Computing and Informatics, Vol. 42, 2023, 1471–1490, doi: 10.31577/cai_2023_6_1471

SOLVING THE TWO-LEVEL HIERARCHICAL COVERING LOCATION PROBLEM WITH AN ELECTROMAGNETISM-LIKE METAHEURISTIC

Milica Milivojević Danas

Faculty of Engineering, University of Kragujevac
Sestre Janjić 6, 34 000 Kragujevac, Serbia
e-mail: milica.milivojevic@kg.ac.rs

Milena Bogdanović

Faculty of Information Technologies, Belgrade Metropolitan University
Tadeuša Koušćuška 63, 11 000 Belgrade, Serbia
e-mail: milena.bogdanovic@metropolitan.ac.rs

Abstract. In this paper, an electromagnetism-like approach (EM) for solving the two-level hierarchical covering location problem (TLHCLP) is proposed. An EM metaheuristic is a powerful algorithm for global optimization that converges rapidly to the optimum. Therefore, it has the potential to solve this type of problem since movement based on the attraction-repulsion mechanisms, combined with the proposed scaling technique, directs EM to promising search regions. The fast implementation of the objective function and local search procedure for TLHCLP additionally improves the efficiency of the overall EM system. The proposed EM approach reaches all optimal solutions in a relatively short amount of computational time. EM also obtains high-quality solutions for large-scale problem instances that are out of reach for exact methods.

Keywords: Hierarchical location problems, covering models, electromagnetism-like metaheuristic, combinatorial optimization

1 INTRODUCTION

The network-based facility location modeling has been very popular in the last two decades since it has many practical applications in diverse areas such as telecommunications and computer networking, health-care systems, supply chain, waste management, etc. Only several recent applications will be mentioned, such as unpopular location problems on networks [1] and reverse logistics facility location [2].

Location models are usually application specific, i.e. there is no general location model that is appropriate for all existing and potential discrete location problems. Much work has been done in the facility location modeling because new and flexible location models are being formulated. It will be adequate for various applications and for developing efficient solution techniques to solve more general models.

One specific branch of these problems are covering location models. They are very suitable in the case related to the location of emergency facilities. A demand area is covered if it is within a predefined service distance from at least one of the existing facilities. A primary objective is to establish some of the potential facilities, in order to cover as much of the potential customer demand as possible. A recent review of various covering problems in facility location can be seen in [3].

The importance of the facility location modeling cannot be overstated, as it has numerous practical applications across different industries. One of the significant advantages of network-based facility location modeling are that it can provide insights into the optimal location of facilities based on network considerations, such as minimizing transportation costs and maximizing accessibility to customers. However, developing effective location models can be challenging since they need to be tailored to specific applications. Therefore, researchers have focused on developing new and flexible location models that can be adapted to different scenarios, as well as efficient solution techniques for solving them. One type of location model which has gained attraction is the covering location model. This model is particularly relevant to emergency facilities, where the objective is to cover as much demand as possible within a predefined service distance. Researchers have extensively studied various covering problems in facility locations, as highlighted in [3].

In this paper, the authors present an electromagnetism-like approach (EM) for solving facility location problems. The electromagnetism-like approach (EM) is applied to this problem for the first time and represents a completely new method for solving the problem.

The two-level hierarchical covering location problem (TLHCLP) is a complex optimization problem which involves two levels of decision-making: locating facilities at the first level and assigning customers to these facilities at the second level.

EM is a population-based optimization method that is inspired by the physics of electromagnetism. It simulates the behaviour of charged particles which are attracted to each other by electrostatic forces and repelled by like charges. In the context of optimization, EM algorithms use these forces to move candidate solutions in search of better solutions. The use of EM in solving the TLHCLP allows for a more efficient and effective search for optimal solutions. The algorithm can explore the search space more thoroughly and quickly than traditional optimization methods, such as linear programming or genetic algorithms. Additionally, the use of EM allows for the inclusion of multiple objectives in the optimization process, such as minimizing total cost and maximizing coverage.

Overall, the use of an EM metaheuristic to solve the TLHCLP represents a novel and promising approach to solving this complex optimization problem.

This approach is described in detail in Section 4, and the computational results are presented in Section 5. The authors also provide directions for future work in Section 6, emphasizing the need to investigate how the EM approach can be applied to other types of location models. Overall, this paper contributes to the ongoing efforts to develop effective location models and solution techniques for solving them.

The rest of the paper is organized as follows. In Section 2, previous work is presented, while the existing mathematical formulations are given in Section 3. The next two sections contain the main features of the electromagnetism-like approach (EM) and computational results on existing and new generated instances, respectively. Section 6 contains conclusions and directions for future work.

2 PREVIOUS WORK

One well-known and important case of the covering location problems is the maximal covering location problem (MCLP). A node is covered if there exists an established facility within a predefined coverage radius. The objective is to maximize the total coverage by establishing a fixed predefined number of facilities. For more information about this problem and its applications, the interested reader could refer to [4].

In some situations, locations are not equal, but they are in some type of hierarchy. For example, health care facilities have several levels of services. In the case of health care, clinics as lower-level facilities provide only a basic service, while hospitals can provide both lower and higher levels of services. The hierarchy is backwards inclusive since every facility on a certain level provides its own level of service and all lower level services. Another example is production-distribution systems consisting factories and warehouses, where a given product can be shipped to a client directly from the factory or through one of the warehouses. A third example is higher education where technical schools can cover applied studies, while universities can cover both academic and applied studies. In this paper, we consider a two-level hierarchical covering location problem (TLHCLP), as a direct generalization of the maximal covering problem, since two levels of hierarchy are usually enough for practical applications.

Moore and ReVelle in [5] first proposed a MLHCLP and applied it to the health services of Honduras. In their model, the lower-level facilities (clinics) provide only a level-one service, whereas the higher-level facilities (hospitals) provide both the level-two service and the level-one service. This hierarchy is said to be successively inclusive in the sense that a facility provides its own level of service and all lower levels of services. The problem was formulated as a 01 programming problem and used the linear programming relaxation of their formulation how to solve it. The authors recommend that this relaxation should be supplemented, where necessary, by a branch-and-bound algorithm. However, their computational experience was restricted to a test network developed from the provinces in Honduras, for which no use of a branch-and-bound algorithm is reported.

