Scientia Iranica E (2011) 18 (6), 1564–1570

SCIENTIA IRANICA

Sharif University of Technology

Scientia Iranica

Transactions E: Industrial Engineering

www.sciencedirect.com

The large scale maximal covering location problem

M.H. Fazel Zarandi [a,*], **S. Davari** [a], **S.A. Haddad Sisakht** [b]

[a] *Department of Industrial Engineering, Amirkabir University of Technology, 424 Hafez Avenue, Tehran, P.O. Box 15875-4413, Iran*

[b] *Department of Industrial Engineering, Sharif University of Technology, Tehran, P.O. Box 14588-89649, Iran*

Received 20 November 2010; revised 10 August 2011; accepted 20 September 2011

KEYWORDS

Analysis of variance (ANOVA);
Facility location;
Genetic algorithm;
Maximal covering location problem (MCLP);
Metaheuristics.

Abstract The maximal covering location problem (MCLP) is a challenging problem with numerous applications in practice. Previous publications in the area of MCLP proposed models and presented solution methodologies to solve this problem with up to 900 nodes. Due to the fact that in real-life applications, the number of nodes could be much higher, this paper presents a customized Genetic Algorithm (GA) to solve MCLP instances, with up to 2500 nodes. Results show that the proposed approach is capable of solving problems with a fair amount of exactness. In order to fine-tune the algorithm, Tukey's Least Significant Difference (LSD) tests are employed on a set of test problems.

1. Introduction and problem description

Facility location has been a fundamental area of research for over a century. It has a decisive role in the success of supply chains, with applications including locating gas stations, schools, plants, landfills, police stations, etc. A well-known location problem which has been studied since the very beginning of location science is the covering location problem. In a covering location problem, one seeks a solution to cover a subset of customers, using a pre-defined number of facilities, considering one or more objectives.

The covering location problem is often categorized as Location Set Covering Problem (LSCP) and Maximal Covering Location Problem (MCLP). In a classical MCLP, one seeks the location of a number of facilities on a network in such a way that the covered population is maximized. A facility covers a demand node, if it is established in a distance less than the threshold to the demand node. This pre-defined threshold is often called the coverage radius, which directly affects the solution of the problem.

MCLP was first introduced by Church and ReVelle [1] on a network, and since then, various extensions to the original problem have been made. Normally, MCLP is considered whenever there are insufficient resources or budget to cover the demand of all the nodes. Therefore, the decision maker determines a fixed budget/resource to cover the demands as much as possible. Figure 1 shows a sample solution to the problem with 20 facilities. In this figure, bold nodes represent the facilities, and circles are the coverage area of each facility. Moreover, smaller dots are the demand centers to be covered.

Many approaches have been proposed to solve various versions of MCLP. These include exact, heuristic and meta-heuristic methods, which are discussed in the next section. Obviously, the solution of large instances of MCLP is cumbersome, using exact methods. Therefore, heuristics and meta-heuristics have been employed in order to solve MCLPs of larger sizes. Recently, ReVelle et al. [2] presented heuristic concentration to solve MCLP, and solved it with 900 nodes. In this paper, we solve problems with up to 2500 nodes and show the performance of the proposed GA on a set of test problems. Our experiments show that GA is able to solve large-scale problems with negligible errors in a fair amount of time.

To begin presenting the model of this paper, some assumptions are presented. It is assumed that a node is covered whenever there is at least one facility within a pre-defined distance of it. Furthermore, each node could host a facility. In other words,

* Corresponding author. Tel.: +98 21 64 54 53 78; fax: +98 21 66 41 30 25.
E-mail address: zarandi@aut.ac.ir (M.H. Fazel Zarandi).
URL: http://cil.aut.ac.ir (M.H. Fazel Zarandi).

doi:10.1016/j.scient.2011.11.008

Production and hosting by Elsevier

ELSEVIER

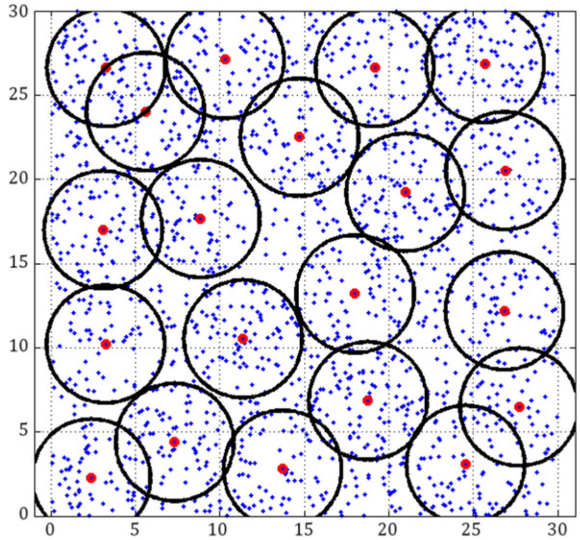

Figure 1: A sample solution to the covering location problem.

