Available online at http://ijim.srbiau.ac.ir/
Int. J. Industrial Mathematics (ISSN 2008-5621)
Vol. 13, No. 3, 2021 Article ID IJIM-1058, 14 pages
DOR: http://dorl.net/dor/20.1001.1.20085621.2021.13.4.8.7
Research Article

Science and Research Branch (IAU)

Hybrid Genetic for the Single-Source Capacitated Multi-Facility Weber Problem

S. Jahadi *†, M. Solimanpur ‡

Received Date: 2018-04-25 Revised Date: 2020-04-03 Accepted Date: 2021-06-07

Abstract

In this paper, we investigate the Single-Source Capacitated Multi-Facility Weber Problem. The aim is to locate several new facilities among existing customers and simultaneously allocate customers to facilities, so that capacity constraints are met and each customer will provide all its demand from just one facility, and as a result, the total cost of transporting between new facilities and existing customers gets minimum. A Genetic Algorithm is proposed for solving the problem, in which a local search method is embedded. The proposed Genetic Algorithm is tested on existing data sets to evaluate its robustness over available methods in the literature.

Keywords : Weber Problem; Location-allocation, Metaheuristic, Local Search, Genetic Algorithm.

1 Introduction

Location problems are considered to be crucial problems in decision making by governments, organizations and companies. The simplest form of location problems is the single-facility location problem, which is known as the Weber Problem. An extension of the Weber Problem is the Multi-Source Weber Problem, which is known as the location-allocation problem. The objective of the location-allocation problem is to locate some new facilities among several customers with determined locations and fixed demands, and simultaneously allocate customer demands to facilities, so that the total cost of transportation between facilities and existing customers gets minimum. In a division, we can divide the location-allocation problem into three continuous, discrete, and on-network categories. In another division, it is divided to uncapacitated problem or capacitated problem. In this study another type of the problem known as the Single-Source Capacitated Multi-Facility Weber Problem (SSCMFWP) is studied in which each customer must provide all of its demand only from one facility. In addition, taking into account the capacity constraints, one can study situations that are much closer to reality. In practice, there are certain situations where each customer must provide its demand only from one facility. In addition, working just with one facility is sometimes more attractive from a customer viewpoint because the

*Corresponding author. saed.jahadi@gmail.com, Tel:+98(914)1907167.
†Department of Industrial Engineering, Urmia University of Technology, Urmia, Iran.
‡Department of Industrial Engineering, Urmia University, Urmia, Iran.

customer can get a discount due to the economies of scale. This applies to locating factories, warehouses, public facilities [8], as well as industrial supplies, transportation planning, irrigation and oil pit location systems [36], hospitals, schools, industries, and the like. The optimal solution of the Multi-Source Weber Problem might be unusable in practice, as several facilities may be located in some forbidden area such as a lake, a mountain, etc. [37]. In general, single-facility location problems are convex, and their optimal solution can be obtained either through exact methods or through some heuristic algorithms [13], while multi-facility location problems are non-convex and non-linear, and exact algorithms cannot solve large-scale problems optimally in a polynomial time. In addition, Cooper [12] has proven that the objective function of these problems is neither concave nor convex, and may include several local optima. Hence, the Multi-Source Weber Problem has been located in the field of general optimization problems [20]. In addition, even if all customers are on a straight line, the problem is NP-Hard [40, 37]. In the concept of the Multi-Source Weber Problem, the large-scale problem points to the existence of high local optima, which makes it difficult if it is not impossible, to find the optimal solution at a reasonable computational time. In addition, the complexity of the problem depends on a nonlinear behavior on the number of facilities and the number of customers [8]. If in any location-allocation problem the allocation of customers to the facility is assumed to be fixed, it will be reduced to a location problem and Similarly, if facility locations are determined, it is reduced to an allocation problem.