In [6], five heuristic approaches based on a combined Lagrangean surrogate (LS) relaxation were proposed, which reduce TLHCLP to a 01 knapsack problem. Tests were carried out using a sub-gradient-based heuristic incorporating the LS relaxation, with the resulting knapsack problems being solved both with and without the integrality constraints relaxed. Results were obtained for test problems available in the literature ranging from 55-node to 700-node networks. These were compared, where possible, with exact results obtained using CPLEX. It was found that computing times were reasonable.

Genetic algorithm for solving TLHCLP was proposed in [7, 8]. The binary encoding with new genetic operators which keep the feasibility of individuals was proposed. Computational performance is additionally improved by caching technique. A modification to resolve the problem of frozen bits in the genetic code was proposed and tested. The genetic algorithm was tested, its parameters were adjusted on instances up to 100 nodes. It performed well and proved robust and the optimal solutions were reached in all cases.

The description of all contributions in the literature about hierarchical facility location models is out of the scope of this paper, so only several of them will be mentioned which are devoted to similar but still slightly different problems:

- Hierarchical maximal covering location problems in the presence of partial coverage were studied in [9, 10];
- A coherent hierarchical covering location problem was proposed in [11];
- The two-stage maximal covering location problem was used for missile maintenance activity in [12].
- The bi-level maximal covering location problem considers two decision levels, one associated with facility location, and the other related to the allocation of customers to open facilities in [13].

Interested readers can also consult the excellent survey about various hierarchical facility location models in [14].

3 MATHEMATICAL FORMULATION

Let $J = \{1, 2, \dots, n\}$ be the set of nodes that represent demand areas, $I = \{1, 2, \dots, m\}$ be the set of potential facility sites, d_{ij} be the distance between potential facility i and node j, f_j be the weight of node j, and R_1, T_1 and R_2 be the cover radia. A node is level-1 service covered if its distance from some level-1 established facility is not more than R_1 or if the distance from some level-2 established facility is not more than T_1. Similarly, a node is level-2 service covered if its distance from some level-2 established facility is not more than R_2. The objective of TLHCLP is to establish exactly p level-1 facilities and q level-2 facilities to maximize the sum of weights of completely service-covered nodes (which are both level-1 and level-2 service covered).

In real life, people may be willing to travel a greater distance to receive the same service from a facility with more resources (a higher-level facility). Therefore, the service distance for the level-2 facility T_1 is expected to satisfy $T_1 > R_1$. However, in practice, people may also be willing to travel further to receive the more sophisticated level-2 service. Therefore, in the TLHCLP model, it is considered that $R_1 < T_1 < R_2$.

The two-level hierarchical covering location problem considered in the paper is common in some real-life cases. Practical aspects can be seen in the following example: shopping in small and large markets, such as mini markets and Metro. The lower level, such as mini markets, can provide only smaller quantities of products, while Metros (higher level), in addition to small quantities, can also provide much larger quantities of products. As can be seen from the above, the two levels of hierarchy are generally sufficient to describe real problems.

Example 1. Let the number of objects be $n = 5$, where on the first and second level can be determined, i.e. $p = 1$ and $q = 1$. Let be the radii $R_1 = 15, T_1 = 20$ and $R_2 = 30$. Let be the cost of covering the objects $f = [11, 10, 9, 5, 12]$, (f_i – the cost of covering the object i) and the distance matrix between objects d given below (also in Figure 1)

$$d = \begin{bmatrix} 0 & 13 & 28 & 19 & 33 \\ 13 & 0 & 31 & 32 & 22 \\ 28 & 31 & 0 & 35 & 9 \\ 19 & 32 & 35 & 0 & 44 \\ 33 & 22 & 9 & 44 & 0 \end{bmatrix}.$$

Optimal solution value is 33 and it is obtained by total enumeration.

At level-1 ($p = 1$) facility 3 is established.

At level-2 ($q = 1$) facility 2 is established.

At level-2 ($q = 1$) nodes 1, 2 and 5 are covered. ($d_{21} = 13 \leq 30, d_{22} = 0 \leq 30$ and $d_{25} = 22 \leq 30$).

At level-1 ($p = 1$) nodes 1, 2, 3 and 5 are covered. With level-1 nodes 3 and 5 are covered ($d_{33} = 0 \leq 15$ and $d_{35} = 9 \leq 15$) and with level-2 nodes 1 and 2 are covered ($d_{21} = 13 \leq 20$ and $d_{22} = 0 \leq 20$).

Since nodes 1, 2 and 5 are completely covered, the objective function has value $f_1 + f_2 + f_5 = 11 + 10 + 12 = 33$.

In this paper, the integer linear programming formulation of HCLP from [5] is used. Let $a_{ij} = \begin{cases} 1, & d_{ij} \leq R_1, \\ 0, & d_{ij} > R_1, \end{cases}$ $b_{ij} = \begin{cases} 1, & d_{ij} \leq T_1, \\ 0, & d_{ij} > T_1, \end{cases}$ and $c_{ij} = \begin{cases} 1, & d_{ij} \leq R_2, \\ 0, & d_{ij} > R_2. \end{cases}$

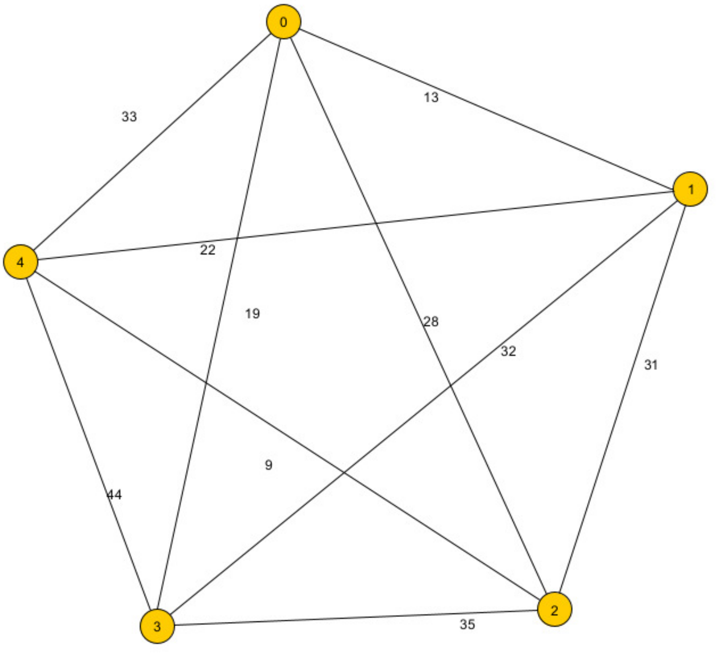

Figure 1. Distances graph in Example 1

Let decision variables be defined as follows: $x_j = \begin{cases} 1, & j \text{ is completely covered,} \\ 0, & \text{otherwise,} \end{cases}$
$y_i = \begin{cases} 1, & i \text{ is level-1 facility,} \\ 0, & \text{otherwise,} \end{cases}$ and $z_i = \begin{cases} 1, & i \text{ is level-2 facility,} \\ 0, & \text{otherwise.} \end{cases}$