the set of nodes and the set of potential facilities are identical. Also, it is assumed that each node has a specific amount of demand to be met. Besides, proximity of the population to these facilities is desirable. This means that facilities such as landfills are excluded from the scope of this paper. In addition, an exogenous version of MCLP is considered. Hereby, the mathematical program of MCLP is given. First, we define problem parameters and variables as discussed in [2] (for further information about the formulation, interested readers can refer to [2]):

- i, I The index and set of demand nodes,
- j, J The index and set of eligible facility sites,
- a_i The population or demand at node i
- d_{ij} The shortest distance (or time) from demand node i to facility at node j,
- S The distance (or time) standard within which coverage is expected,
- N_i $\{j \mid d_{ij} \leq S\}$ = the nodes j that are within a distance of S to node i,
- p The number of facilities to be established,
- x_j A binary variable that equals one when a facility is sited at the jth node and zero otherwise,
- y_i A binary variable which equals one if node i is covered by one or more facilities stationed within S and zero otherwise,

$$\text{Maximize } z = \sum_{i \in I} a_i y_i, \tag{1}$$

$$\text{Subject to} : y_i \leq \sum_{j \in N_i} x_j, \quad i \in I, \tag{2}$$

$$\sum_{j \in J} x_j = p, \tag{3}$$

$$0 \leq y_i \leq 1 \quad i \in I, \tag{4}$$

$$x_j \in \{0, 1\} \quad j \in J. \tag{5}$$

The rest of the paper is organized as follows: a concise literature review of covering problems and related issues is presented in Section 2. Section 3 is dedicated to the proposed solution algorithm. Numerical examples appear in Section 4. Finally, conclusions and outlooks for potential future research are given in Section 5.

2. Literature review

The large number of publications in the area of MCLP prevents a comprehensive review in this paper. Thus, some of the main contributions to the field are discussed in this section. Two valuable and recent reviews by ReVelle et al. [3] and Berman et al. [4] may be consulted for further information regarding the subject.

MCLP was first presented by Church and ReVelle [1] in the mid 70s. Since then, numerous papers about various types of the problem have appeared. Some examples are the proposal of hierarchical MCLP by Moore and ReVelle [5], modeling the MCLP in competitive environments by Plastria and Vanhaverbeke [6], gradual covering models by Church and Roberts [7], and Berman et al. [8,9], using GIS in MCLP by Alexandris and Giannikos [10], and covering nodes, using parallelograms by Younies and Wesolowski [11], to name a few.

Another stream of research in the field of MCLP deals with employing heuristics and metaheuristics to tackle the computational hurdle of solving MCLP models. Table 1 presents some of the main non-exact solution algorithms that have been used in order to solve variants of MCLP from the beginning of the 21st century. This review apparently shows that metaheuristics have been an attractive area of research to solve MCLP models.

As far as we are concerned, the literature on real-world case studies of MCLP is comparatively limited. A case to ameliorate the accessibility of demand nodes to fixed facilities has been considered by Murawski and Church [25], which is called the "maximal covering network improvement problem", and which has been employed in Ghana as a real-world case. In another paper, Curtin et al. [26] studies the distribution of police patrol areas in Dallas, USA. In addition, Ratick et al. [27] can be considered as another real-world case study in the Kohat district of Pakistan, by applying the model of Moore and ReVelle [5]. Recently, Basar et al. [24] surveyed the multi-period version of MCLP to locate emergency services in Istanbul, Turkey.

A recent interest in MCLP has arisen about the uncertainty of parameters, such as demands or even the location of demand nodes. Batanovic et al. [28] suggested a MCLP in networks with uncertainty, and studied the problem with three different types of demand node weight:

(a) Equal weights.
(b) Relative deterministic weights.
(c) Linguistic terms as the demand weights.

