RAIRO-Oper. Res. 52 (2018) 1245–1260
https://doi.org/10.1051/ro/2018026

RAIRO Operations Research
www.rairo-ro.org

MODELING AND SOLUTION OF MAXIMAL COVERING PROBLEM CONSIDERING GRADUAL COVERAGE WITH VARIABLE RADIUS OVER MULTI-PERIODS

Alireza Eydi[1,*] and Javad Mohebi[2]

Abstract. Facility location is a critical component of strategic planning for public and private firms. Due to high cost of facility location, making decisions for such a problem has become an important issue which have gained a large deal of attention from researchers. This study examined the gradual maximal covering location problem with variable radius over multiple time periods. In gradual covering location problem, it is assumed that full coverage is replaced by a coverage function, so that increasing the distance from the facility decreases the amount of demand coverage. In variable radius covering problems, however, each facility is considered to have a fixed cost along with a variable cost which has a direct impact on the coverage radius. In real-world problems, since demand may change over time, necessitating relocation of the facilities, the problem can be formulated over multiple time periods. In this study, a mixed integer programming model was presented in which not only facility capacity was considered, but also two objectives were followed: coverage maximization and relocation cost minimization. A metaheuristic algorithm was presented to solve the maximal covering location problem. A simulated annealing algorithm was proposed, with its results presented. Computational results and comparisons demonstrated good performance of the simulated annealing algorithm.

Mathematics Subject Classification. 90B80

Received November 25, 2016. Accepted April 27, 2018.

1. Introduction

Covering problem is one of the most popular facility location models. While covering models is not any new, they have always been very attractive for researchers. This is due to their real-world applications such as determining the number and locations of public schools, police stations, libraries, hospitals, public buildings, post offices, parks, military bases, radar installations, branch banks, shopping centers and waste-disposal facilities. Covering models deal with covering demands. In most covering models, demand is said to be covered once it reaches and maintained within a predefined standard distance from at least one facility. According to the literature, facility location covering problem was first proposed by Toregas *et al.* as the location set covering problem (LSCP) [13]. The LSCP is a mandatory covering model with its objective being to find the minimum number of facilities to cover all demand points. However, full coverage is hard to achieve in reality due to limited

Keywords and phrases: Gradual maximal covering location problem, variable radius coverage, capacitated facilities, multi-objective coverage, facility relocation, simulated annealing algorithm.

[1] Faculty of Engineering, University of Kurdistan, Pasdaran Blvd., Post Box No. 416, Sanandaj, Iran.
[2] MSC of Industrial Engineering, University of Kurdistan, Sanandaj, Iran.

[*] Corresponding author: Alireza.Eydi@uok.ac.ir

© EDP Sciences, ROADEF, SMAI 2018

resources. For a demand point far away from other points, it is probably not possible to be covered within the predefined distance standard. A few years later, the first maximum deterministic covering problem was proposed by Church and ReVelle who termed it the maximal covering location problem (MCLP) given a limited number of facilities [12]. This model aims to maximize the demand coverage. The coverage objective makes three main assumptions implicitly, as follows:

A1: Either complete coverage or nothing covered. Each and any demand point within the coverage radius of a facility is completely covered (*i.e.*, the full demand existing at this point is satisfied as per the objective), while the demand points outside the coverage radius are not covered at all.
A2: Individual coverage. Whether a certain demand point is covered or not is determined by its proximity to a single (individual) facility; namely the closest one. The next closest facility has no bearing on the coverage.
A3: Fixed coverage radius. The coverage radius R (*i.e.*, the maximum travel distance which determines whether a customer is covered or not) is specified exogenously and is not a decisive variable.

The gradual cover models seek to relax the "all or nothing" assumption A1 by replacing it with a general coverage function which represents the proportion of covered demand at a certain distance from the facility.

Variable radius problem is primarily designed to relax the "fixed coverage radius" assumption A3, making the coverage radius an endogenously determined function of the facility cost. Thus, instead of locating a certain pre-determined number of facilities, the decision-maker has a certain budget to be used to construct facilities of different types, with the more expensive facilities having larger coverage radii. Therefore, the model adds a design aspect (what type(s) of facility to build?) to the typical question of location asking where the facilities should be sited.

The set covering location problem was first introduced by Hakimi. The model aimed to determine the minimum number of police officers needed to cover strategic nodes on a network of highways [17]. Megiddo *et al.* proved that the maximal covering location problem is NP-Hard [21]. The original paper on gradual coverage problems seems to be presented by Church and Roberts [8] who described a discrete model with a step-coverage function. Berman and Krass [5] discussed network version again with the step-coverage function and provided efficient formulations and heuristic approaches. Berman *et al.* [6] offered a more general form of the gradual coverage function on a network. Their model combined the p-median problem and covering problem. Karasakal and Karasakal [19] presented a model with partial coverage. They presented a Lagrangian-based solution approach. Drezner *et al.* [10] considered gradual coverage problem with coverage functions on a plane. In the plane model, a facility could be located anywhere across a given plane. Eiselt and Marianov [11] considered gradual coverage in the form of a set covering problem. In their model, service quality was considered as a measure which was divided into several classes.

Rahim [23] presented a variable neighborhood search approach for the combination of gradual covering problem with traveling salesman problem. Orbay [22] suggested an evolutionary algorithm for solving the bi-objective relocation problem, including the concept of partial coverage. The first objective maximized the covered demand while the second one worked to minimize the number of facilities. Toreyen [26] presented the hierarchical maximal covering location problem considering the partial coverage. He also presented an integer formulation for this problem, and designed a genetic algorithm to achieve a near optimal solution.

Berman *et al.* [4] studied a variable radius covering problem. They sought finding optimal radius along with optimum number and location of facilities. The objective function introduced in the model by these authors minimized the cost of facility location while assuring the coverage of all demand points. They presented discrete and continuous problems and approaches for solving the discrete case, whereas heuristic approaches were presented to solve large-scale problems on the plane.

Ballou was the first to publish a paper on limited application of static and deterministic location problems [3]. Attempting to locate a single warehouse in such a way to maximize profit over a finite planning horizon, Ballou used a series of static deterministic optimal solutions to solve the dynamic problem. For each period in the specified horizon, he solved for the optimal warehouse location, establishing a set of potential "good" location sites. Dynamic programming was then used to determine the best schedule for selecting a subset of these sites as an "optimal" location and relocation strategy for the planning period. Albareda-Sambola *et al.* [2] introduced the multi-period incremental service facility location problem whose goal was to set a number of new facilities over a finite time horizon in such a way to dynamically cover a given set of customers' demand. They proposed a solving approach that would provide both lower and upper bounds by combining sub-gradient optimization to solve a Lagrangian dual with an *ad hoc* heuristic wherein the information obtained from the Lagrangian sub-problem was used to generate corresponding feasible solutions. Canel *et al.* [7] developed an algorithm to solve capacitated, multi-commodity, multi-period, multi-stage facility location problem. The proposed algorithm had two parts. In the first part, branch and bound method was used to generate a list of candidate solutions for each period and then dynamic programming was used to find optimal sequence of configurations over the multi-period planning horizon. The proposed algorithm was particularly effective when the facility reopening and closing costs were relatively significant in a multi-period problem. Wesolowsky and Trucott [27] extended the analysis of multi-period node location-allocation problems, allowing facilities to be relocated in response to predicted changes in demand. An integer programming model was presented, with a constraint restricting the number of location changes in each period. A dynamic programming formulation was also presented. Gendreau *et al.* proposed a model they termed the maximal expected coverage relocation problem (MECRP), and provided a dynamic relocation strategy for idle emergency medical service (EMS) facilities siting in low demand areas [16]. The objective was to maximize the expected demand coverage with the number of relocated facilities not exceeding a predefined value. Zarandi *et al.* [28] considered MCLP over several time periods and used a simulated annealing algorithm which was capable of solving problems with up to 2500 nodes and 200 facilities.