2 Literature review

2.1 Multi-Source Weber Problem

The literature of the location-allocation problem focuses on solution methods, since this has a lot of local optima [20]. For the first time, Cooper [12] in 1963 studied the location-allocation problem and proposed an exact algorithm, as well as a method for testing all possible allocations. Cooper [13] proposed the first heuristic algorithm for solving the continuous location allocation problem, which is known as the Alternate Location Allocation (ALA) algorithm, and is an interesting algorithm in the literature. Teitz and Bart [47] studied the location-allocation problem on network and presented a heuristic algorithm, swap of vertexes, in order to find an absolute median. Eilon et al. [17], for a problem with 5 facility and 50 customer, according to 200 random initial solution, found 61 local optima, that the worst of them had 40.9 percent difference with the best of them. Kuenne and Soland [27] presented an approximate algorithm and a branch and bound algorithm to solve the problem. Love and Juel [30], studied the location-allocation problem and presented five methods for solving the problem. Moreno et al. [39] presented a drop heuristic algorithm that starts with an initial solution of N customer, in which N is selected between m (number of facilities) and 2m. Then the extra facilities were removed by a greedy method up to the point where exactly m facility remained. Love and Morris [31] presented the Set Reduction method and the P-median Algorithm to solve the location-allocation problem with Rectilinear distances. Liu et al. [29] used a Simulated Annealing algorithm with Rectilinear distances to solve the problem. Bongartz et al. [5] presented the Projection method for solving the Multi-Source Weber Problem with Lp norm distances. Brimberg and Mladenovic [10] used the Tabu Search algorithm to solve the problem. Brimberg and Mladenovic [11] presented a Neighborhood Search method. Hansen et al. [22], by considering the location of customers as potential locations for the establishment of facilities, defined the P-median problem corresponding to the main problem and solved it exactly. Brimberg et al. [8], compared different algorithms for solving the Multi-Source Weber Problem. Gamal and Salhi [19] presented a constructive heuristic algorithm. Gamal and Salhi [18] presented a Cellular heuristic algorithm. Salhi and Gamal [46] also presented a Genetic Algorithm for solving the continuous location-allocation problem without capacity constraints. Resende and Werneck [44] introduced an application of Teitz and Barts swap based local search [47] and Resende and Werneck [45] presented another application for the P-median problem, which is faster than the previous one. Jabalameli and Ghaderi [26] presented three hybrid algorithms for solving the continuous location-allocation problem without capacity constraints. Neema et al. [41] proposed two Genetic Algorithms to solve the continuous location-allocation problem. Ghaderi and Jabalameli [20] presented two classic and hybrid PSO methods. Brimberg and Drezner [6] presented a modified version of the ALA algorithm to solve the continuous problem. Brimberg et al. [7] presented an innovative random method to produce a good initial solution to the continuous location-allocation problem. For further study on the uncapacitated location-allocation problems, readers can refer to review articles such as Mladenovic and Brimberg et al. [38] for the p-Median and Brimberg et al. [9] for the continuous problem. Arnaout [4] considered the location-allocation problem with unknown number of facilities and introduced an Ant Colony Optimization (ACO) Algorithm. Drezner et al. [15] proposed new heuristic algorithms for solving the p-median problem. Also Drezner et al. [16] peresented new local searches for solving the Multi-Source Weber Problem. Dehkordi [42] proposed a modified version of Coopers ALA algorithm for solving the MSWP.

2.2 Capacitated Multi-Source Weber Problem

There are very few articles that have studied the problem with capacity constraints. First time, Cooper [14] studied this problem in 1972 and presented exact and heuristic algorithms to solve it. Aras et al. [3] proposed three heuristic algorithms to solve the problem with Euclidean and Squared Euclidean distances. Zainuddin and Salhi [49] presented a Perturbation based heuristic algorithm. Luis et al. [33] presented a heuristic algorithm called Region-Rejection Based Algorithm (RRA) for Multi-Source Weber Problem with capacity constraints. Luis et al. [34] presented a random search algorithm called Guided Reactive GRASP. Luis et al. [35] introduced fixed costs of opening facilities for the Capacitated Multi-Source Weber Problem and proposed a Greedy Randomized Adaptive Search Procedure (GRASP) for solving the problem. Akyuz et al. [1] proposed location and allocation based branch and bound algorithms for the Capacitated Multi-Source Weber Problem. Hosseininezhad et al. [23] considered fixed cost for the Capacitated Multi-Source Weber Problem and proposed a Cross Entropy heuristic for solving the problem. Luis et al. [32] considered various capacity constraints to the Multi-Source Weber Problem and proposed a Greedy Randomised Adaptive Search Procedure (GRASP) to deal with the problem. Lara et al. [28] considered the Capacitated Multi-Source Weber Problem with fixed cost and proposed a Decomposition Algorithm. Akyuz et al. [2] proposed two branch and bound algorithms for exactly solving multi-commodity extension of the Capacitated Multi-Source Weber Problem.

2.3 Single-Source Capacitated Multi-Facility Weber Problem

The number of studies considering the SSCMFWP is very rare. Gong et al. [21] presented a hybrid evolutionary method. Manzour-al-Ajdad et al. [37] presented an iterative two-phase heuristic algorithm. Manzour-al-Ajdad et al. [36] presented two modified versions of the Cooper's ALA algorithm to solve the problem. Temel Oncan [40] studied the problem with Euclidean and Rectilinear distances, and presented several methods to solve the problem. Irawan et al. [24] investigated the single-source location problems with the presence of several possible capacities and fixed cost. By considering both the discrete and the continuous cases using Rectilinear and Euclidean distances, they proposed two solution methods. Irawan et al. [25] investigated the SSCCMFWP with setup cost of opening facilities and introduced a nonlinear mathematical model. They used Rectilinear and Euclidean distances and proposed two metaheuristic algorithms based on Variable Neighbourhood Search and Simulated Annealing. They also constructed a new data set.

3 Mathematical formulation

The goal of the Single-Source Capacitated Multi-Facility Weber Problem (SSCMFWP) is to find optimal location of facilities among existing customers and simultaneously determine how customers are allocated to facilities, so that each customer must provide all its demand just from one facility, and the total cost of transporting between customers and facilities gets minimum. The following notations are used.

Parameters:

m: the number of facilities.
n: the number of customers.
w_j: the demand of customer j.
c_i: the capacity of facility i.
$A_j = (a_j, b_j)$: coordinates of customer j.

Decision variables:

$X_i = (x_i, y_i)$: coordinates of facility i.

$$z_{ij} = \begin{cases} 1; \text{if customer j is assigned to facility i,} \\ 0; \text{otherwise} \end{cases}$$

Let $d(X_i, A_j)$ be the distance between facility i and customer j which is defined as follows:

$$d(X_i, A_j) = \sqrt{(x_i - a_j)^2 + (y_i - b_j)^2}$$

The mathematical model of the SSCMFWP can be formulated as follows:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ij} \times d(X_i, A_j) \tag{3.1}$$

subject to

$$\sum_{i=1}^{m} z_{ij} = 1, \ \forall j = 1, 2, \ldots, n \tag{3.2}$$

$$\sum_{j=1}^{n} w_j \times z_{ij} \leq c_i, \ \forall i = 1, 2, \ldots, m \tag{3.3}$$

$$z_{ij} \in \{0, 1\}, \ i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, n \tag{3.4}$$

Eq. (3.1) minimizes the total transportation cost between the facilities and the customers. Eq. (3.2) states that each customer must provide all its demand only from one facility. Eq. (3.3) ensures that the total demand, which is provided by each facility should not exceed its capacity limit. Eq. (3.4) indicates that z_{ij} is a binary variable.