Moreover, Corrêa et al. [21] is another recent publication which addresses the problem of MCLP, considering congestion in queue lines. Araz et al. [29] presented a fuzzy goal programming approach for locating emergency service facilities. A recent paper by Berman et al. [30] studies the gradual covering problem with uncertain demands. In another publication by Berman and Wang [31], they surveyed a case in which the demands are random variables with unknown probability distributions. Fazel Zarandi et al. [32] is another recent attempt in modeling covering problems in the fuzzy sense. They modeled a closed-loop supply chain network with coverage considerations, and used fuzzy goal programming to solve it.

To the best of the authors' knowledge, the only attempts to model and solve MCLP of larger sizes are ReVelle et al. [2] and Park and Ryu [15]. Although these two papers are valuable contributions to the literature, the problem sizes which were targeted are below 900 nodes, in both papers. Moreover, it is

Table 1: Some heuristics and metaheuristics for various types of MCLP.

Author(s)	Problem	Year
Galvão et al. [12]	MCLP	2000
Aytug and Saydam [13]	Maximum expected coverage	2002
Espejo et al. [14]	Hierarchical MCLP	2003
Park and Ryu [15]	Large-scale MCLP	2004
Karasakal and Karasakal [16]	Partial covering	2004
Shavandi and Mahlooji [17]	Fuzzy queuing maximal covering location	2006
Rajagopalan et al. [18]	Multi-period set covering problem	2008
ReVelle et al. [2]	MCLP	2008
Qu and Weng [19]	Hub-maximal covering location problem	2009
Canbolat and von Massow [20]	MCLP on the plane using ellipses	2009
Corrêa et al. [21]	Probabilistic covering location–allocation problem	2009
Lee and Lee [22]	Hierarchical MCLP	2010
Davari et al. [23]	MCLP with fuzzy travel times	2011
Basar et al. [24]	Multi-period double coverage	2011

worth noting that while Park and Ryu [15] considered a case in which the set of facilities and demand nodes were not identical, this paper deals with a case where each demand node can host a facility. Therefore, the problem in this paper is quite different from Park and Ryu [15], and appears to fill the gap of solving large-scale MCLPs in the literature by applying an efficient GA to solve MCLP instances of up to 2500 nodes. Indeed, the paper deals with a solution of the MCLP model proposed by ReVelle et al. [2] for very large-scale problems, where exact methods are handicapped.

3. The proposed solution procedure

3.1. Review of genetic algorithm

GA is a bio-inspired optimization procedure which was first proposed by Holland [33]. From the mid 70s to date, various types of GA have been employed in problems such as vehicle routing problems [34], facility location [35], portfolio selection [36], scheduling [37], and quadratic assignment problems [38]. The main thrust of GA is to improve generations gradually by applying reproduction mechanisms, such as crossover and mutation. Each generation consists of a set of individual solutions and each iteration involves selection of a set of parents, based on their fitness value and application of reproduction schemes to generate a set of offspring. Based on a selection strategy, the population is updated and the process is followed, until the termination criterion is met.

The main motivations to use GA in this paper are twofold. First, GA has been proven to be one of the best methods to solve facility location models. Additionally, our preliminary experiments showed that GA is a better approach to solve MCLP, compared to local search procedures such as simulated annealing and the tabu search. The rest of this section is devoted to elaboration of the proposed GA.

3.2. Encoding scheme

Representation is one of the main steps towards developing an efficient GA, since it affects the type of crossover and mutation and also the runtime of the algorithm. Although many different approaches are able to represent a solution of MCLP, herein a binary vector representation is employed, in which each bit represents the status (open/close) of the node. In other words, a one value in the ith bit of a solution string means that there is a facility at node i. Clearly, there should be exactly p non-zero values in each solution representation. It should be noted that using the solution vector, travel time/distance matrix and covering radius, one can easily find the set of covered nodes.

One special feature of basic MCLP is that any generated solution is feasible, as long as p facilities are located. This special characteristic is a great advantage which facilitates searching the solution space considerably. Since the sizes of problems targeted in this paper are very large, there is a need to devise a method to calculate the fitness of a solution in a fast and easy way. Hence, a similar approach to ReVelle et al. [2] has been employed.

3.3. Selection

In this paper, the Roulette Wheel Selection (RWS) has been exercised as the selection approach. RWS mimics a roulette wheel where the area of each section is proportional to the individual's fitness, and selection is made randomly, based on probabilities corresponding to each area.