Rashidi and Jafari [24] offered a nonlinear model for solving the MCLP considering partial coverage constraint and facility capacity. Jabal Ameli and Bankian Tabrizi [18] further considered the MCLP with gradual coverage and variable radius coverage.

As was defined above, facility location is among the strategic issues for governmental agencies as well as private sectors. Covering problems are among the most common and useful discussions on the facility location. Although many researches have worked on covering problems, still many subjects have remained to be investigated to expand the knowledge of coverage problems. Hence, the present paper tries to formulate real-world situations in the form of a model by combining some of the covering models, among which the followings are notable:

Sometimes, service facilities have a limited capacity for service delivery. Coverage radii are different depending on the size and type of the facilities. Since the demand may change over time, it is necessary to formulate the problem within a multi time period framework. This change in demand may occur as people migrate from one point to another. Also in the case of emergency services such as ambulance services, in warm seasons when people are used to travel to places with more moderate climates, there will be a rise in the rate of accidents, resulting in an increased need for emergency facilities. In other words, the number of demand points increases. This is also the case for police officers allocation in such an area. In this special case, some clear and predictable changes in demand are assumed. However, a covering problem is NP-Hard, so that one may try to solve the model by a simulated annealing algorithm which is to be applied within a reasonable time.

The present paper deals with an important logistic problem that combines facility location management problems. A large number of variations in maximal covering problems were previously considered in different works. The present work, however, aims to present a generalization of the previous models considering a vast number of real-world cases. The paper addresses a new version of the maximal covering problem considering gradual coverage, variable radius, multi-periods and capacitated facilities.

Research findings can be outlined as follows:

- Providing a formulation for research questions with the following features: facility relocation within different periods is considered in the gradual coverage model with variable radius.
- Due to the computational complexity of the maximal covering location problems as well as their derivative problems, a simulated annealing algorithm for solving the problem will be studied.

The rest of the paper is organized as follows. A mathematical model is defined in Section 2. The simulated annealing algorithm is presented in Section 3 while the computational results are brought in Section 4. Some recommendations for future research are suggested in Section 5.

2. Problem formulation

The following assumptions are made in this study:

- The problem is a maximal covering location problem.
- There are a number of demand points.
- The demand is specified at each point but may vary over time.
- There are a number of potential locations where facility can be constructed.
- There are a number of facilities to be deployed in potential locations.
- With increasing the distance between facility and demand point, the covered demand decreases.
- The facilitator deals with a fixed cost as well as a variable cost for constructing the facility, with the variable cost having direct influence on the coverage radius.
- There is a limited budget available for facility location in the first period.
- Services provided by each facility cannot exceed the facility capacity.
- Facility relocation occurs at discrete points of time.
- The problem is formulated over multiple time periods and number of facility relocations in each period is limited.

In this section, the proposed model is formulated. The main purpose of this model is to consider real-world situations. The notations (indexes, parameters and variables) used to express the mathematical form of the problem are introduced below.

(a) Indexes:
   i: Set of demand points, $i = 1, 2, \ldots, I$.
   j: Set of potential sites for facility construction, $j = 1, 2, \ldots, J$.
   t: Set of time periods, $t = 1, 2, \ldots, T$.

(b) Data of the problem:
   w_{it}: Weight of demand point i over time period t.
   B: The maximum budget available for facility construction over the first time period.
   P: The maximum number of facilities that can be built over any time period.
   d_{ij}: The minimum distance between the demand point i and the facility j. This distance is calculated as an Euclidian distance.
   F_{jt}: Net present value (NPV) of the fixed cost of facility construction for facility j over time period t.
   K_{jt}: Coverage capacity of the facility j over time period t.
   m_t: Maximum number of facility location changes over time period t.
   c_{jt}: NPV of the destruction cost of the facility j over time period t.

(c) Decision variables:

$$x_{jt}\colon \begin{cases} 1 & \text{If facility } j \text{ is to be existed in time period } t \\ 0 & \text{Otherwise} \end{cases}.$$

$$z_{jt}\colon \begin{cases} 1 & \text{If facility } j \text{ is to be removed in time period } t \\ 0 & \text{Otherwise} \end{cases}.$$

$$z'_{jt}\colon \begin{cases} 1 & \text{If facility } j \text{ is to be constructed in time period } t \\ 0 & \text{Otherwise} \end{cases}.$$

$$y_{ijt}\colon \begin{cases} 1 & \text{If demand point } i \text{ is to be covered by facility } j \text{ in time period } t \\ 0 & \text{Otherwise} \end{cases}.$$

   r_{jt}: The coverage radius of facility j over time period t, $r_{jt} = \max_{i=1,2,\ldots,I}(d_{ij}y_{ijt})$.
   l_{ijt}: Fraction of demand point i to be covered by facility j over time period t.
   $q_{jt}(r_{jt})$: NPV of the variable cost of facility construction for facility j in time period t.
   c'_{jt}: NPV of the facility construction cost for facility j in time period t.

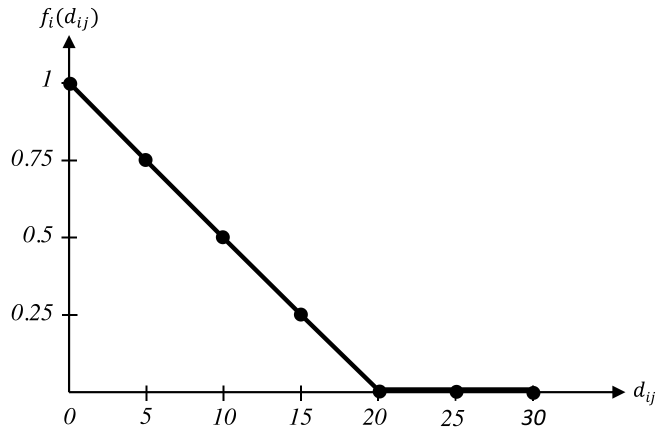

FIGURE 1. An example of the coverage function.

c_{ijt}: Coverage function $\left\{ \begin{array}{ll} w_{it} f_i\left(d_{ij}\right) & \text{If } 0 \leq d_{ij} \leq r_{jt} \\ 0 & \text{Otherwise} \end{array} \right.$, where $f_i\left(d_{ij}\right)$ is a function of d_{ij} with its value falling within $[0,\ 1]$.