Then, integer linear programming (ILP) formulation is:

$$\max \sum_{j \in J} f_j x_j \tag{1}$$

s.t.

$$\sum_{i \in I} y_i = p, \tag{2}$$

$$\sum_{i \in I} z_i = q, \tag{3}$$

$$\sum_{i \in I} a_{ij} y_i + \sum_{i \in I} b_{ij} z_i - x_j \geq 0, \quad j \in J, \tag{4}$$

$$\sum_{i \in I} c_{ij} z_i - x_j \geq 0, \quad j \in J, \tag{5}$$

$$x_j \in \{0, 1\}, \quad j \in J, \tag{6}$$

$$y_i, z_i \in \{0, 1\}, \quad i \in I. \tag{7}$$

In the presented ILP formulation, the objective function (1), to be maximized, represents the sum of weights of the completely covered nodes. Constraints (2) and (3) provide that the number of established level-1 and level-2 facilities is p and q, respectively. Constraint (4) states that for each covered node j there exists at least one established level-1 facility with a distance of at most R_1 (i.e. $a_{ij} = 1$) or at least one established level-2 facility with a distance of at most T_1 (i.e. $b_{ij} = 1$). Similarly, constraint (5) states that for each covered node j there exists at least one established level-2 facility with a distance of at most R_2 (i.e. $c_{ij} = 1$). Finally, constraints (6) and (7) represent the binary nature of the decision variables.

4 PROPOSED EM METHOD

An electromagnetism-like (EM) metaheuristic was introduced in the literature by Birbil and Fang in [15] as a powerful algorithm for global optimization that converges rapidly to the optimum. The method is also used for combinatorial optimization as a stand-alone approach or as an accompanying algorithm for other methods. It is a robust and effective metaheuristic, which can be seen from its successful applications for solving various problems. A detailed description of different EM variants is out of the scope of this paper, so only several recent and successful EM applications will be mentioned:

- Wei et al. presented an EM algorithm in order to solve cell formation problems in [16]. Based on the attraction-repulsion principle in electromagnetic theory, each combination of the part and machine groups was regarded as a charged particle. A comparison with other algorithms in the related literature found that the final result of the EM algorithm met the expected quality in terms of the exceptional elements and number of vacancies, regardless of whether or not new cells were added.
- The next successful application, described in [17] was performed for solving the response time variability problem, which has a real-life application in the automobile industry. The EM method was compared with three metaheuristic algorithms from the literature, and it was shown that, on average, the EM procedure strongly improved the previously obtained results.
- In [18], the uncapacitated hub location problem with multiple allocations was considered. An appropriate objective function, which natively conformed with the problem, 1-swap local search, and scaling technique were used to achieve good overall performance. Computational tests demonstrated the reliability

of this method, since the EM-inspired metaheuristic reached all optimal/best-known solutions, except one, in a reasonable time.

- Su and Lin in [19] utilized an electromagnetism-like mechanism combined with the 1-nearest-neighbor method as a wrapper approach to feature selection and classification in data mining. Experimental results indicated that the proposed method outperformed other well-known algorithms in not only balanced classification accuracy but also efficiency of feature selection.
- In [20], an EM approach for solving the maximum set splitting problem was presented. The hybrid approach consisted of movement based on the attraction-repulsion mechanisms combined with the proposed scaling technique directed EM to promising search regions. The fast implementation of the local search procedure additionally improved the efficiency of the overall EM system. The results showed that except in one case, EM reached optimal/best-known solutions from the literature.
- Abdullah et al. [21] proposed a hybrid metaheuristic that combines an electromagnetic-like mechanism (EM) and the great deluge algorithm (GD) for the university course timetabling problem. The proposed method was applied to a range of benchmark university course timetabling test problems from the literature, demonstrating that the method can produce improved solutions compared to those currently published.

The proposed EM algorithm for solving TLHCLP is given by the following pseudocode in Algorithm 1.

Algorithm 1: EM pseudo code

```
Algorithm EM ;
Random_Init() ;
while iteration < max_iteration do
    foreach EM point p_k in the solution_set do
        Calculate_objective_value(p_k);
        Local_search(p_k);
        Scale_solution(p_k);
    end
    Calculate_charge_and_forces();
    Move();
end
```

4.1 Objective Function

EM is a population-based algorithm that can solve nonlinear optimization problems. In the following text, each member p_k, $k = 1, 2, \ldots, N_p$ of the population maintained by the algorithm will be referred to as an EM point (or solution), and the population itself will be referred to as a solution set. Each EM point is a real vector of length $2m$, divided into m segments of two rational numbers related to decision variables z_i and y_i, respectively. Numbers related to decision variables z are sorted in non-increasing order, and potential facilities related to the q highest numbers are established as level-2 facilities. For the remaining $m - q$ non-established potential facilities, decision variables y are sorted in a non-increasing order, and potential facilities related to the p highest numbers are established as level-1 facilities. EM points in the first iteration are randomly initialized from $[0, 1]^{2m}$ (function Random_Init()). It should be noted that for all instances is $m = n$, i.e. potential facilities are actually nodes. Therefore, distance matrices are symmetric and $(\forall i), d_{ii} = 0$.

After determining which facilities are established as p level-1 and q level-2 ones, it is necessary to identify which nodes are completely covered and which ones are not. Since this step is the most computationally expensive part of the objective function evaluation, it must be implemented very efficiently. Therefore, the decision as to whether a particular node is covered by some established level-1 and level-2 facilities is divided into two parts.