3.4. Reproduction

In order to search for better solutions, we define the set, $N(X)$, to be the set of solutions neighboring a solution, X. In each iteration, the next solution, Y, is generated from $N(X)$ by using a crossover, mutation or migration operator. While a crossover is carried out when two chromosomes are mated together to produce a new solution, a mutation is employed to make a diversification on a solution. These moves are discussed further in Section 4.2.

4. Numerical examples

4.1. Test problems

To generate test problems, a similar procedure as ReVelle et al. [2] has been followed. The locations of the nodes in the test problems are randomly generated using a uniform distribution between 0 and 30 for x and y coordinates. The distances between the nodes are then defined as their Euclidean distances. Using the Euclidean measure, it is guaranteed that the triangular inequality holds for the network. Populations on the nodes are randomly generated, using a uniform distribution between [0, 100]. The procedure has been used for problems with 1800 and 2500 nodes. It should be noted that while ReVelle et al. [2] solved problems with less than 900 nodes, we considered problems of larger sizes up to 2500 nodes. The set of test problems have been given in Table 2. The CPLEX

Table 2: Test problems of this paper.

Number of nodes	Number of facilities	Coverage radius
1800	15, 20, 25	3.5, 3.75, 4
2500	15, 20, 25	3.5, 3.75, 4

commercial solver has been used to solve these problems, and results are compared against those obtained from the proposed GA.

4.2. Parameter setting

First, each problem was solved ten times, using the proposed GA. Then, the worst, average and the best solutions were considered to select the optimal combination of parameters. We came to notice that a crossover ratio of 0.7 leads to a good solution. Furthermore, the rate of migration was set to 0.05, and there was a 0.25 chance of using mutation to get new solutions. In addition, we realized that after about 30,000 iterations, the algorithm reaches satisfactory results, and there is almost no sign of improvement in any run. Hence, the stopping criterion of the algorithm was set to reach 30,000 iterations. It should be noted that in order to speed-up the process of search, and considering the fact that locations are fixed, the total coverage of a node is calculated in the beginning of the algorithm and stored in a vector called "CoverageValue". This vector is of high value in the performance of GA and its value does not change throughout the algorithm. Therefore, it can save much solution time in fitness evaluation.

To report the results of the tuning phase, each parameter setting is encoded using a string of three ratios, as e–f–g, where e denotes the initialization type, f is the crossover, and g is the mutation type. Two levels for each GA mechanism (initialization, crossover, mutation) have been defined as follows:

$e(1)$: Random selection of p facilities out of n.
$e(2)$: Selection of p facilities out of n based on the RWS method using the "CoverageValue" vector.
$f(1)$: The two parents are combined, duplicate values are removed and p facilities are selected based on RWS and the "CoverageValue" vector.
$f(2)$: A two-point crossover is conducted. Then, if the number of facilities is more than p, facilities are removed randomly till p facilities remain. Moreover, if the offspring have less than p facilities, the remaining facilities are added randomly.
$g(1)$: Removing a random facility and adding another random facility.
$g(2)$: Two different procedures are followed with the same probability of being used. The first type replaces the facility with the least coverage with a random node. In the second type, a random facility is removed and the node with the greatest coverage in "CoverageValue", which is not currently being used, is added.

Figures 2 and 3 depict the Box–Whisker diagram of fitness and runtime of the algorithm using each parameter setting, respectively. These experiments have been run for the case with $n = 2500, p = 25$ and $S = 3.75$, which is one of the hardest instances studied in this paper. Figures 2 and 3 should be studied together in order to analyze the results. Obviously, while "1–1–1" leads to the best solution in terms of solution quality, the runtime needed is fair and competitive with other settings. This fact has led us to use this setting as the optimal setting to be used in this paper.

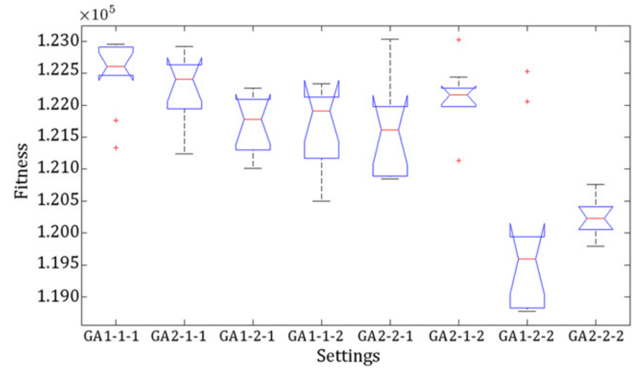

Figure 2: The Box–Whisker diagram of the solution quality for the eight solution alternatives.