In this problem, the coverage, as a quantity, is defined as a function of the distance between demand point i and facility j, as follows [6]:

$$f_i\left(d_{ij}\right) = \left\{ \begin{array}{ll} 1 - d_{ij}/r_{jt} & d_{ij} \leq r_{jt} \\ 0 & \text{Otherwise} \end{array} \right. .$$

The value of r_{jt} is determined during the optimization process. The coverage function is shown in Figure 1 where $r_{jt} = 20$.

The problem formulation can be presented as follows:

$$\text{Max}\, z_1 = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{t=1}^{T} (w_{it} l_{ijt}), \tag{2.1}$$

$$\text{Min}\, z_2 = \sum_{j=1}^{J} \sum_{t=2}^{T} (c'_{jt} z'_{jt} + c_{jt} z_{jt}), \tag{2.2}$$

S.T.

$$\sum_{j=1}^{J} x_{jt} \leq P, \qquad t = 1, \ldots, T, \tag{2.3}$$

$$\sum_{j=1}^{J} y_{ijt} = 1, \qquad i = 1, \ldots, I;\ t = 1, \ldots, T, \tag{2.4}$$

$$y_{ijt} = x_{jt}, \qquad i = 1, \ldots, I;\ j = 1, \ldots, J;\ t = 1, \ldots, T, \tag{2.5}$$

$$l_{ijt} = y_{ijt}, \qquad i = 1, \ldots, I;\ j = 1, \ldots, J;\ t = 1, \ldots, T, \tag{2.6}$$

$$l_{ijt} w_{it} \leq c_{ijt}, \qquad i = 1, \ldots, I;\ j = 1, \ldots, J;\ t = 1, \ldots, T, \tag{2.7}$$

$$\sum_{i=1}^{I} l_{ijt} w_{it} \leq K_{jt} x_{jt}, \qquad j = 1, \ldots, J;\ t = 1, \ldots, T, \tag{2.8}$$

$$x_{jt} - x_{j,t-1} = z'_{jt} - z_{jt}, \qquad j = 1, \ldots, J;\ t = 2, \ldots, T, \tag{2.9}$$

$$\sum_{j=1}^{J} \left(F_{jt}x_{jt} + q_{jt}(\max_{i=1,2,\ldots,n}(y_{ijt}d_{ij}))\right) \leq B, \qquad t = 1, \tag{2.10}$$

$$F_{jt}x_{jt} + q_{jt}\left(\max_{i=1,2,\ldots,n}(y_{ijt}d_{ij})\right) = c'_{jt}, \qquad j = 1, \ldots J;\ t = 2, \ldots, T, \tag{2.11}$$

$$\sum_{j=1}^{J} z_{jt} \leq m_t, \qquad t = 2, \ldots, T, \tag{2.12}$$

$$\begin{aligned} &x_{jt}, z_{jt}, z'_{jt}, y_{ijt} \in \{0, 1\};\ l_{ijt} \geq 0; \\ &r_{jt}, c_{ijt}, c'_{jt}, q_{jt} \geq 0 \qquad\qquad i = 1, \ldots, I;\ j = 1, \ldots, J;\ t = 1, \ldots, T. \end{aligned} \tag{2.13}$$

Objective function (2.1) maximizes the covered demand while objective function (2.2) minimizes the facility relocation cost over different time periods. Constraint (2.3) determines the maximum number of facilities that can be located in the time period t. Constraint (2.4) states that any node must be covered by exactly one facility over each period of time. Constraint (2.5) guarantees that demand point i is covered only if one facility is placed within site j in the time period t. Constraint (2.6) shows the fraction of demand i that is covered by facility j in the time period t. Constraint (2.7) states that, in time period t, facility j covers the demand point i by as much as c_{ijt}. Constraint (2.8) limits the capacity of facility j in the time period t. Constraint (2.9) ensures that the relocation costs are considered. Constraint (2.10) shows the budget constraint in the first time period (part of the objective function seeks minimizing the cost of facility relocation. Since the relocation of facilities is undertaken after the first period, the primary location of facilities has to be budget-constrained). Constraint (2.11) indicates the location cost of facility j in the time period t. Constraint (2.12) limits the number of relocation changes in each time period from period 2 to T (in a dynamic problem, the number of facility location changes in each period may be limited to reflect tolerable levels of organizational disruption. This dynamic configuration constraint is a natural counterpart of a static restriction on the number of facilities which is motivated by organizational policy). Constraint (2.13) specifies the sign of variables.

2.1. Total weighted method

In the context of economics, utility emphasizes a decision-maker's satisfaction. In terms of multi-objective optimization, a utility function is defined for each objective and represents the relative importance of the objective. A special form of the utility model (often used in multi-objective problems) involves the use of weights w_j (objective coefficients). In this method, which is known as total weighted method, it is assumed that the weights scale and convert the goals to utilities. It means that a multi-objective problem would be converted into a single-objective one.

In such a way, the user can convert a multi-objective problem to a single-objective one by multiplying each objective by its proposed weight. This method is the easiest and possibly the most common classic approach to a multi-objective problem and represents the most convenient way to deal with such a problem [9].

Suppose we solve the problem (Eqs. (2.1)–(2.13)) two different times by considering only one of the objective functions z_1 and z_2, each time. In order to build a utility function we use two weights with $w_1 + w_2 = 1$. Therefore, the utility function is as follows:

$$\begin{aligned} \text{Max utility function} = w_1 &\left[\frac{\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{t=1}^{T} w_{it}l_{ijt} - f^1_{\min}}{f^1_{\max} - f^1_{\min}}\right] \\ &+ w_2 \left[\frac{f^2_{\max} - \sum_{j=1}^{J}\sum_{t=2}^{T}\left(c'_{jt}z'_{jt} + c_{jt}z_{jt}\right)}{f^2_{\max} - f^2_{\min}}\right], \end{aligned} \tag{2.14}$$

```
Input: Cooling schedule.
s = s_0; /* Generation of the initial solution */
T = T_max; /* Starting temperature */
Repeat
  Repeat /* At a fixed temperature */
    Generate a neighbor s';
    ΔE = f(s') − f(s);
    If ΔE ≤ 0 Then s = s' /* Accept the neighbor solution */
    Else Accept s' with a probability of e^(−ΔE/T);
  Until Equilibrium condition
  /* e.g. a given number of iterations executed at each temperature T */
  T = g(T); /* Temperature update */
Until Stopping criterion satisfied /* e.g. T < T_min */
Output: Best solution found.
```

Figure 2. Pseudo code of the SA algorithm.

where $f^1_{\min}$ is minimum amount of coverage function, $f^1_{\max}$ is maximum amount of coverage function, $f^2_{\min}$ is minimum amount of relocation cost function, and $f^2_{\max}$ is maximum amount of relocation cost function. Note that the normalization approach is used to build the objective function (Eq. (2.14)).