It is worth noting that, for a given set of determined allocations, the SSCMFWP is reduced into m single-facility location problems, each of which can be separately solved exactly by the well-known Weiszfeld's method [48]. It is reminded that, the proposed model is a non-convex mixed integer non-linear programming (MINLP) model. It is also noted that the SSCMFWP is much more complicated than the CMFWP. In fact, after fixing the location of facilities, the CMFWP reduces to the Transportation Problem (TP), while the SSCMFWP after fixing facility locations reduces to the Generalized Assignment Problem (GAP) [37].

4 The proposed Genetic Algorithm

The Genetic Algorithm is a random search algorithm which acts based on the rules of natural evolution. A Genetic Algorithm is a set of initial solutions that each of them is called a chromosome in the population, which are produced during each generation, according to genetic crossover and mutation operators. Now, some of the best chromosomes are selected and some are selected randomly for the next generation. The process is repeated until stopping condition is satisfied. Unlike local search algorithms that usually get stuck in local optima, the Genetic Algorithm is able to search all of the solution space and find the optimal or near optimal solution. We will present a Hybrid Genetic Algorithm (HGA) in which a local search is embedded to solve the SSCMFWP. The main steps of the proposed Genetic Algorithm are as follows:

Require: K_1, K_2, I_{max}, P
1: **repeat**
2:    Produce an initial solution.
3:    Improve the produced initial solution by applying the local search approach.
4: **until** P times
5: **repeat**
6:    **for** $O = 1$ to P **do**
7:       Produce a new solution as a child of the solution O using the proposed mutation approach.

Improve the produced child by applying the proposed local search approach.
end for
Select among all parents and childeren, K_1 number of the best and K_2 number randomly and create the new population $(k_1 + k_2 = P)$ for the next iteration.
until I_{max} times
return the best solution

4.1 Encoding

Each chromosome in the population is made up of a number of genes with a particular sequence. Each gene can be represented by binary numbers, real numbers, integers, symbols and the like. Generally, in the Genetic Algorithm for representing the chromosomes the binary encoding is used. In practice, the use of binary encoding is not always appropriate. In addition, it is possible to produce infeasible solutions when using genetic operators. In this study, we use the real number encoding in which each chromosome is represented as follows:

$$[(x_{k1}, y_{k1}), (x_{k2}, y_{k2}), \ldots, (x_{km}, y_{km})] \to f_k$$

In which f_k represents the value of the objective function of the chromosome k and (x_{km}, y_{km}) is the coordinates of the facility m in the solution k. It represents the mth gene in the chromosome k.

4.2 Initial solution

Typically, randomly generated location of facilities are used to start the search process in solving location problems. But this method is not very suitable because facilities may be located in the same area of the solution space and are chosen very close together and the rest of the areas be vacant, which will be an inappropriate initial solution. Hence, we produce every initial solution in the primary population as follows:

Require: m, n
Denote:
T_i: the set of allocated customers to facility i, $i = 1, 2, \ldots, m$,
$q_i = |T_i|$: the amount of customer demands that are met by facility i, $i = 1, 2, \ldots, m$.
Set $z_{ij} = 0$, $q_i = 0$ $\quad for \quad i = 1, 2, \ldots, m, j = 1, 2, , n$.
Select customer j randomly among all customers.
Locate facility 1 on the location of the customer j and set $X_1 = A_j$.
repeat
Let $j\prime$ be index of the closest customer to the facility 1.
Allocate the customer $j\prime$ to the facility 1 and set $z_{1j}\prime = 1$, $q_1 + = w_{j\prime}$.
until $q_1 < c_1$
for $i = 2\ to\ m$ **do**
Set $D_{min} = \infty$.
for $j = 1\ to\ n$ **do**
if the customer j is an unallocated customer **then**
Let D_j be cumulative distance between the jth customer and all already located facilities and calculate $D_j = \sum_{k=1}^{i} d(X_k, A_j)$
if $D_j \leq D_{min}$ **then**
Let $j^\star$ be index of the customer with minimum D_j and set $j \to j^\star$.
end if
end if
end for
Locate the facility i on the location of the customer $j^\star$ and set $X_i = A_{j^\star}$.
repeat
let $j\prime$ be the index of nearest unallocated customer to the facility i.
Assign the customer $j\prime$ to the facility i and set $z_{ij\prime} = 1$ and $q_i + = w_{j\prime}$.
until $q_i < c_i$.
end for
Calculate the total cost as: $f = \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ij} \cdot w_j \cdot d(X_i, A_j)$.
return $f,\ X_i,\ q_i,\ z_{ij}\ for\ i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.

4.3 The proposed local search approach

Local search algorithms start with a point in the solution space and try to move to a point in the neighborhood, which is more suitable than the initial starting point. The search ends when a more appropriate solution in the neighborhood

Table 1: Computational results of the test problem with 50 customers.