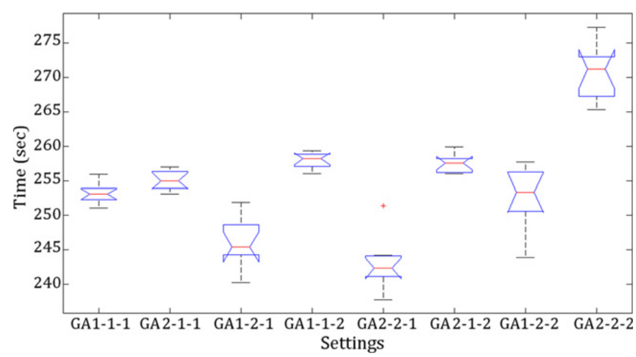

Figure 3: The Box–Whisker diagram of the time needed to solve each of the eight solution alternatives.

Table 3: The ANOVA table of the solution fitness.

Source	*SS*	*df*	*MS*	*F*	Prob > *F*
Columns	59 803 316	7	8 543 330.839	18.238	0.000
Error	33 726 475	2	468 423.259		
Total	93 529 791	79			

Table 4: The ANOVA table of the time needed.

Source	*SS*	*df*	*MS*	*F*	Prob > *F*
Columns	5153.286	7	736.184	88.831	0.000
Error	596.695	2	8.287		
Total	5749.981	79			

Moreover, we have conducted a set of Tukey's LSD tests to get more findings. The results clearly show that there is a significant difference between settings, as shown in Figure 4. This figure shows the 95% confidence interval of fitness for each parameter setting. Clearly, the setting, "GA 1–1–1", leads to the highest fitness value among the settings analyzed. Moreover, it is not significantly different from 'GA 2–1–1' and 'GA 2–1–2', because of the intersection of the lines. Anyhow, considering all the experiments already explained, setting "GA 1–1–1" has been selected as the optimal setting to be used.

Tables 3 and 4 show the results of running the ANOVA tests on all eight combinations of settings, after 10 runs for each. While Table 3 shows the results for solution quality, Table 4 deals with the time needed to solve problems. It should be noted that these results have been realized for the case with

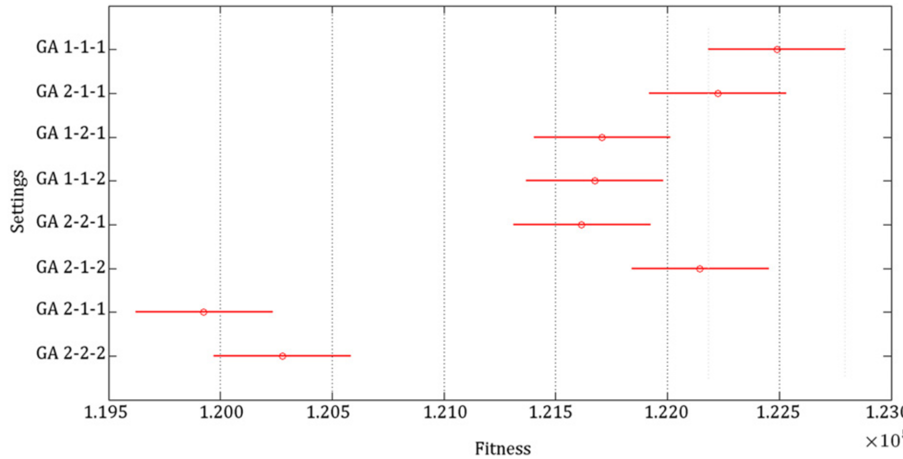

Figure 4: The result of the LSD test.