To solve the model described by equations (2.3)–(2.14), GAMS software version 23.3 and BARON solver (because of its capability for solving nonlinear mixed integer models) were used [14, 15]. Due to the nonlinear term $r_{jt} = \max_{i=1,2,\ldots,I}(d_{ij}y_{ijt})$, GAMS solvers fail to find a suitable solution for this problem. In other words, after running this solver, all integer variables are found to have continuous values, *i.e.*, nonlinear programming (NLP) solution is printed as output solution. Hence, linear form of the above constraint, namely $r_{jt} \geq d_{ij}y_{ijt}$, was used. In this case, the coverage radius was the same for all facilities, which was in contradiction with the definition of a variable coverage radius problem. Despite having the same coverage radius, GAMS solution can still be considered as an upper or lower bound. With such an interpretation, the use of a heuristic or meta-heuristic algorithm is important when achieving a solution wherein variable coverage radius concept is taken into account matters. This algorithm is described in the next section.

3. Simulated annealing algorithm

Simulated annealing (SA) metaheuristic is one of the well-known algorithms inspired from physical annealing of solids. It was first introduced by Kirkpatrick *et al.* [20] to solve large combinatorial optimization problems. SA algorithm simulates energy changes in a system, so that the cooling process continues until a steady state is reached. The algorithm convergence is guaranteed only when the cooling process is done gradually, but experiments have shown that even when the program comes with relatively rapid cooling phase, this algorithm remains to be an effective optimization technique. SA attempts to escape from local optima by probabilistically choosing nonimproving solutions.

Simplicity of implementation, good adaptability of the algorithm with our model, and the way to represent solution structure were among the most important reasons why we chose SA algorithm among other available meta-heuristics in the literature. Furthermore, SA is a local search algorithm and can work with models with many constraints, such as that of ours. Simulated annealing is a robust general technique which has been widely and successfully used for solving NP-Hard problems [1].

Figure 2 shows the pseudo code of the SA algorithm followed by a summary of the characteristics of SA algorithm.

Demands point					Facilities location	
1	4	2	3	5	3	5
2	5	4	1	3	3	5
2	5	1	4	3	3	5

FIGURE 3. Solution representation.

3.1. Solution representation

The first step to implement a meta-heuristic algorithm is to encode solutions of the problem on which the operators of the algorithm can be performed. A matrix of dimension $t * (i + P)$ is used for this purpose. In this matrix, t represents the number of time periods, and i is the number of demand points. Assuming 5 demand points, 3 time periods and $P = 2$, we have the solution represented in Figure 3.

In Figure 3, highlighted cells in each period indicate the location of facilities to which demand points are assigned. For example, demand points 1, 2 and 4 are covered by facility 3 while demand points 3 and 5 are covered by the facility 5 during the first time period.

3.2. Initial and neighborhood solution

In this algorithm, the initial solution is generated by taking several steps are as follows:

Step 1. Generate P random numbers between successive demand points. If generated numbers are equal to each other, this means that the number of facilities is less than P in the corresponding time period. These numbers determine facility locations in the first time period.
Step 2. Determine facilities locations in each time period according to the number of location changes (m_t) as well as P. For example, if the location of a facility is supposed to be changed in the next time period, randomly select one of the facilities that is to be removed in the next time period (facility 4). Then, randomly select a facility (other than facility 4) between the corresponding demand points (facility 5) and replace it with facility 4. In this way, facilities 4 and 5 are determined as generated and removed facilities in the second time period, respectively. Following the same procedure, specify the location of facilities in each time period (x_{jt}). Once finished with this step, constraints (2.3), (2.9) and (2.13) are completely satisfied, while constraint (2.12) is satisfied in most cases.
Step 3. Allocate the demand points to the facilities (y_{ijt}) randomly; therefore, select an arbitrary facility and allocate it to a demand point. Here an internal replication procedure to determine the best assignment is embedded which can be performed several times to achieve the best allocation performance. The solution is generated in this step, where constraints (2.4) and (2.5) are satisfied while the above-mentioned constraints are also still satisfied.
Step 4. Specify the values of facility relocation variables (z_{jt}, z'_{jt}) according to the values of x_{jt} and constraint (2.9).
Step 5. Calculate facility coverage radius and cost of facility construction as $r_{jt} = \mathrm{Max}_{i=1,2,\ldots,n}(d_{ij}y_{ijt})$ and $c'_{jt} = F_{jt}x_{jt} + q_{jt}(r_{jt})$, respectively (observe constraint (2.11)).
Step 6. To determine l_{ijt} values corresponding to the fraction of the demands covered by parts of the facility, we should first set to a small value the l_{ijt} values for which corresponding y_{ijt} values are equal to 1. Then, the l_{ijt} values will be increased in such a way that the constraints (2.6)–(2.8) are not breached.

Once the above steps are completed, the values of all variables are specified.

Neighborhood solution generation procedure is as follows:

By changing the location of demand in the first period, neighborhood solution is produced. In this case, two points can be chosen at random before moving their places. This process can be seen in Figure 4.

Initial solution

1	4	2	3	5	3	5

Neighborhood solution

1	5	2	3	4	3	5

FIGURE 4. Generated solution neighborhood procedure.

Once the neighborhood solution is generated in the first period, the process continues until all of the decision variables are generated during various periods, as it is mentioned above [20].

3.3. Other features of the SA algorithm

Fitness function: In this algorithm, coverage and relocation cost objective function values are used as fitness functions. Step 1: Calculate the value of coverage using the formula (2.1). Step 2: Calculate the value of relocation cost using the formula (2.2). Step 3: Calculate the value of fitness function using the formula (2.14).

Feasibility of solution: To make sure that the solution is feasible, one must check the constraints (2.10) and (2.12). A scalar penalty is considered for violation of these constraints. A penalty value of zero indicates feasibility of the generated solution.

Cooling schedule: There exist several types of cooling schedule in the literature. In this paper, the geometric cooling schedule is used, which can be described by the temperature-update formula $T_{k+1} = \alpha T_k$.

Equilibrium condition: In this algorithm, equilibrium condition is to reach a particular number of iterations.

Stopping criterion: Reaching near-zero temperatures stops the algorithm.

4. COMPUTATIONAL RESULTS

4.1. Dataset generation

To the best of our knowledge, the hybrid approach proposed in this article has not been presented in available literature. Most of the covering problems presented in literature have been generated randomly. Also in this research similar to the paper of ReVelle *et al.* [25], the problem data have been produced using uniform random distributions in a specific information domain. Coordinates of the demand points, demand (or population) of the points and the facility capacities are generated from [0, 30], [100, 200], and [300, 400] intervals, respectively. The distances between the demand points are calculated as Euclidean distances. In order to generate present cost of removing facilities, constant cost of constructing facilities, one may begin with calculating the values corresponding to the first period using a uniform distribution. Then, considering rate of return, the corresponding values to the next periods are computed. For this purpose, values of variables (present cost of facility destruction, fixed cost of facility construction) in the first time period were produced from [500, 900], [500, 1000] intervals, respectively, utilizing MS Excel software. Then, corresponding values to the next period were generated using the following relationship:

$$F = P(F/P, i\%, n),$$

where F is the data value in the period n (future value); P is the data value in the first period (present value); i is the desired rate of return and n is the year in which the value of F is being calculated.