	Previous methods								**Presented method**	
	SSALA		**SSALA-VLSN**		**TPH-S**		**TPH-P**		**HGA**	
m	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**
2	135.80	1.2	135.80	0.1	135.80	0.7	135.80	0.3	**135.79**	0.15
3	106.35	2.1	106.74	0.1	106.36	0.6	106.69	0.5	**106.33**	0.45
4	86.49	2.9	87.60	0.2	86.49	0.9	86.49	0.8	**86.48**	0.46
5	74.46	3.7	75.02	0.2	79.73	1.0	74.12	1.2	**74.10**	0.4
6	61.54	5.0	61.54	0.3	65.82	1.4	**61.17**	1.4	61.35	0.88
7	56.37	5.7	57.39	0.3	56.37	1.8	56.37	1.7	**56.36**	0.66
8	**51.17**	7.1	51.85	0.4	54.28	1.9	54.28	2.0	**51.17**	0.76
9	47.31	8.2	47.47	0.5	51.09	2.7	51.09	2.7	**47.29**	0.83
10	42.56	9.8	43.08	0.5	43.45	3.0	46.02	2.9	**42.42**	0.78
11	**40.36**	11.2	43.05	0.6	40.86	3.1	40.42	2.8	40.39	0.49
12	36.56	12.9	38.29	0.6	35.99	3.2	35.99	3.1	**35.76**	0.46
13	**35.1**8	14.9	37.92	0.8	35.72	6.2	35.72	5.9	35.25	0.63
14	31.49	16.4	32.65	0.6	34.50	7.1	34.50	6.5	**31.07**	1.35
15	28.10	18.6	30.52	0.7	28.95	6.5	28.95	6.7	**28.07**	0.57
16	26.54	20.5	27.91	0.8	29.19	7.5	29.19	7.8	**25.86**	0.58
17	26.07	22.8	26.81	0.7	26.66	4.0	**25.84**	3.8	25.91	0.28
18	23.79	24.4	25.41	0.7	24.24	3.8	24.20	3.4	**23.73**	1.36
19	22.13	26.6	24.29	0.6	22.54	4.1	22.54	4.1	**21.69**	1.08
20	21.09	29.3	23.00	0.7	23.02	4.0	23.02	3.6	**20.00**	0.63
21	20.80	31.9	21.72	0.8	20.85	4.9	20.85	4.0	**18.77**	0.69
22	19.36	34.7	19.59	0.7	19.93	4.3	19.36	4.2	**17.27**	0.6
23	18.48	37.4	19.82	0.8	17.48	5.2	17.48	5.3	**16.18**	0.55
24	16.36	40.6	18.46	0.7	20.13	5.0	15.52	4.8	**14.7**8	1.30
25	15.66	43.4	16.38	0.7	17.06	4.5	17.06	4.3	**13.81**	0.44
Average	43.50	17.9	44.68	0.5	44.85	3.6	44.28	3.5	**42.91**	0.68

Table 2: Computational results of the test problem with 654 customers.

	Previous methods								**Presented method**	
	SSALA		**SSALA-VLSN**		**TPH-S**		**TPH-P**		**HGA**	
m	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**
5	**321970**	300.2	419366	12.2	**321970**	2.4	321974	2.2	321972.30	5.1
10	**164717**	1104.5	171004	10.8	**164717**	6.4	**164717**	7.9	167331.33	22.2
15	**134446**	2586.4	142417	12.3	**134446**	17.2	**134446**	24.3	143609.87	33.4
20	**107358**	4278.8	110056	14.9	108622	26.2	107362	22.3	109685.84	41.2
25	**77017.7**	6560.8	78879.8	14.4	78695.6	41.1	77019.3	28.5	78354.88	56.8
30	**78831.6**	9362.3	80008.8	17.3	78832.4	64.3	78832.4	64.1	84989.98	85.7
35	**70131.7**	13080.4	71331.8	19.6	70612.1	63.1	70741.6	77.8	73235.40	81.6
40	52711.4	17414.1	**52213**	23.4	57704.1	91.2	**52213**	87.4	52224.80	80.4
45	50323.5	21321.3	50687.1	24.1	51503.1	135.6	**50287.1**	140.4	51260.66	89.3
50	38511.7	25870.6	38511.7	21.4	44274.7	174.5	40422.1	174.1	**38223.60**	73.1
Average	**109601.86**	10187.9	121447.52	17.0	111137.70	62.2	109801.45	62.9	112088.87	56.9

can not be found [20]. We define $0 \leq \theta_j \leq 1$ as the priority of customer j's allocation. The closer the parameter θ_j is to 1, it indicates that the customer j is a borderline customer, it means, the distance from first closest and second closest facility are almost equal. If the allocation of a borderline customer changes from the first closest facility to the second closest facility, the amount of change in the objective function will be very inconsiderable. Contrary to the uncapacitated problem, in the capacitated problem, the capacity of some facilities is completed and some are not. Completed capacity facilities are usually located in a place close to many

Table 3: Computational results of the test problem with 1060 customers.