Table 5: Results for 1800 nodes.

p	S	CPLEX			Meta-heuristic			Gap (%)
		Z^*	Time (sec)	Coverage (%)	Z^*	Time (sec)	Coverage (%)	
	3.5	65 322	4	70.61	64 569	155	69.94	1.151
15	3.75	71 726	5	78.39	71 200	157	77.67	0.733
	4	77 988	10	84.83	77 630	149	84.55	0.459
	3.5	78 648	10 922	85.17	77 477	176	83.16	1.489
20	3.75	84 064	10 877	91.83	82 552	170	90.33	1.799
	4	88 025	10 846	96.72	86 874	166	94.94	1.307
	3.5	87 169	10 836	95.67	85 531	192	94.39	1.880
25	3.75	89 857	10 962	99.17	89 050	187	97.77	0.898
	4	90 340	15	100	89 789	181	99.39	0.609

Table 6: Results for 2500 nodes.

p	S	CPLEX			Meta-heuristic			Gap (%)
		Z^*	Time (sec)	Coverage (%)	Z^*	Time (sec)	Coverage (%)	
	3.5	87 723	15	68.32	86 876	224	68.08	0.965
15	3.75	96 806	24	75.88	96 712	218	75.56	0.097
	4	105 743	33	83.20	103 655	215	81.72	1.975
	3.5	108 221	267	84.68	106 606	250	83.72	1.492
20	3.75	115 173	10 800	91.44	112 892	246	89.72	1.980
	4	121 182	10 800	96.72	119 019	239	94.92	1.785
	3.5	121 100	10 800	95.84	118 692	273	94.12	1.988
25	3.75	124 453	10 776	99.24	122 268	274	97.40	1.756
	4	125 081	2 916	100	124 484	268	99.44	0.477

$n = 2500$, $p = 25$ and $S = 3.75$. The ANOVA test testifies that there is a significant difference among alternatives, from both fitness and runtime aspects.

4.3. Computer specifications

The proposed GA was programmed, using Visual C++ and run on a high-level computer system. In addition, CPLEX was used in order to solve the problem to optimality on the same computer.

4.4. Results, validation, and discussions

The test problems were solved using two approaches. First, each problem was solved using the CPLEX optimization software. Then, solutions were compared against the results of the proposed GA. Computational results of the problem are summarized in Tables 5 and 6. Results show that although the proposed algorithm is unable to solve most of the cases to optimality, it works well when solving problems of various sizes, and the errors are all below 2%. The first two columns of the tables represent the number of facilities to be established and the coverage radius, respectively. Results and the amount of time needed for running the proposed GA are reported in separate columns. Moreover, the errors of each problem are reported in the last column.

It is remarkable that the times needed to solve problems, using the proposed algorithm, are negligible compared to CPLEX in some cases. Tables 5 and 6 confirm that as the problem size grows, generally it takes more and more time for CPLEX to reach optimal solutions. This can be attributed to the combinatorial nature of MCLP. Despite this fact, the runtime of our proposed approach remains relatively the same. This is because of the fact that we run the algorithm for 30 000 steps, regardless of the test problem. Another notable finding of the numerical experiments is the great ability of GA to escape local optima. The efficiency of the proposed algorithm has been evidenced by yielded results. It means that the proposed algorithm works well.

It may be argued that since CPLEX reaches optimal solutions in all the cases of Tables 5 and 6, there is no need to use a heuristic or metaheuristic. Although this is true for most cases studied in this paper, it should be noted that for larger instances, CPLEX fails to reach optimality in much longer times. Moreover, solution quality is not the only measure of interest. When the solution algorithm is embedded within another algorithm such as simulation algorithms, the solution time becomes a valuable measure. Thus, using GA has considerable advantages over exact methods.

Moreover, a fair amount of consistency was observed in solutions. In other words, in most cases, the same subset of facilities is selected in two cases with different values of p. For instance, the optimal solutions of $p = 15$ and $p = 20$ for 2500 nodes and $\beta = 3.75$ share 12 facilities. Such a finding is of great importance from a practical point of view. This means that one may locate 12 facilities and in case of emergencies or for any other reason, the costs of adding new facilities are low, since there is no need to drastically change the locations.

5. Conclusion and future research areas

In this paper, the large-scale MCLP has been considered, and a GA has been proposed to solve it. We have shown that while our proposed approach is superior to the exact method, in terms of runtime, there are negligible errors compared to the optimal solutions. Although GA shows great performance in solving MCLP, a possible future study could be comparisons, using various heuristics/metaheuristics on this problem. Another avenue for future research is to assess the performance of GA for other variants of covering location problems, such as the set covering problem, or considering some parameters of the problem as fuzzy variables. Furthermore, MCLP could be enriched, adding some other assumptions such as gradual covering. The authors of this paper are currently working on adjustment of the proposed metaheuristic for the gradual covering location problem.