4.2. Parameter setting

The performance of SA algorithm depends highly on the values of parameters of the algorithm. Considering one parameter at-a-time, a good combination of SA parameters was found. Table 1 summarizes the initial values for temperature (T_0), cooling rate (α), number of iterations at each temperature (k) and final temperature ($T_{\min}$). In order to investigate the best value for each parameter, five problems were tested against various values, with

Table 1. Tested levels of SA parameters.

Parameters	Levels		
T_0	100	1000	10 000
α	0.90	0.95	0.99
K	3	6	10
$T_{\min}$	0.01	0.005	0.001

Table 2. Parameter setting results for initial temperature.

Problem No.	(I, J, T)	P	100		1000		10 000	
			Objective function	Time	Objective function	Time	Objective function	Time
1	$(5, 5, 3)$	2	0.9583	9	0.9484	11	0.9484	12
2	$(7, 7, 3)$	4	0.5392	13	0.6531	16	0.4831	18
3	$(10, 10, 3)$	5	0.7436	26	0.7707	31	0.7777	36
4	$(12, 12, 3)$	5	0.7992	20	0.8108	24	0.781	28
5	$(14, 14, 3)$	6	0.7342	23	0.8005	28	0.7494	33
Average	–	–	0.7549	18.2	0.7967	22	0.7479	25.4

Table 3. Parameter setting results for cooling rate.

Problem No.	(I, J, T)	P	0.9		0.95		0.99	
			Objective function	Time	Objective function	Time	Objective function	Time
1	$(5, 5, 3)$	2	0.9185	1	0.9286	3	0.9575	12
2	$(7, 7, 3)$	4	0.4217	2	0.5353	4	0.4743	18
3	$(10, 10, 3)$	5	0.6825	4	0.7596	8	0.7386	37
4	$(12, 12, 3)$	5	0.7527	3	0.7554	6	0.7856	28
5	$(14, 14, 3)$	6	0.7468	4	0.7141	7	0.7357	33
Average	–	–	0.7044	2.8	0.7386	5.6	0.7383	25.6

the results given in Tables 2–5. All running times are in seconds, and $(w_1, w_2) = (0.5, 0.5)$. The best values were found to be $T_0 = 1000$, $\alpha = 0.95$, $k = 6$, and $T_{\min} = 0.005$.

4.3. Numerical examples solved by SA algorithm

In Tables 6–9, several examples solved by SA algorithm are presented together with their results. It should be noted that the following conclusions are obtained by taking $(w_1, w_2) = (0.5,\ 0.5)$. For sensitivity analysis, one can test different values of w_1 and w_2. These example cases were run on a computer equipped with a Core i5, M 460, 2.53 GHz processor and 4 GB of RAM.

In Table 6, facilities 2 and 4 are constructed within the first time period. The demand points 1, 3, 4 and 5 are covered by facility 4 while the demand point 2 is covered by facility 2. Since no relocation is performed in the second time period, facilities 2 and 4 will be constant. In this time period, the demand points 1, 2, 3, 4 and 5 are covered by facility 2. In the third time period, due to one location change, facility 2 is removed and no new facility is located, with the demand points 1, 2, 3, 4 and 5 covered by facility 4. In Table 6, it is considered

TABLE 4. Parameter setting results for final temperature.

Problem No.	(I, J, T)	P	0.01		0.005		0.001	
			Objective function	Time	Objective function	Time	Objective function	Time
1	$(5, 5, 3)$	2	0.9376	10	0.9391	11	0.9393	12
2	$(7, 7, 3)$	4	0.5501	15	0.6119	16	0.514	18
3	$(10, 10, 3)$	5	0.803	31	0.7748	32	0.7773	36
4	$(12, 12, 3)$	5	0.7399	24	0.7908	25	0.7701	28
5	$(14, 14, 3)$	6	0.7501	28	0.7677	29	0.8543	33
Average	–	–	0.7561	21.6	0.7768	22.6	0.771	25.4

TABLE 5. Parameter setting results for number of iterations at each temperature (thermal equilibrium).

Problem No.	(I, J, T)	P	3		6		10	
			Objective function	Time	Objective function	Time	Objective function	Time
1	$(5, 5, 3)$	2	0.926	4	0.94	7	0.9298	12
2	$(7, 7, 3)$	4	0.5477	6	0.4877	11	0.4123	18
3	$(10, 10, 3)$	5	0.7454	12	0.7874	22	0.7741	36
4	$(12, 12, 3)$	5	0.7604	9	0.7594	17	0.7974	27
5	$(14, 14, 3)$	6	0.72	10	0.7708	20	0.7673	32
Average	–	–	0.7399	8.2	0.749	15.4	0.7362	25

TABLE 6. Results for five demand points over three time periods.

Number	Variable	Variables equal to 1	P	Objective function
1	x_{jt}	x_{21}, x_{31}, x_{22}, x_{32}, x_{33}	2	0.9497
2	y_{ijt}	y_{121}, y_{122}, y_{133}, y_{221}, y_{222}, y_{233}, y_{331}, y_{332}, y_{333}, y_{421}, y_{422}, y_{433}, y_{521}, y_{522}, y_{533}		
3	z_{jt}	z_{23}		
4	z'_{jt}	–		

TABLE 7. Results for seven demand points over three time periods.

Number	Variable	Variables equal to 1	P	Objective function
1	x_{jt}	x_{21}, x_{31}, x_{41}, x_{51}, x_{12}, x_{32}, x_{42}, x_{72}, x_{33}, x_{53}, x_{73}	4	0.7048
2	y_{ijt}	y_{141}, y_{172}, y_{173}, y_{231}, y_{242}, y_{273}, y_{331}, y_{332}, y_{373}, y_{441}, y_{442}, y_{473}, y_{521}, y_{572}, y_{533}, y_{641}, y_{632}, y_{673}, y_{721}, y_{712}, y_{753}		
3	z_{jt}	z_{22}, z_{52}, z_{13}, z_{43}		
4	z'_{jt}	z'_{12}, z'_{72}, z'_{53}		

Table 8. Results for five demand points over five time periods.

Number	Variable	Variables equal to 1	P	Objective function
1	x_{jt}	x_{11}, x_{31}, x_{41}, x_{12}, x_{32}, x_{52}, x_{33}, x_{43}, x_{34}, x_{44}, x_{45}, x_{55}	3	0.9357
2	y_{ijt}	y_{111}, y_{112}, y_{143}, y_{134}, y_{145}, y_{231}, y_{252}, y_{243}, y_{244}, y_{245}, y_{331}, y_{332}, y_{343}, y_{334}, y_{355}, y_{411}, y_{432}, y_{443}, y_{444}, y_{445}, y_{541}, y_{552}, y_{533}, y_{544}, y_{555}		
3	z_{jt}	z_{42}, z_{13}, z_{52}, z_{35}		
4	z'_{jt}	z'_{52}, z'_{43}, z'_{55}		

Table 9. Results for seven demand points over five time periods.