The first part, which is computationally expensive, is performed only once, as preprocessing, in the initialization part of the algorithm. For each potential facility (at that moment, it is not known whether this facility will be later established or not), indices of nodes that are covered by radia R_1, T_1 and R_2, are computed and stored. Since a particular potential facility usually does not cover all nodes, this step significantly speeds up the objective function evaluations later on.

The second part is performed within the objective function evaluation. For each node, three values are noted: av, bv and cv representing the number of established facilities covering that node by R_1, T_1 and R_2 radius, respectively. All three values, av, bv and cv are initialized to zero. At this point, it is known which potential facilities are established as level-1 and level-2 ones. Therefore, for each established level-1 facility, the value of av is increased by one for all memorized indices of nodes that are covered by radius R_1. Similarly, for each established level-2 facility, the values of bv and cv are increased by one for all memorized indices of nodes that are covered by radius T_1 and R_2, respectively. After that, nodes with value $cv \geq 1$ and at least one of the two conditions $av \geq 1$ or $bv \geq 1$ are completely covered, while others are not covered.

4.2 Local Search and Scaling

This step is used to move the sample points towards the local optima that are near them. Points are pushed towards the local valleys using a neighborhood search procedure. The local search method used in this algorithm is simple but effective.

In every step, the first strategy tries a swap of one level-1 established facility with one level-2 established facility, checking if it obtains a better objective value. In order to obtain the new objective value, only coverage for the given two facilities must be recomputed. That recomputation can be made much faster than the evaluation of the objective function, since the last configuration about coverage of each node was memorized by av, bv and cv values. The second strategy tries to swap one of the level-2 facilities with one non-established facility, checking if it improves the objective function value. Similarly to the previous strategy, only the coverage for the given two facilities must be recomputed, so recomputations are made using the av, bv and cv values. The last strategy tries to swap one of the level-1 facilities with one non-established facility, also performing a recomputation using the av, bv and cv values.

The proposed local search procedure uses the first improvement strategy, which means that when an improvement is detected, the improvement is immediately applied, and the local search is performed again. For example, if some strategy is successful in swapping one facility with another, that swap is performed, and the local search is applied again with the first strategy. If all swaps in some strategy fail to produce a better objective value, the next strategy is applied. If all strategies fail, the local search procedure ends with no further improvement.

In this implementation, a scaling procedure is also applied, which additionally moves points towards solutions obtained by local search. It is considered only with some factor $\lambda \in (0, 1)$ to prevent falling into a local optimum and becoming trapped there. An EM point p_k is moved by the following formula:

$$p_k \leftarrow \lambda \cdot p'_k + (1 - \lambda) \cdot p_k, \tag{8}$$

where p'_k has ones on places of established facilities and zeros otherwise (same as in decision variables y and z), of the k^{th} EM point in the current iteration when the local search procedure finished its work.

Choosing an appropriate value of the scale factor λ is significant for governing the search process. In the extremal case, when λ is close to 1, the search process will likely fall into a local optimum and become trapped. Another extremal case, when λ is equal to 0, obviously represents no-scaling situation. Experiments showed that $\lambda = 0.1$ is a good compromise which yields satisfactory results.

4.3 Attraction-Repulsion Mechanism

As it can be seen from the literature, the strength of the EM algorithm lies in the idea of directing the sample points towards local optima utilizing an attraction-repulsion mechanism. Therefore, after applying the local search procedure to each solution in the current population, the solutions must be moved towards promising regions in order to get closer to the optimal solution.

In this process, each sample point is considered as a charged particle. The charge of each sample point is calculated by the following formula:

$$q_k = \exp\left(-n \frac{Obj^{best} - Obj_k}{\sum_{l=1}^{N_p} Obj^{best} - Obj_l}\right). \tag{9}$$

The amount of charge is related to the value of the objective function Obj_k of the point p_k after the local search has finished its work. This determines the magnitude of attraction or repulsion of the point over the sample population.

According to the superposition principle of electromagnetic theory, the force exerted on a point by another point is inversely proportional to the distance between the points and directly proportional to the product of their charges. Mathematically, the power of attraction or repulsion of charges is calculated as follows:

$$F_k = \sum_{l=1, l \neq k}^{m} F_k^l,$$

where

$$F_k^l = \begin{cases} \left(\frac{q_k q_l}{||p_k - p_l||^2}\right) \cdot (p_l - p_k), & f(p_k) < f(p_l), \\ \left(\frac{q_k q_l}{||p_k - p_l||^2}\right) \cdot (p_k - p_l), & f(p_k) \geq f(p_l), \end{cases} \tag{10}$$

where $\|p_k - p_l\|$ is the Euclidean distance between EM points p_k and p_l.

As mentioned before, the Move() procedure of the electromagnetism approach is used to shift current solutions towards the best ones. All EM points are moved except for the current best solution. Detailed explanations about the movement are given in Algorithm 2.

Algorithm 2: Move pseudo code

Function `Move`() ;
foreach *EM point* p_k *in the solution_set* **do**
    if $p_k \neq p^{best}$ **then**
        $\beta \leftarrow Random[0, 1]$;
        $F_k \leftarrow F_k / \|F_k\|$;
        for $i = 1$ *to* $2m$ **do**
            if $F_{ki} > 0$ **then**
                $p_{ki} \leftarrow p_{ki} + \beta \cdot F_{ki} \cdot (1 - p_{ki})$
            end
            else
                $p_{ki} \leftarrow p_{ki} + \beta \cdot F_{ki} \cdot p_{ki}$
            end
        end
    end
end

As shown in Algorithm 2, the movement of each EM point is in the direction of the total force exerted on it by a random step length β, which is generated from a uniform distribution between $[0, 1]$. As described in [15], candidate solutions have a non-zero probability of moving towards unvisited solutions in this direction when a random step length is selected. Additionally, normalizing the total force exerted on each candidate solution ensures that infeasible solutions cannot be produced.

5 EXPERIMENTAL RESULTS

All computations were executed on a single processor of a Quad Core 2.5 GHz PC with 4 GB RAM. The EM implementation was coded in C language. All EM runs were made with the following empirically determined parameters: $N_p = 10$, $iter_{max} = 1\,000$ and $\lambda = 0.1$. These values cause most charges to exhibit convergent behavior with a few individuals diverging, thereby providing a good balance between local and global search. In this case, all these values were chosen experimentally for convenience because they provide good results.