References

[1] Church, R.L. and ReVelle, C.S. "The maximal covering location problem", *Papers of the Regional Science Association*, 32, pp. 101–118 (1974).

[2] ReVelle, C., Scholssberg, M. and Williams, J. "Solving the maximal covering location problem with heuristic concentration", *Computers & Operations Research*, 35(2), pp. 427–435 (2008).

[3] ReVelle, C.S., Eiselt, H.A. and Daskin, M.S. "A bibliography for some fundamental problem categories in discrete location science", *European Journal of Operational Research*, 184(3), pp. 817–848 (2008).

[4] Berman, O., Drezner, Z. and Krass, D. "Generalized coverage: new developments in covering location models", *Computers & Operations Research*, 37(10), pp. 1675–1687 (2010).

[5] Moore, G.C. and ReVelle, C.S. "The hierarchical service location problem", *Management Science*, 28, pp. 775–780 (1982).

[6] Plastria, F. and Vanhaverbeke, L. "Discrete models for competitive location with foresight", *Computers & Operations Research*, 35(3), pp. 683–700 (2008).

[7] Church, R.L. and Roberts, K.L. "Generalized coverage models and public facility location", *Papers of the Regional Science Association*, 53, pp. 117–135 (1984).

[8] Berman, O., Kalcsics, J., Krass, D. and Nickel, S. "The ordered gradual covering location problem on a network", *Discrete Applied Mathematics*, 157(18), pp. 3689–3707 (2009).

[9] Berman, O., Krass, D. and Drezner, Z. "The gradual covering decay location problem on a network", *European Journal of Operational Research*, 151(3), pp. 474–480 (2003).

[10] Alexandris, G. and Giannikos, I. "A new model for maximal coverage exploiting GIS capabilities", *European Journal of Operational Research*, 202(2), pp. 328–338 (2010).

[11] Younies, H. and Wesolowsky, G.O. "A mixed integer formulation for maximal covering by inclined parallelograms", *European Journal of Operational Research*, 159(1), pp. 83–94 (2004).

[12] Galvão, R.D., Gonzalo Acosta Espejo, L. and Boffey, B. "A comparison of Lagrangean and surrogate relaxations for the maximal covering location problem", *European Journal of Operational Research*, 124(2), pp. 377–389 (2000).

[13] Aytug, H. and Saydam, C. "Solving large-scale maximum expected covering location problems by genetic algorithms: a comparative study", *European Journal of Operational Research*, 141(3), pp. 480–494 (2002).

[14] Espejo, L.G.A., Galvão, R.D. and Boffey, B. "Dual-based heuristics for a hierarchical covering location problem", *Computers & Operations Research*, 30(2), pp. 165–180 (2003).

[15] Park, T. and Ryu, K.L. "Exploiting unexpressed genes for solving large-scaled maximal covering problems", *Lecture Notes in Computer Science*, 3157, pp. 342–351 (2004).

[16] Karasakal, O. and Karasakal, E.K. "A maximal covering location model in the presence of partial coverage", *Computers & Operations Research*, 31(9), pp. 1515–1526 (2004).

[17] Shavandi, H. and Mahlooji, H. "A fuzzy queuing location model with a genetic algorithm for congested systems", *Applied Mathematics and Computation*, 181(1), pp. 440–456 (2006).

[18] Rajagopalan, H.K., Saydam, C. and Xiao, J. "A multiperiod set covering location model for dynamic redeployment of ambulances", *Computers & Operations Research*, 35(3), pp. 814–826 (2008).

[19] Qu, B. and Weng, K. "Path relinking approach for multiple allocation hub maximal covering problem", *Computers & Mathematics with Applications*, 57(11–12), pp. 1890–1894 (2009).

[20] Canbolat, M.S. and von Massow, M. "Planar maximal covering with ellipses", *Computers & Industrial Engineering*, 57(1), pp. 201–208 (2009).

[21] Corrêa, F.D.A., Lorena, L.A.N. and Ribeiro, G.M. "A decomposition approach for the probabilistic maximal covering location–allocation problem", *Computers & Operations Research*, 36(10), pp. 2729–2739 (2009).

[22] Lee, J.M. and Lee, Y.H. "Tabu based heuristics for the generalized hierarchical covering location problem", *Computers & Industrial Engineering*, 58(4), pp. 638–645 (2010).