Number	Variable	Variables equal to 1	P	Objective function
1	x_{jt}	x_{21}, x_{31}, x_{41}, x_{51}, x_{32}, x_{42}, x_{52}, x_{62}, x_{13}, x_{33}, x_{43}, x_{14}, x_{34}, x_{44}, x_{15}, x_{65}, x_{75}	4	0.9053
2	y_{ijt}	y_{141}, y_{132}, y_{143}, y_{144}, y_{115}, y_{241}, y_{252}, y_{243}, y_{244}, y_{215}, y_{331}, y_{362}, y_{343}, y_{334}, y_{375}, y_{441}, y_{442}, y_{413}, y_{444}, y_{415}, y_{521}, y_{552}, y_{513}, y_{534}, y_{565}, y_{621}, y_{662}, y_{613}, y_{634}, y_{665}, y_{731}, y_{752}, y_{733}, y_{714}, y_{715}		
3	z_{jt}	z_{22}, z_{53}, z_{63}, z_{35}, z_{45}		
4	z'_{jt}	z'_{62}, z'_{13}, z'_{65}, z'_{75}		

that decreasing the relocation cost has more utility than increasing covered demand. Similar comments can be expressed on Tables 7–9.

SA algorithm was run on various examples and the results were compared with the results of the example described in the problem formulation (Eqs. (2.3)–(2.14)) when solved by GAMS software (Table 10).

The column headed by %Gap, which is related to covering objective function, is determined as follows:

$$\%\text{Gap} = \frac{\text{GAMS upper bound} - \text{SA}}{\text{GAMS upper bound}} \times 100.$$

The column headed by %Gap, which is related to relocation cost objective function is determined as follows:

$$\%\text{Gap} = \frac{\text{SA} - \text{GAMS lower bound}}{\text{SA}} \times 100.$$

As was shown, recommended SA algorithm could provide suitable solutions within an acceptable time (see Table 10 and Figure 5), as compared to those of GAMS software. In Figures 6 and 7, the terms GAMS upper bound, GAMS lower bound, SA covered demand and SA relocation cost indicate upper bound of coverage objective function in GAMS software, lower bound of relocation cost objective function, coverage objective function value obtained from the SA algorithm, and relocation cost objective function value obtained from the SA algorithm, respectively.

5. Conclusion and recommendations for future research

This study examined the gradual maximal covering location problem with variable radius over multiple time periods. An overview of the research was presented, including definitions, necessity and objectives, assumptions and innovation of the problem. Following a literature review, the problem model was presented. Then, a simulated annealing algorithm was described and its sequential steps were stipulated.

TABLE 10. Comparing SA algorithm results with those of GAMS software.

Problem number	(I, J, T)**	P	GAMS solution time (s)*	Coverage objective function			Relocation cost objective function		
				GAMS upper bound	SA	% Gap	GAMS lower bound	SA	% Gap
1	$(5, 5, 3)$	2	0.5	1453	1384	4.7	809	809	0.0
2	$(7, 7, 3)$	4	1.6	1874	1790	4.5	3226	3954	18.4
3	$(10, 10, 3)$	5	3.4	2728	2202	19.3	3776	5476	31.0
4	$(12, 12, 3)$	5	10.4	3011	2729	9.4	3755	5300	29.2
5	$(14, 14, 3)$	6	17.5	3263	2916	10.6	3680	4749	22.5
6	$(15, 15, 3)$	6	21	4041	3183	21.2	3680	4825	23.7
7	$(17, 17, 3)$	7	31	4458	3369	24.4	3456	4499	23.2
8	$(20, 20, 3)$	9	46	4834	3830	20.8	3456	4530	23.7
9	$(21, 21, 3)$	9	62	5409	3868	28.5	5802	9319	37.7
10	$(22, 22, 3)$	10	75	5664	4129	27.1	5744	8909	35.5
11	$(24, 24, 3)$	10	98	6192	4643	33.4	5880	9056	35.1
12	$(25, 25, 3)$	11	119	6425	4836	32.9	6450	9438	31.7
13	$(26, 26, 3)$	12	153	6838	4996	36.9	6890	10 112	31.9
14	$(28, 28, 3)$	12	194	7392	5321	38.9	7084	11 295	37.3
15	$(30, 30, 3)$	12	262	7830	5623	39.2	7470	12 496	40.2
16	$(40, 40, 3)$	14	346	9436	6742	40.0	8372	14 768	43.3
17	$(50, 50, 3)$	16	451	10 544	7158	47.3	10 768	19 743	45.5
18	$(65, 65, 3)$	20	580	13 640	8965	52.1	13 640	25 849	47.2
19	$(85, 85, 3)$	23	718	14 260	9064	57.3	14 076	30 064	53.2
20	$(100, 100, 3)$	25	868	16 100	9638	67.0	18 575	44 971	58.7
21	$(5, 5, 5)$	3	1.3	2284	2227	2.5	4575	6257	26.9
22	$(7, 7, 5)$	4	5	2937	2589	11.8	5614	7440	24.5
23	$(10, 10, 5)$	5	12	4204	3498	16.8	14 640	16 532	11.4
24	$(12, 12, 5)$	5	28	5190	4147	20.1	8064	10 950	26.4
25	$(14, 14, 5)$	6	45	6099	4621	24.2	10 981	14 452	24.0
26	$(15, 15, 5)$	6	53	6271	4886	22.1	11 158	14 329	22.1
27	$(17, 17, 5)$	7	100	8040	5294	34.2	16 031	20 190	20.6
28	$(20, 20, 5)$	9	172	8352	5832	30.0	13 517	19 442	30.5
29	$(21, 21, 5)$	9	283	9572	5936	38.0	14 973	20 822	28.1
30	$(22, 22, 5)$	10	395	9522	6169	35.2	13 838	19 524	29.1
31	$(24, 24, 5)$	11	546	9888	6328	36.0	16 176	23 247	30.4
32	$(25, 25, 5)$	11	621	10 575	6893	34.8	14 475	20 894	30.7
33	$(26, 26, 5)$	12	953	11 232	7079	37.0	18 018	27 089	33.5
34	$(28, 28, 5)$	12	1328	12 292	7286	40.7	19 964	29 572	32.5
35	$(30, 30, 5)$	12	1649	13 380	7534	43.7	20 820	31 295	33.5
36	$(40, 40, 5)$	14	1970	18 102	8984	50.4	25 382	39 476	35.7
37	$(50, 50, 5)$	16	2359	22 224	9843	55.7	29 808	47 589	37.4
38	$(65, 65, 5)$	20	2812	29 240	11 289	61.4	37 840	62 972	39.9
39	$(85, 85, 5)$	23	3309	36 547	12 943	64.6	44 114	76 325	42.2
40	$(100, 100, 5)$	25	3872	42 800	13 891	67.5	48 025	89 326	46.2

* As already mentioned, none of these examples can provide a feasible solution.
** Each problem consists of the two index variables $x_{jt}, z_{jt}, z'_{jt}, r_{jt}, c'_{jt}, q_{jt}$ and the three index variables c_{ijt}, y_{ijt}, l_{ijt}. For example, a problem with $I = 5$, $J = 5$, $T = 3$ contains $3 \times 5 = 15$ variables x_{jt}, 15 variables z_{jt}, 15 variables z'_{jt}, 15 variables r_{jt}, 15 variables c'_{jt}, 15 variables q_{jt}, $3 \times 5 \times 5 = 75$ variables c_{ijt}, 75 variables y_{ijt}, 75 variables l_{ijt}. Total number of variables for this problem is 315. In this way, the size of other problems can be determined based on the number of decision variables.