For the first experimental testing in this implementation, the instances described in [8] with up to 100 nodes, and $G\,\&\,R\,150$ instance with 150 nodes from [6], are used. For the first group, an experimental setup of $R_1 = 10$, $T_1 = 12$ and $R_2 = 22$ is used, while for the last instance, an experimental setup of $R_1 = 40$, $T_1 = 55$ and $R_2 = 60$, is used, the same as in the corresponding papers.

The finishing criterion for EM is the maximal number of iterations $N_{iter} = 200$, while λ has a value of 0.1 in this experiment. Because EM is not a deterministic algorithm, all experiments were executed 20 times.

Table 1 summarizes the results of EM on these instances. In the first column, the names of the instances are given. The next three columns contain the values of n, p and q, respectively. The fifth and sixth columns contain the optimal solution values and the values of the solutions obtained by EM. In the next two columns, the running time t of the first occurrence of the best EM solution and the total running time t_{tot} needed to satisfy finishing criterium are given. The next two columns contain the relative error err and the standard deviation σ in all 20 runs. The last column contains the average overall number of local search steps through all 200 iterations.

As can be seen in Table 1, EM was able to reach optimal solutions for almost all instances, except for the last two cases. The $G\,\&\,R\,150$ instance seems to be particularly challenging, as EM did not reach the optimal solutions in these two cases and required many more local search iterations. Additionally, the running time for EM on the $G\,\&\,R\,150$ instance was much larger compared to the previous instances.

A direct comparison of EM results with GA implementation from [8] is presented in Table 2. The first five columns contain instance names, n, p, q, and optimal solution of the given instance, presented in the same way as in Table 1. The next three columns contain information (best solution Sol, time t and total running time t_{tot}) of the GA algorithm from [8] executed on the same computer as the EM method. The last three columns contain the respective information about EM (Sol, t and t_{tot}).

Inst.	n	p	q	Opt	Sol	t (s)	t_{tot} (s)	err (%)	σ (%)	LS_{iter}
ex1	20	3	3	355	opt	0.002	0.135	0.000	0.000	4 847
		6	5	497	opt	0.002	0.177	0.000	0.000	4 737
		7	5	508	opt	0.005	0.177	0.217	0.651	4 582
ex2	20	3	3	441	opt	0.001	0.135	0.000	0.000	4 565
		6	3	493	opt	0.004	0.163	0.000	0.000	4 618
		3	4	508	opt	0.002	0.157	0.000	0.000	4 983
ex3	50	4	4	880	opt	0.010	1.291	0.000	0.000	5 592
		9	14	1 106	opt	0.009	1.901	0.000	0.000	4 000
ex4	100	5	6	2 297	opt	2.813	8.524	0.157	0.255	6 636
		9	12	2 567	opt	0.054	9.987	0.000	0.000	4 576
G & R 150	150	12	10	8 198	opt	42.570	146.087	0.676	0.276	18 464
		14	10	8 260	opt	37.231	160.009	1.400	0.346	16 839
		16	10	8 272	opt	50.136	175.778	1.385	0.496	18 575
		14	12	9 149	opt	53.572	175.315	0.383	0.361	19 407
		16	12	9 252	opt	50.233	190.759	1.390	0.479	18 240
		18	12	9 261	opt	62.699	205.807	1.347	0.550	19 678
		18	16	10 587	opt	80.009	236.601	0.505	0.154	21 712
		20	16	10 587	10 567	93.874	253.651	0.316	0.166	23 061
		22	16	10 587	10 571	106.986	270.767	0.318	0.167	24 185

Table 1. EM results on instances from [8]

Inst.	n	p	q	Opt	*GA*			*EM*		
					Sol	t (s)	t_{tot} (s)	Sol	t (s)	t_{tot} (s)
ex1	20	3	3	355	opt	0.003	0.277	opt	0.002	0.135
		6	5	497	opt	0.046	0.364	opt	0.002	0.177
		7	5	508	opt	0.060	0.376	opt	0.005	0.177
ex2	20	3	3	441	opt	0.007	0.289	opt	0.001	0.135
		6	3	493	opt	0.089	0.452	opt	0.004	0.163
		3	4	508	opt	0.005	0.301	opt	0.002	0.157
ex3	50	4	4	880	opt	0.199	1.283	opt	0.010	1.291
		9	14	1 106	opt	0.042	35.037	opt	0.009	1.901
ex4	100	5	6	2 297	opt	1.862	6.492	opt	2.813	8.524
		9	12	2 567	opt	0.546	291.033	opt	0.054	9.987

Table 2. Comparison of results on instances from [8]

As shown in Table 2, both GA and EM were able to reach all optimal solutions, and the running time on all instances was relatively small. Although EM performed slightly faster than GA in most cases, definitive conclusions about superiority cannot be drawn from the presented data in Table 2.

In order to obtain these answers, as well as a comparison with the results of the Lagrangean surrogate heuristic (LSH) from [6], Table 3 is presented. Note that experimental results of LSH, tested on a less powerful computer, are assumed as in [6], tested on less powerful computer Pentium II with 300 MHz processor. The data in Table 3 is presented in a similar way as in Table 2, with additional 3 columns for the results of LSH. Additionally, instance name and dimension are omitted to obtain a compact presentation.

p	q	Opt	*LSH*		*GA*			*EM*		
			Sol	t (s)	Sol	t (s)	t_{tot} (s)	Sol	t (s)	t_{tot} (s)
12	10	8 198	8 134	5.05	8 095	175.856	995.1	opt	42.570	146.0
14	10	8 260	8 071	4.67	8 095	230.109	1 114.3	opt	37.231	160.0
16	10	8 272	8 233	7.25	8 090	229.186	1 172.8	opt	50.136	175.7
14	12	9 149	9 057	3.68	9 087	24.833	915.7	opt	53.572	175.3
16	12	9 252	9 087	5.88	9 114	84.453	1 096.6	opt	50.233	190.7
18	12	9 261	9 087	7.58	9 113	29.871	1 107.4	opt	62.699	205.8
18	16	10 587	10 434	6.26	10 501	298.131	1 481.1	opt	80.009	236.6
20	16	10 587	10 459	5.65	10 571	212.101	1 463.4	10 567	93.874	253.6
22	16	10 587	10 340	4.39	10 495	121.593	1 416.0	10 571	106.986	270.7