[23] Davari, S., Fazel Zarandi, M.H. and Hemmati, A. "Maximal covering location problem (MCLP) with fuzzy travel times", *Expert Systems with Applications*, 38, pp. 14535–14541 (2011).

[24] Basar, A., Catay, B. and Unluyurt, T. "A multi-period double coverage approach for locating the emergency medical service stations in Istanbul", *Journal of the Operational Research Society*, 62, pp. 627–637 (2011).

[25] Murawski, L. and Church, R.L. "Improving accessibility to rural health services: the maximal covering network improvement problem", *Socio-Economic Planning Sciences*, 43(2), pp. 102–110 (2009).

[26] Curtin, K.M., Hayslett, K. and Qiu, F. "Determining optimal police patrol areas with maximal covering and backup covering location models", *Networks and Spatial Economics*, 10(1), pp. 125–145 (2007).

[27] Ratick, S.J., Osleeb, J.P. and Hozumi, D. "Application and extension of the Moore and ReVelle hierarchical maximal covering model", *Socio-Economic Planning Sciences*, 43(2), pp. 92–101 (2009).

[28] Batanovic, V., Petrovic, D. and Petrovic, R. "Fuzzy logic based algorithms for maximum covering location problems", *Information Sciences*, 179(1–2), pp. 120–129 (2009).

[29] Araz, C., Selim, H. and Ozkarahan, I. "A fuzzy multi-objective covering-based vehicle location model for emergency services", *Computers & Operations Research*, 34(3), pp. 705–726 (2007).

[30] Berman, O., Krass, D. and Wang, J. "The probabilistic gradual covering location problem on a network with discrete random demand weights", *Computers & Operations Research*, 38(11), pp. 1493–1500 (2011).

[31] Berman, O. and Wang, J. "The minmax regret gradual covering location problem on a network with incomplete information of demand weights", *European Journal of Operational Research*, 208(3), pp. 233–238 (2011).

[32] Fazel Zarandi, M.H., Haddad Sisakht, A. and Davari, S. "Design of a closed-loop supply chain (CLSC) model using an interactive fuzzy goal programming", *The International Journal of Advanced Manufacturing Technology*, 56, pp. 809–821 (2011).

[33] Holland, J.H., *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor (1975).

[34] Ghoseiri, K. and Ghannadpour, S.F. "Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm", *Applied Soft Computing*, 10(4), pp. 1096–1107 (2010).

[35] Topcuoglu, H., et al. "Solving the uncapacitated hub location problem using genetic algorithms", *Computers & Operations Research*, 32(4), pp. 967–984 (2005).

[36] Soleimani, H., Golmakani, H.R. and Salimi, M.H. "Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm", *Expert Systems with Applications*, 36(3), pp. 5058–5063 (2009).

[37] Balin, S. "Non-identical parallel machine scheduling using genetic algorithm", *Expert Systems with Applications*, 38(6), pp. 6814–6821 (2011).

[38] Liu, L. and Li, Y. "The fuzzy quadratic assignment problem with penalty: new models and genetic algorithm", *Applied Mathematics and Computation*, 174(2), pp. 1229–1244 (2006).

Mohammad Hossein Fazel Zarandi is Professor at the Department of Industrial Engineering of Amirkabir University of Technology, Tehran, Iran, and a member of the Knowledge-Information Systems Laboratory of the University of Toronto, Toronto, Ontario, Canada. His main research interests focus on: intelligent information systems, soft computing, computational intelligence, fuzzy sets and systems, multi-agent systems, networks, meta-heuristics and optimization. Professor Fazel Zarandi has published many books, scientific papers and technical reports in the above areas, most of which are accessible on the web. He has taught several courses in fuzzy systems engineering, decision support systems, management information systems, artificial intelligence and expert systems, systems analysis and design, scheduling, neural networks, simulations, and production planning and control, in several universities in Iran and North America.

Soheil Davari is a current Ph.D. degree student of Industrial Engineering at AmirKabir University of Technology, Tehran, Iran. His research interests focus on supply chain management, uncertainty theory, fuzzy logic and simulation techniques. He is an IEEE member and has published more than 15 papers in international journals and seminars around the world.

Seyyed Ali Haddad Sisakht is currently an M.S. degree student in Sharif University of Technology. He is an active researcher in fields such as logistics and supply chain management, management information systems, fuzzy logic and intelligent algorithms in industrial engineering.