	Previous methods								Presented method	
	SSALA		**SSALA-VLSN**		**TPH-S**		**TPH-P**		**HGA**	
m	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**	**VALUE**	**CPU**
5	**1870070**	801.9	1982440	28.0	**1870070**	4.0	**1870070**	3.7	1870125.67	36.9
10	1282800	3487.5	1331420	31.5	1282490	18.9	**1283840**	21.8	1285856.54	81.7
15	**996270**	7589.7	1032960	34.0	1006820	51.7	1006190	31.7	999521.96	97.9
20	**848121**	12526.3	873307	36.5	854624	78.1	866235	109.4	851943.7	135.5
25	**750596**	20756.6	779328	41.5	764416	181.1	751215	123.7	758849.63	197.4
30	**664243**	27303.4	688912	46.5	671964	195.1	674342	157.6	669013.77	199.53
35	**598097**	40235.3	610757	63.5	598208	294.1	614270	157.3	599051.33	186.2
40	**566357**	56384.4	578769	74.0	583318	288.1	597698	398.6	569867.16	307.2
45	533346	67218.3	546675	87.0	546168	323.2	**533241**	480.3	536444.16	381.1
50	483096	74328	490115	104.5	488748	556.6	486715	503.7	**482903.94**	201.8
Average	**859299.6**	31063.2	891468.3	54.7	866682.6	199.1	868381.6	198.8	862357.87	182.5

customers. Therefore, with changing customer allocation, different solutions can be obtained. So some customers should be allocated to the second or even the third closest facility due to the capacity completion of the first closest facility. By releasing a borderline customer from a facility with completed capacity, we create an empty space so that another customer can be allocated to its first closest facility. The proposed local search algorithm is described as follows:

Require: $m,\ n,\ f,\ X_i,\ q_i,\ c_i,\ z_{ij}\ for\ i = 1, 2, \dots, m,\ j = 1, 2, \dots, n.$
Denote:
T_i: the set of allocated customers to facility i, $i = 1, 2, \dots, m$,
$q_i = |T_i|$: the amount of customer demands that are met by facility i, $i = 1, 2, \dots, m$.
Calculate distances $d(X_i, A_j)\ \ for\ \ i = 1, 2, \dots, m, j = 1, 2, \dots, n$.
Denote for $j = 1, 2, \dots, n$:
E_j: the index of the facility that the customer j is allocated to it.
E'_j: the index of the first closest facility to the customer j.
E''_j: the index of the second closest facility to the customer j.
θ_j: the priority of the customer j.
Set $f \to f^\star,\ X_i \to X_i^\star,\ T_i \to T_i^\star,\ q_i \to q_i^\star\ for\ i = 1, 2, \dots, m.$
repeat
  Calculate $\theta_j = \frac{d(X_{E'_j}, A_j)}{d(X_{E''_j}, A_j)}$ for $j = 1, 2, \dots .n$.
  Set $\alpha_{min} = \infty$.
  for k=1 to n **do**
    Set $T_i \to T'_i,\ q_i \to q'_i\ for\ i = 1, 2, \dots, m.$
    Denote $\alpha_k = 0$ be amount of difference in the total cost that is created by releasing a number of customers and reassigning each of them to a new facility.
    if $k \notin T'_{E'_k}$ **then**
      Release the customer k and set $q'_{E_k} -= w_k$.
      Set $\alpha_k -= d(X_{E_k}, A_k)$.
      if $c_{E'_k} - q'_{E'_k} < w_k$ **then**
        Release the customer j with minimum θ_j among all allocated customers to the facility E'_k and set $q'_{E'_k} -= w_j$.
      end if
      if $c_{E'_j} - q'_{E'_j} < w_j$ **then**
        repeat
          Set $j \to j'$
          Release the customer j with minimum θ_j among all allocated customers to the facility $E'_{j'}$ and set $q'_{E'_{j'}} -= w_j$.
          Set $\alpha_k -= d(X_{E'_{j'}}, A_j)$.
        until $c_{E'_j} - q'_{E'_j} \geq w_j$
      end if

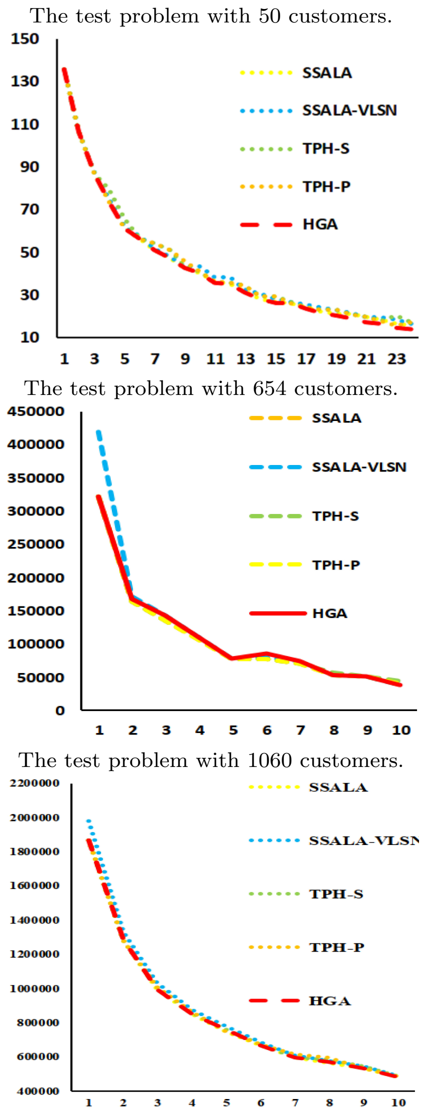

Figure 1: Summary of obtained results.