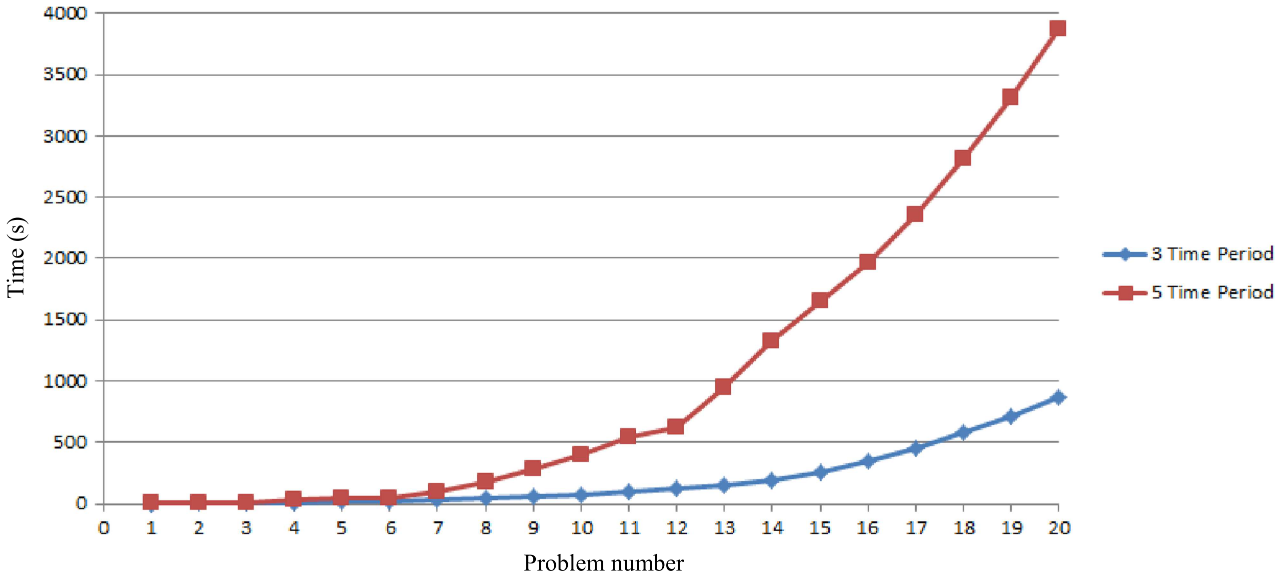

FIGURE 5. GAMS solution time for different problem sizes.

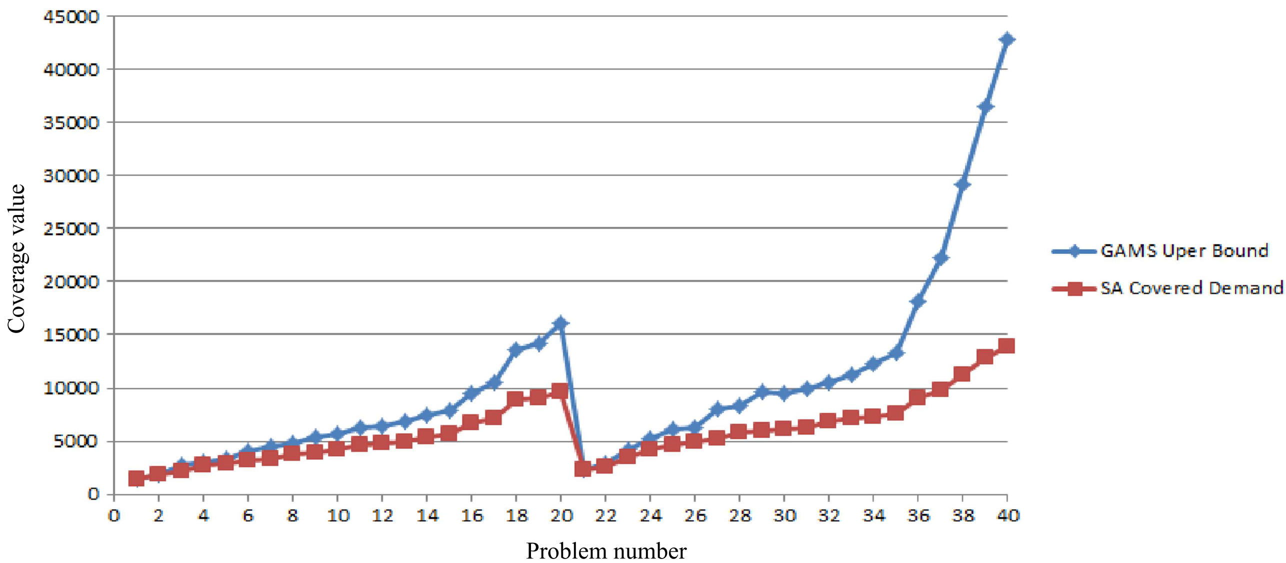

FIGURE 6. Comparison of GAMS upper bound with SA values.

Since some of unrealistic assumptions of the covering problems were no more considered in this problem, it can be referred to as one of the most important problems in the field of facility location. In this study, a mixed integer programming model was presented in which not only facility capacity was considered, but also two objectives were followed: coverage maximization and relocation cost minimization.