Table 3. Comparison of results on $G\,\&\,R\,150$ instance from [6]

Inst.	n	p	q	Sol	t (s)	t_{tot} (s)	*err* (%)	σ (%)	LS_{iter}
ex5	200	3	3	2 645	0.231	18.880	0.000	0.000	15 223
		5	5	3 914	13.766	35.906	0.010	0.020	16 355
		7	8	4 910	24.415	61.288	0.574	0.436	17 819
		10	10	5 097	6.796	74.790	0.000	0.000	17 565
ex6	300	3	3	3 634	0.357	41.386	0.000	0.000	15 286
		5	5	5 422	10.118	82.257	0.015	0.044	16 529
		7	8	6 888	55.011	138.984	0.468	0.195	17 929
		10	10	7 466	76.337	196.165	0.467	0.198	18 868
ex7	400	3	3	4 503	3.121	77.304	0.000	0.000	15 418
		5	5	6 875	51.956	154.271	0.150	0.310	16 591
		7	8	8 955	123.841	261.616	0.997	0.425	17 974
		10	10	9 834	176.424	369.172	0.566	0.294	18 917
ex8	500	3	3	5 494	0.871	123.707	0.000	0.000	15 462
		5	5	8 392	90.384	250.935	0.066	0.133	16 672
		7	8	10 997	176.875	416.600	0.687	0.351	17 882
		10	10	12 173	224.168	585.856	0.886	0.265	18 823

Table 4. EM results on larger instances with $R_1 = 10$, $T_1 = 12$ and $R_2 = 22$

As can be seen from Table 3, the EM method produces significantly better results than the GA approach, both in terms of solution quality and running times. The EM method also produces better solutions compared to the LSH approach, but with much longer running times.

For tables, n is the dimension of the problem and the complexity of the problem depends on increasing the order of n. In such a representation, it is clearly visible what the dimension of the instance is. (In Table 3, there is only one instance, so it is clear which dimension it is about.)

Inst.	n	p	q	Sol	t (s)	t_{tot} (s)	err (%)	σ (%)	LS_{iter}
ex5	200	3	3	3 743	2.726	20.203	0.000	0.000	15 331
		5	5	4 875	14.721	34.861	0.287	0.285	16 053
		7	8	5 097	0.233	41.830	0.000	0.000	15 129
		10	10	5 097	0.222	41.049	0.000	0.000	14 042
ex6	300	3	3	5 342	0.886	45.016	0.000	0.000	15 523
		5	5	6 988	32.493	77.255	0.275	0.193	16 192
		7	8	7 550	9.278	101.961	0.000	0.000	15 820
		10	10	7 550	0.544	97.690	0.000	0.000	14 450
ex7	400	3	3	6 861	1.258	81.085	0.000	0.000	15 409
		5	5	9 197	53.632	142.409	0.723	0.385	16 151
		7	8	10 153	88.383	207.337	0.252	0.132	16 397
		10	10	10 153	1.065	205.403	0.000	0.000	15 084
ex8	500	3	3	8 238	22.821	129.900	0.000	0.000	15 471
		5	5	11 278	120.490	234.111	0.578	0.444	16 348
		7	8	12 593	180.375	342.549	0.293	0.140	16 663
		10	10	12 654	1.823	333.728	0.000	0.000	15 236

Table 5. EM results on larger instances with $R_1 = 15$, $T_1 = 20$ and $R_2 = 30$

In order to observe the behavior of the EM method on large-scale instances that are out of reach for exact methods, instances of dimensions $n = 200$, 300, 400 and 500 were generated using the instance generator from [8]. Since optimal solutions for these instances are not known, except for the two smallest cases, the results of the EM method are compared to the GA method from [8]. Two experimental setups are used:

1. Table 4 contains computational results based on the values $R_1 = 10$, $T_1 = 12$ and $R_2 = 22$, which are the same values used in [8];
2. Setup values $R_1 = 15$, $T_1 = 20$ and $R_2 = 30$ are used in the second experiment, and the results are presented in Table 5.

Note that for higher values of p and q ($p > 10$, $q > 10$) both setups give similar results, since all (or almost all) demand areas are covered, so these cases are not reported in Tables 4 and 5. The data in Tables 4 and 5 are presented in the same way as in Table 1, except that in almost all cases optimal solutions are not known, so that column is omitted. Only two optimal solutions are known (with setup $R_1 = 10$, $T_1 = 12$ and $R_2 = 22$):

1. ex5 with $p = 3, q = 3$ with optimal solution value 2 645 obtained with CPLEX solver 10.1 after 4 minutes;
2. ex6 with $p = 3, q = 3$ with optimal solution value 3 634 obtained with CPLEX solver 10.1 after 55 minutes;
3. ex7 with $p = 3, q = 3$ with optimal solution value 4 503 obtained with CPLEX solver 10.1 after 8 hours and 17 minutes.

Inst.	n	p	q	*GA*			*EM*		
				Sol	t (s)	t_{tot} (s)	Sol	t (s)	t_{tot} (s)
ex5	200	3	3	2 645	4.074	183.9	2 645	0.231	18.9
		5	5	3 914	14.336	730.1	3 914	13.766	35.9
		7	8	4 898	27.898	1 449.8	***4 910***	24.415	61.3
		10	10	5 097	71.92	1 004.6	5 097	6.796	74.8
ex6	300	3	3	3 634	19.87	320.7	3 634	0.357	41.4
		5	5	5 422	45.011	804.4	5 422	10.118	82.3
		7	8	6 760	79.966	1 507.8	***6 888***	55.011	139.0
		10	10	7 410	75.814	989.9	***7 466***	76.337	196.2
ex7	400	3	3	4 503	43.31	61.4	4 503	3.121	77.3
		5	5	6 820	146.928	994.7	***6 875***	51.956	154.3
		7	8	8 743	166.075	825.8	***8 955***	123.841	261.6
		10	10	9 669	242.499	971.4	***9 834***	176.424	369.1
ex8	500	3	3	5 494	122.971	482.6	5 494	0.871	123.7
		5	5	8 314	253.786	918.1	***8 392***	90.384	250.9
		7	8	10 859	505.779	2 059.9	***10 997***	176.875	416.6
		10	10	12 053	401.739	732.6	***12 173***	224.168	585.9

Table 6. Comparison of results on larger instances with $R_1 = 10$, $T_1 = 12$ and $R_2 = 22$

For all other instances, as well as for all instances with setup $R_1 = 15$, $T_1 = 20$ and $R_2 = 30$, CPLEX cannot finish its work after one day of running time.