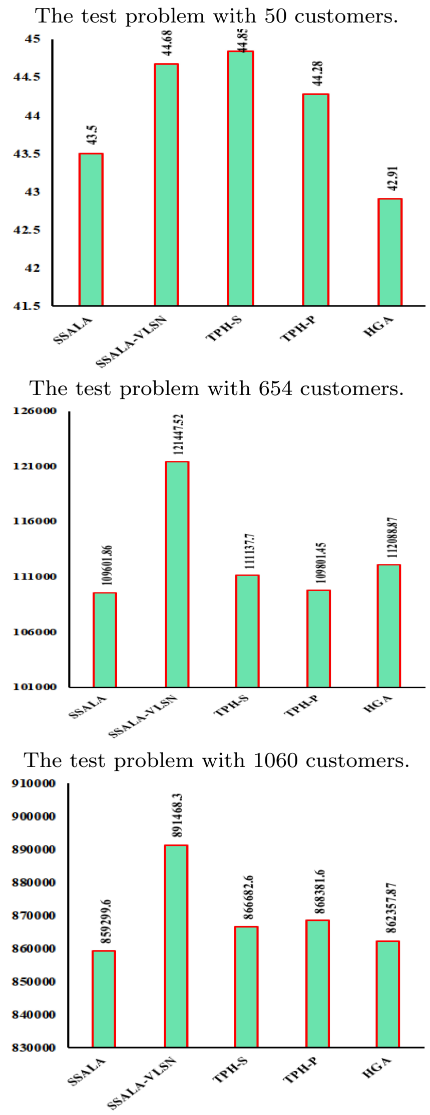

Figure 2: Summary of Average results.

Assign the customer k to E'_k and set $q'_{E'_k} += w_k$.
Set $\alpha_k += d(X_{E'_k}, A_k)$.
repeat
Let j be the index of the unallocated customer with minimum θ_j and i be the index of the nearest facility with enough capacity to the customer j. Assign the customer j to the facility i and set $q'_i += w_j$.
Set $\alpha_k += d(X_i, A_j)$.
until there is an unallocated customer
if $\alpha_k \leq \alpha_{min}$ **then**
Set $T'_i \to T^\star_i$, $q'_i \to q^\star_i$ for $i = 1, 2, \ldots, m$.
end if
end if
end for
for $i = 1$ to m **do**
Determine the new coordinates of facility i using Weiszfeld's equations [48]:

$$x^\star_i = \frac{\sum_{j \in T^\star_i} \frac{w_j \times a_j}{d(X_i, A_j)}}{\sum_{j \in T^\star_i} \frac{w_j}{d(X_i, A_j)}}$$

$$y^\star_i = \frac{\sum_{j \in T^\star_i} \frac{w_j \times b_j}{d(X_i, A_j)}}{\sum_{j \in T^\star_i} \frac{w_j}{d(X_i, A_j)}}$$

end for

Calculate new distances $d(X_i^\star, A_j)$ for $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.
Calculate total cost of obtained new solution as:
$f^\star = \sum_{i=1}^{m} \sum_{j \in T_i^\star} w_j \times d(X_i, A_j)$.
if $f^\star < f$ **then**
  Set $f^\star \to f,\ X_i^\star \to X_i,\ T_i^\star \to T_i,\ q_i^\star \to q_i$ for $i = 1, 2, \ldots, m$.
end if
until $f^\star \le f$
return $f,\ X_i,\ q_i,\ z_{ij}$ for $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.

4.4 The proposed mutation approach

The mutation operator creates small random changes on one parent in the population, so as a result a new child is produced. We apply the mutation operator to all members of the population, so each individual of the population as a parent produces a new offspring. The main steps of the proposed mutation algorithm are as follows:

Require: $m,\ n,\ f,\ c_i,\ q_i,\ X_i,\ z_{ij}$ for $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.
Denote:
T_i: the set of allocated customers to facility i, $i = 1, 2, \ldots, m$,
$q_i = |T_i|$: the amount of customer demands that are met by facility i, $i = 1, 2, \ldots, m$.
Calculate distances $d(X_i, A_j)$ for $i = 1, 2, \ldots, m, j = 1, 2, \ldots, n$.
Denote for $j = 1, 2, \ldots, n$:
E_j: the index of the facility that the customer j is allocated to it.
E'_j: the index of the first closest facility to the customer j.
E''_j: the index of the second closest facility to the customer j.
θ_j: the priority of the customer j.
for $i = 1$ to m **do**
  if $q_i == c_i$ **then**
    Select customer j randomly from the set T_i.
    Move the facility i to the customer location and set $X_i = A_j$.
  end if
end for
Calculate new distances $d(X_i, A_j)$ for $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.
Calculate $\theta_j = \frac{d(X_{E'_j}, A_j)}{d(X_{E''_j}, A_j)}$ for $j = 1, 2, \ldots. n$.
Set $q_i = 0,\ y_{ij} = 0$ for $i = 1, 2, \ldots, m, j = 1, 2, \ldots, n$.
repeat
  Let j be the index of the unallocated customer with minimum θ_j and i be the index of the nearest facility with enough capacity to the customer j. Assign the customer j to the facility i and set $q_i+ = w_j$, $Y_{ij} = 1$.
until there is an unallocated customer
Calculate total cost of obtained new solution as:
$f = \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ij} \times w_j \times d(X_i, A_j)$.
return $f,\ X_i,\ q_i,\ z_{ij}$ for $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$.

4.5 New population

A new population must be created among the individuals and children of the current population. The method of selecting solutions to form the next population has a great influence on the performance of the Genetic Algorithm. In order to avoid trapping in a local optima, we form the next population in such a way that from all the individuals of the current population and the children produced from them, we select some of the best and some at random.

