6{t in T,j in CV}: C[j] >= sum{i in RT[t]}E[t,i,j];
subject to Con7a{j in CV}: DX[1,j] >= CV_X0[j] - CV_X[1,j];
subject to Con7b{j in CV}: DX[1,j] >= CV_X[1,j] - CV_X0[j];
subject to Con7c{t in T, j in CV:t>1}: DX[t,j] >= CV_X[t-1,j] - CV_X[t,j];
subject to Con7d{t in T, j in CV:t>1}: DX[t,j] >= CV_X[t,j] - CV_X[t-1,j];
subject to Con8a{j in CV}: DY[1,j] >= CV_Y0[j] - CV_Y[1,j];
subject to Con8b{j in CV}: DY[1,j] >= CV_Y[1,j] - CV_Y0[j];
subject to Con8c{t in T, j in CV:t>1}: DY[t,j] >= CV_Y[t-1,j] - CV_Y[t,j];
```

subject to Con8d{t in T, j in CV:t>1}: DY[t,j] >= CV_Y[t,j] - CV_Y[t-1,j];
subject to Con9{t in T, j in CV, p in 1..q}:
      D[t,j] >= DX[t,j]*(sin(p*cita)-sin(p*cita-cita))/sin(cita) + DY[t,j]*(cos(p*cita-cita)-cos(p*cita))/sin(cita);
subject to Con10{t in T, j in CV}: D[t,j] <= Lmax;

Fig. A4 The multiple-period C-MCLP model coded by AMPL