Direct comparison of the EM results with the GA implementation from [8] is given in Tables 6 and 7, presented in the same way as Table 2, except that in almost all cases, the column for optimal solutions is omitted since they are not known. In cases where one method obtains a strictly better result than the others, the result is noted in italic boldface.

Direct comparison with GA on large instances also shows the superiority of the present EM approach in both solutions – quality and running times, in most cases.

From Tables 1, 2, 3, 4 and 5 it can be seen that EM provides solutions of a certain quality. In Table 4, maximal total running time $t_{tot} = 585.856$ seconds which is approximately 10 minutes (achievable time). It is shown that EM, for large values n, is capable of attaining solutions of good quality and EM is competitive compared with GA. Data for ex4, from Table 2, shows that when the value of p and q increases, the execution time increases significantly, i.e. for $p = 5$ and $q = 6$, $t_{tot} = 6.492$ seconds with GA, while $t_{tot} = 8.524$ seconds with EM. Too, for $p = 9$ and $q = 12$, $t_{tot} = 261.033$ seconds with GA, while $t_{tot} = 9.987$ seconds with EM.

The data LSH and GA from Table 3 did not reached any optimal solution, while EM reached optimal solutions in all cases except two ($p = 20$ and $q = 16$, $p = 22$ and $q = 16$). Data for ex8 where is $p = 10$ and $q = 10$, from Table 6, shows that solution value obtained with GA is 12 053, while solution value obtained with EM is 12 173. It can be seen that EM provides better solution and that the execution time is less. Data for ex8 where is $p = 10$ and $q = 10$, from Table 7, shows that $t_{tot} = 43\,448.2$ seconds with GA, while $t_{tot} = 333.7$ seconds with EM, but the solution values 12 654

Inst.	n	p	q	*GA*			*EM*		
				Sol	t (s)	t_{tot} (s)	Sol	t (s)	t_{tot} (s)
ex5	200	3	3	3 743	10.194	29.4	3 743	2.726	20.2
		5	5	4 838	17.517	39.4	*4 875*	14.721	34.7
		7	8	5 097	42.529	1 735.3	5 097	0.233	41.8
		10	10	5 097	2.954	2 695.3	5 097	0.222	41.0
ex6	300	3	3	5 342	29.489	75.1	5 342	0.886	45.0
		5	5	6 944	31.285	84.3	*6 988*	32.493	77.3
		7	8	7 550	235.737	903.4	7 550	9.278	102.0
		10	10	7 550	20.337	9 564.3	7 550	0.544	97.7
ex7	400	3	3	6 861	62.856	172.9	6 861	1.258	81.1
		5	5	9 066	149.052	275.8	*9 197*	53.632	142.4
		7	8	10 116	185.899	326.9	*10 153*	88.383	207.3
		10	10	10 153	125.889	28 581.6	10 153	1.065	205.4
ex8	500	3	3	8 210	151.824	338.3	*8 238*	22.821	129.9
		5	5	11 125	237.922	454.1	*11 278*	120.490	234.1
		7	8	12 516	534.011	741.6	*12 593*	180.375	342.5
		10	10	12 654	129.595	43 448.2	12 654	1.823	333.7

Table 7. Comparison of results on larger instances with $R_1 = 15$, $T_1 = 20$ and $R_2 = 30$

are the same. GA found a solution for $t = 129.595$ seconds, while EM found the solution for 1.823 seconds. Too, data for ex8 where is $p = 7$ and $q = 8$, shows that $t_{tot} = 741.6$ seconds with GA, while $t_{tot} = 342.5$ seconds with EM, but GA found a solution for $t = 534.011$ seconds, while EM found the solution for 180.375 seconds and EM obtained a better solution than GA.

6 CONCLUSIONS

This paper presents a robust and effective metaheuristic called electromagnetism-like method for solving the two-level hierarchical covering location problem. The proposed approach utilizes an efficient objective function, local search procedure, attraction-repulsion mechanisms, and scaling techniques resulting in excellent experimental results. This approach and a completely new method for solving the discussed problem were both applied for the first time. The electromagnetism-like metaheuristic (EM) used to solve the two-level hierarchical covering location problem (TLHCLP) in the paper has several novel aspects.

One aspects is adaptation of EM metaheuristic: The paper adapts the EM metaheuristic, originally designed for continuous optimization problems, to the discrete combinatorial problem of TLHCLP. This adaptation involves designing new components for the EM algorithm, such as a discrete representation of solutions, a discrete attraction-repulsion function, and discrete update rules. Next aspect is integration of two-level hierarchical structure: The TLHCLP is a complex optimization problem with a two-level hierarchical structure, where facilities at the upper level serve customers at the lower level. The paper proposes a novel representation of solutions that captures this hierarchical structure and allows for efficient search of the solution space. And, one more novel aspect is performance evaluation: The paper evaluates the performance of the proposed EM metaheuristic on a set of benchmark instances of the TLHCLP and compares it with other state-of-the-art metaheuristics. The results show that the EM algorithm outperforms other metaheuristics in terms of solution quality and computational efficiency. Overall, the paper presents a novel and effective approach for solving the challenging problem of TLHCLP using an adapted version of the EM metaheuristic.

Direct comparison with a genetic algorithm from the literature indicates the superiority of the EM approach in terms of solution quality, with significantly smaller running times. Based on the presented results, the EM method is deemed as a valuable metaheuristic for solving the proposed problem.

For future work, it would be interesting to apply the EM to similar problems and explore hybridization with other metaheuristics or exact methods. Additionally, it could be beneficial to investigate the performance of the EM approach on real-world instances and compare it with the existing solutions. Moreover, exploring the potential of incorporating parallel computing techniques or adaptive parameters could further enhance the efficiency and effectiveness of the proposed method. Overall, the presented electromagnetism-like method provides a promising avenue for solving complex hierarchical covering location problems, with potential applications in various domains, including transportation, logistics, and facility location.