4.6 Stopping condition

The Genetic Algorithm is repeated from one population to another until it reaches the stopping condition. We use the maximum number of iterations in this study as a criterion to stop the search process.

4.7 Computational experiments

In this section, we present the computational results obtained from the experiments performed on the presented Hybrid Genetic Algorithm. The presented algorithm was coded with C++ and has been executed on a Core i5 Laptop with processor specifications of 2 GHZ and 4 GB RAM. Temel Oncan [40] presented heuristic algorithms to solve the SSCMFWP and compared obtained results with those of two-phase algorithm presented by Manzour-al-Ajdad et al. [37]. Given that there are very few studies on the SSCMFWP, we compared the performance of our algorithm with the results available in Temel Oncan [40]. We evaluated the performance of algorithms in terms of solution quality. Given that we did not have the pseudo code of previous methods to run on the same computer in the same conditions, we evaluated the performance of our method according to the average results. We used the proposed Genetic Algorithm 10 times on each sample problem and recorded the best obtained result.

4.8 Sample Problems

To do our computational experiments, we used the data set which had been used by Brimberg et al [8]. The first data set with 50 customers is available in Eilon et al. [17] and the second and third data sets with 654 and 1060 customers are listed in the TSP library [43]. In all data sets, customer demand is equal to 1. The capacity of each facility is considered to be equal to:

$$c_i = \lceil \frac{\sum_{j=1}^{n} w_j}{m} \rceil, \; i = 1, 2, \ldots, m.$$

4.9 Parameters

The following parameters were obtained by a preliminary experiment.

$P = 5$: the population size.
$k' = 2$: the number of best solutions to produce new population.
$k'' = 3$: the number of random solutions to produce new population.
Maximum number of iterations:

$$I_{max} = \begin{cases} 100; & \text{for } n = 50 \\ 40; & \text{for } n = 654 \\ 40; & \text{for } n = 1060 \end{cases}$$

4.10 Computational Results

The computational results of the performed experiments on the test problems with 50, 654 and 1060 customers are presented in Table 1, Table 2 and Table 3, respectively. In all tables, the title "Previous methods" points to the previous available methods of this problem, and the title "presented method" describes the method presented in this study. The second line in each table represents the name of each algorithm, in which the titles "HGA" refers to the presented Hybrid Genetic Algorithm. The title "VALUE" under the name of each algorithm represents the obtained results of each method, and the title "CPU" denotes the computational time of each method. The first column in each table entitled "m", represents the number of facilities. Figure 1 summarizes the obtained results of the computational experiments and Figure 2 summarizes the average of the obtained results of methods. In the first test problem with 50 customers, HGA has obtained the lowest average results among all other methods. Although computational time is not considered as a comparative criterion, it is observed that the HGA obtained this result at a fairly reasonable time. In addition, in 20 instances, HGA has obtained the best result among the other methods. In the test problem with 654 customers, despite the fact that the HGA did not achieve the lowest average but it has a little variance with other methods. In the test problem with 1060 customers, among all methods except SSALA, the lowest average results has been obtained by HGA.

5 Conclusion

In this study, we considered the SSCMFWP and presented a Hybrid Genetic Algorithm. The presented algorithm was evaluated on three well-known test problems in the literature and the results were compared with some available methods in the literature. In general, for all test problems in comparison with other methods, there was no distant point among the results obtained by the HGA. In addition, HGA obtained the best results among the other methods in some instances. Extending the exact and heuristic methods, taking into account the fixed cost of establishing facilities and a modified version of this problem, in which customers have different demands, can be attempted in future studies.

References

[1] M. H. Akyuz, I. K. Altinel, T. Oncan, Location and allocation based branch and bound algorithms for the capacitated multi-facility Weber problem,*Annals of Operations Research* 222 (2014) 45-71.

[2] M. H. Akyuz, T. Oncan, Branch and bound algorithms for solving the multi-commodity capacitated multi-facility Weber problem, *Annals of Operations Research* 279 (2019) 1-42.

[3] N. Aras, M. Orbay, New heuristic methods for the capacitated multifacility Weber problem, *Naval Research Logistics* (NRL) 54 (2007) 21-32.

[4] J. P. Arnaout, Ant colony optimization algorithm for the Euclidean location-allocation problem with unknown number of facilities, *Journal of Intelligent Manufacturing* 24 (2013) 45-54.

[5] I. Bongartz, P. H. Calamai, A. R. Conn, A projection method for lp norm location-allocation problems, *Mathematical programming* 66 (1994) 283-312.

[6] J. Brimberg, Z. Drezner, A new heuristic for solving the p-median problem in the plane, *Computers & Operations Research* 40 (2013) 427-437.

[7] J. Brimberg, Z. Drezner, N. Mladenovic, S. Salhi, Generating good starting solutions for the p-median problem in the plane, *Electronic Notes in Discrete Mathematics* 39 (2012) 225-232.

[8] J. Brimberg, P. Hansen, N. Mladenovic, E. D. Taillard, Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem, *Operations research* 48 (2000) 444-460.

[9] J. Brimberg, P. Hansen, N. Mladenovic, S. Salhi, A survey of solution methods for the continuous location-allocation problem, *International Journal of Operations Research* 5 (2008) 1-12.

[10] J. BRIMBERG, N. Mladenovic, Solving the continus location-allocation problem with tabu search, *Studes in location annals* 8 (1996) 23-32.