Since this problem is a mixed integer nonlinear programming (MINLP) and NP-Hard problem, GAMS software was seen to fail to generate suitable solutions within an acceptable time span. However, the SA algorithm succeeded to provide good solutions within an acceptable period of time. The obtained results from GAMS

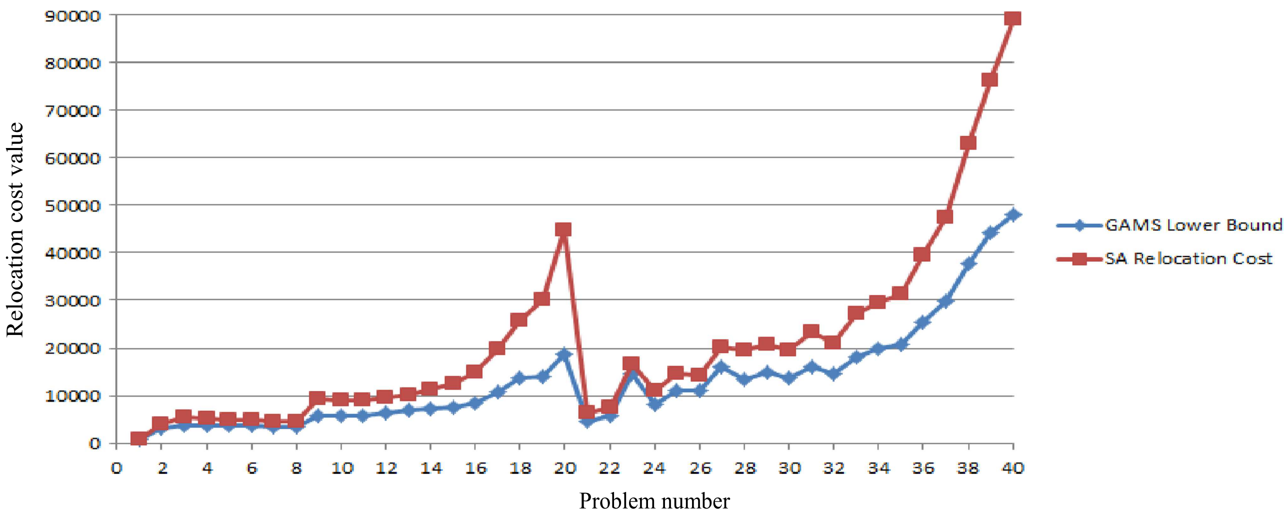

FIGURE 7. Comparison of GAMS lower bound with SA values.

software were not reliable as the software did not take the variable radius coverage concept into account. Considering the limitations of GAMS software, the results of SA algorithm were reliable. With such interpretations, the SA algorithm results are also acceptable for larger problems.

Considering previously conducted studies and in order to more precisely match the proposed model with real-world applications, the following directions are recommended for future research:

- Since the travel time is a function of several factors such as traffic, weather condition and path quality, in the real world, considering a variable travel time can contribute to achieving more realistic problems.
- In order to improve the SA algorithm for this problem, one can test new neighborhood definitions.
- Designing a method to achieve optimal solution can be investigated in future studies. With the second objective function and constraints (2.10) and (2.11) been nonlinear, one can try to linearize them.
- One can use other coverage functions like step coverage function, nonlinear coverage function, etc. for similar purposes.

References

[1] E. Aarts and J.K. Lenstra, Local Search in Combinatorial Optimization. Princeton University Press, New Jersey (2003).

[2] M. Albareda-Sambola, E. Fernandez, Y. Hinojosa and J. Puerto, The multi-period incremental service facility location problem. *Comput. Oper. Res.* **36** (2009) 1356–1375.

[3] R.H. Ballou, Dynamic warehouse location analysis. *J. Mark. Res.* **5** (1968) 271–276.

[4] O. Berman, Z. Drezner, D. Krass and G.O. Wesolowsky, The variable radius covering problem. *Eur. J. Oper. Res.* **196** (2009) 516–525.

[5] O. Berman and D. Krass, The generalized maximal covering location problem. *Comput. Oper. Res.* **29** (2002) 563–581.

[6] O. Berman, D. Krass and Z. Drezner, The gradual covering decay location problem on a network. *Eur. J. Oper. Res.* **151** (2003) 474–480.

[7] C. Canel, B.M. Khumawala, J. Law and A. Loh, An algorithm for the capacitated, multi-commodity multi-period facility location problem. *Comput. Oper. Res.* **28** (2001) 411–427.

[8] R.L. Church and K.L. Roberts, Generalized coverage models and public facility location. *Pap. Reg. Sci. Assoc.* **53** (1983) 117–135.

[9] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms. Indian Institute of Technology, Kanpur, India (2001).

[10] Z. Drezner, A. Suzuki and T. Drezner, Locating multiple facilities in a planar competitive environment. *J. Oper. Res. Soc. Jpn.* **50** (2007) 1001–1014.

[11] H.A. Eiselt and V. Marianov, Gradual location set covering with service quality. *Socioecon. Plan. Sci.* **43** (2009) 121–130.

[12] H.A. Eiselt and V. Marianov, Foundations of location analysis. *International Series in Operations Research & Management Science, Part IV, Covering Problem*, Springer (2011).

[13] R.Z. Farahani, N. Asghari, N. Heidari, M. Hosseinini and M. Goh, Covering problem in facility location: a review. *Comput. Ind. Eng.* **62** (2011) 368–407.

[14] GAMS Development Corporation, GAMS – A User's Guide. Tutorial by Richard E. Rosenthal, Washington, DC, USA (2008).

[15] GAMS Development Corporation, GAMS – The Solver Manuals. Tutorial by Richard E. Rosenthal, Washington, DC, USA (2008).

[16] M. Gendreau, G. Laporte and F. Semet, The maximal expected coverage relocation problem for emergency vehicles. *J. Oper. Res. Soc.* **57** (2006) 22–28.

[17] S.L. Hakimi, Optimum distribution of switching centers in a communication network and some related graph theoretic problems. *Oper. Res.* **13** (1965) 462–475.

[18] M.S. Jabal Ameli and B. Bankian Tabrizi, Maximal covering problem considering gradual coverage and variable coverage radius, in *7th International Conference of Industrial Engineering, Isfahan, Iran* (2010).

[19] O. Karasakal and E.K. Karasakal, A maximal covering location model in the presence of partial coverage. *Comput. Oper. Res.* **31** (2004) 1515–1526.

[20] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, Optimization by simulated annealing. *Science* **220** (1983) 671–680.

[21] N. Megiddo, E. Zemel and S.L. Hakimi, The maximum coverage location problem. *SIAM J. Algebra Discrete Method* **4** (1983) 253–261.

[22] B. Orbay, *An Interactive Evolutionary Algorithm for the Multi Objective Relocation Problem With Partial Coverage*. Dissertation, Middle East Technical University (2011).

[23] F. Rahim, *A Variable Neighborhood Search Procedure for the Combined Location With Partial Coverage and Selective Traveling Salesman Problem*. Dissertation, Middle East Technical University (2010).

[24] M.V. Rashidi and A. Jafari, Present nonlinear model for solving the MCLP considering gradual coverage constraint and facility capacity, in *International Conference on Nonlinear Modeling and Optimization (ICNMO), Amol, Iran* (2012).

[25] C. ReVelle, M. Scholssberg and J. Williams, Solving the maximal covering location problem with heuristic concentration. *Comput. Oper. Res.* **35** (2008) 427–435.

[26] O. Toreyen, *Hierarchical Maximal Covering Location Problem With Referral in the Presence of Partial Coverage*. Dissertation, Middle East Technical University (2007).

[27] G.O. Wesolowsky and W.G. Trucott, The multi period location–allocation problem with relocation of facilities. *Manag. Sci.* **22** (1975) 57–65.

[28] M.H.F. Zarandi, S. Davari and S.A.H. Sisakht, The large-scale dynamic maximal covering location problem. *Math. Comput. Model.* **57** (2013) 710–719.