REFERENCES

[1] Alizadeh, B.—Burkard, R. E.: A Linear Time Algorithm for Inverse Obnoxious Center Location Problems on Networks. Central European Journal of Operations Research, Vol. 21, 2013, No. 3, pp. 585–594, doi: 10.1007/s10100-012-0248-5.

[2] Kovačić, D.—Bogataj, M.: Reverse Logistics Facility Location Using Cyclical Model of Extended MRP Theory. Central European Journal of Operations Research, Vol. 21, 2013, No. Suppl 1, pp. S41–S57, doi: 10.1007/s10100-012-0251-x.

[3] Farahani, R. Z.—Asgari, N.—Heidari, N.—Hosseininia, M.—Goh, M.: Covering Problems in Facility Location: A Review. Computers and Industrial Engineering, Vol. 62, 2012, No. 1, pp. 368–407, doi: 10.1016/j.cie.2011.08.020.

[4] Galvão, R. D.—ReVelle, C.: A Lagrangean Heuristic for the Maximal Covering Location Problem. European Journal of Operational Research, Vol. 88, 1996, No. 1, pp. 114–123, doi: 10.1016/0377-2217(94)00159-6.

[5] Moore, G. C.—ReVelle, C.: The Hierarchical Service Location Problem. Management Science, Vol. 28, 1982, No. 7, pp. 775–780, doi: 10.1287/mnsc.28.7.775.

[6] Espejo, L. G. A.—Galvao, R. D.—Boffey, B.: Dual-Based Heuristics for a Hierarchical Covering Location Problem. Computers and Operations Research, Vol. 30, 2003, No. 2, pp. 165–180, doi: 10.1016/S0305-0548(01)00087-9.

[7] Maric, M.—Tuba, M.—Kratica, J.: One Genetic Algorithm for Hierarchical Covering Location Problem. Proceedings of the 9th WSEAS International Conference on Evolutionary Computing (EC'08), 2008, pp. 122–126.

[8] Maric, M.—Tuba, M.—Kratica, J.: Parameter Adjustment for Genetic Algorithm for Two-Level Hierarchical Covering Location Problem. WSEAS Transactions on Computers, Vol. 7, 2008, No. 6, pp. 746–755.

[9] Lee, J. M.—Lee, Y. H.: Tabu Based Heuristics for the Generalized Hierarchical Covering Location Problem. Computers and Industrial Engineering, Vol. 58, 2010, No. 4, pp. 638–645, doi: 10.1016/j.cie.2010.01.005.

[10] Töreyen, Ö.: Hierarchical Maximal Covering Location Problem with Referral in the Presence of Partial Coverage. Master Thesis. Middle East Technical University, 2007.

[11] Serra, D.: The Coherent Covering Location Problem. Papers in Regional Science, Vol. 75, 1996, No. 1, pp. 79–101, doi: 10.1007/BF02406961.

[12] Overholts II, D. L.—Bell, J. E.—Arostegui, M. A.: A Location Analysis Approach for Military Maintenance Scheduling with Geographically Dispersed Service Areas. Omega, Vol. 37, 2009, No. 4, pp. 838–852, doi: 10.1016/j.omega.2008.05.003.

[13] Casas-Ramírez, M. S.—Camacho-Vallejo, J. F.—Díaz, J. A.—Luna, D. E.: A Bi-Level Maximal Covering Location Problem. Operational Research, Vol. 20, 2020, No. 2, pp. 827–855, doi: 10.1007/s12351-017-0357-y.

[14] Şahin, G.—Süral, H.: A Review of Hierarchical Facility Location Models. Computers and Operations Research, Vol. 34, 2007, No. 8, pp. 2310–2331, doi: 10.1016/j.cor.2005.09.005.

[15] Birbil, Ş. İ.—Fang, S. C.: An Electromagnetism-Like Mechanism for Global Optimization. Journal of Global Optimization, Vol. 25, 2003, No. 3, pp. 263–282, doi: 10.1023/A:1022452626305.

[16] Wei, N. C.—Lin, H. K.—Wu, P.: An Electromagnetism-Like Mechanism for Solving Cell Formation Problems. Scientific Research and Essays, Vol. 7, 2012, No. 9, pp. 1022–1034.

[17] García-Villoria, A.—Moreno, R. P.: Solving the Response Time Variability Problem by Means of the Electromagnetism-Like Mechanism. International Journal of Production Research, Vol. 48, 2010, No. 22, pp. 6701–6714, doi: 10.1080/00207540902862545.

[18] Filipović, V.: An Electromagnetism Metaheuristic for the Uncapacitated Multiple Allocation Hub Location Problem. Serdica Journal of Computing, Vol. 5, 2011, No. 3, pp. 261–272, doi: 10.55630/sjc.2011.5.261-272.

[19] Su, C. T.—Lin, H. C.: Applying Electromagnetism-Like Mechanism for Feature Selection. Information Sciences, Vol. 181, 2011, No. 5, pp. 972–986, doi: 10.1016/j.ins.2010.11.008.

[20] Kratica, J.: An Electromagnetism-Like Method for the Maximum Set Splitting Problem. Yugoslav Journal of Operations Research, Vol. 23, 2016, No. 1.

[21] Abdullah, S.—Turabieh, H.—McCollum, B.—McMullan, P.: A Hybrid Metaheuristic Approach to the University Course Timetabling Problem. Journal of Heuristics, Vol. 18, 2012, No. 1, pp. 1–23, doi: 10.1007/s10732-010-9154-y.

Milica Milivojević Danas received her Bachelor with Honours in mathematics (2010), Master in mathematics (2012) and Ph.D. in mathematics (2022) from the University of Kragujevac, Faculty of Science and Mathematics. In 2013, she joined the Faculty of Science and Mathematics as a researcher, currently she works at the Faculty of Engineering. Her research interests include extremal graph theory, discrete mathematics and mixed metric dimension.

Milena Bogdanović is a Ph.D. Doctor of computer science, 2010 Faculty of Mathematics, Belgrade, Serbia; Master of Mathematical Sciences, 2001 Faculty of Science and Mathematics, Niš, Serbia; Graduate mathematician of Computing and Informatics, 1994 Faculty of Science and Mathematics, Belgrade, Serbia. She is the author of seven books and of about 80 published scientific papers in the field of the mathematics and computer science. Her professional papers discuss problems in the field of applications of multimedia in teaching, combinatorial optimization, genetic algorithms, directable automata, graphs invariant.