[11] J. Brimberg, A variable neighborhood algorithm for solving the continuous location-allocation problem,*Studies in Locational Analysis* 10 (1996) 1-12.

[12] L. Cooper, Location-allocation problems, *Operations research* 11 (1963) 331-343.

[13] L. Cooper, Heuristic methods for location-allocation problems, *SIAM review* 6 (1964) 37-53.

[14] L. Cooper, The transportation-location problem, *Operations Research* 20 (1972) 94-108.

[15] Z. Drezner, J. Brimberg, N. Mladenovic, S. Salhi, New heuristic algorithms for solving the planar p-median problem, *Computers & Operations Research* 62 (2015) 296-304.

[16] Z. Drezner, J. Brimberg, N. Mladenovic, S. Salhi, New local searches for solving the multi-source Weber problem, *Annals of Operations Research* 246 (2016) 181-203.

[17] S. Eilon, C. D. T. Watson-Gandy, N. Christofides, R. de Neufville, Distribution management-mathematical modelling and practical analysis, *IEEE Transactions on Systems, Man, and Cybernetics* 6 (1974) 589-589.

[18] M. D. H. Gamal, S. Salhi, A cellular heuristic for the multisource Weber problem, *Computers & Operations Research* 30 (2003) 1609-1624.

[19] M. D. H. Gamal, S. Salhi, Constructive heuristics for the uncapacitated continuous location-allocation problem, *Journal of the Operational Research Society* 52 (2001) 821-829.

[20] A. Ghaderi, M. S. Jabalameli, F. Barzinpour, R. Rahmaniani, An efficient hybrid particle swarm optimization algorithm for solving the uncapacitated continuous

location-allocation problem, *Networks and Spatial Economics* 12 (2012) 421-439.

[21] D. Gong, M. Gen, G. Yamazaki, W. Xu, Hybrid evolutionary method for capacitated location-allocation problem, *Computers & industrial engineering* 33 (1997) 577-580.

[22] P. Hansen, N. Mladenovic, E. Taillard, Heuristic solution of the multisource Weber problem as a p-median problem, *Operations Research Letters* 22 (1998) 55-62.

[23] S. J. Hosseininezhad, S. Salhi, M. S. Jabalameli, A cross entropy-based heuristic for the capacitated multi-source weber problem with facility fixed cost, *Computers & Industrial Engineering* 83 (2015) 151-158.

[24] C. A. Irawan, M. Luis, S. Salhi, A. Imran, The incorporation of fixed cost and multilevel capacities into the discrete and continuous single source capacitated facility location problem, *Annals of Operations Research* 275 (2019) 367-392.

[25] C. A. Irawan, S. Salhi, K. Soemadi, The continuous single-source capacitated multifacility Weber problem with setup costs: formulation and solution methods, *Journal of Global Optimization* 78 (2020) 271-294.

[26] M. S. Jabalameli, A. Ghaderi, Hybrid algorithms for the uncapacitated continuous location-allocation problem, *The International Journal of Advanced Manufacturing Technology* 37 (2008) 202-209.

[27] R. E. Kuenne, R. M. Soland, Exact and approximate solutions to the multisource Weber problem, *Mathematical Programming* 3 (1972) 193-209.

[28] C. L. Lara, F. Trespalacios, I. E. Grossmann, Global optimization algorithm for capacitated multi-facility continuous location-allocation problems, *Journal of Global Optimization* 71 (2018) 871-889.

[29] C. M. Liu, R. L. Kao, A. H. Wang, Solving location-allocation problems with rectilinear distances by simulated annealing, *Journal of the Operational Research Society* 45 (1994) 1304-1315.

[30] R. F. Love, H. Juel, Properties and solution methods for large location-allocation problems, *Journal of the Operational Research Society* 33 (1982) 443-452.

[31] R. F. Love, J. G. Morris, G. O. Wesolowsky, Facilities Location: Models & Methods, *North-Holland* 18 (1988) 51-60.

[32] M. Luis, M. F. Ramli, A. Lin, A greedy heuristic algorithm for solving the capacitated planar multi-facility location-allocation problem, *AIP Conference Proceedings* 1782 (2016) 040010.

[33] M. Luis, S. Salhi, G. Nagy, Region-rejection based heuristics for the capacitated multi-source Weber problem, *Computers & Operations Research* 36 (2009) 2007-2017.

[34] M. Luis, S. Salhi, G. Nagy, A guided reactive GRASP for the capacitated multi-source Weber problem, *Computers & Operations Research* 38 (2011) 1014-1024.

[35] M. Luis, S. Salhi, G. Nagy, A constructive method and a guided hybrid grasp for the capacitated multi-source weber problem in the presence of fixed cost, *Journal of Algorithms & Computational Technology* 9 (2015) 215-232.

[36] H. Manzour, A. Torabi, M. S. Pishvaee, New heuristic methods for the single-source capacitated multi facility Weber problem, *The International Journal of Advanced Manufacturing Technology* 69 (2013) 1569-1579.

[37] S. M. H. Manzour-al-Ajdad, S. A. Torabi, K. Eshghi, Single-source capacitated multifacility weber problem an iterative two phase heuristic algorithm, *Computers & Operations Research* 39 (2012) 1465-1476.

[38] N. Mladenovic, J. Brimberg, P. Hansen, J. A. Moreno-Prez, The p-median problem: A survey of metaheuristic approaches, *European Journal of Operational Research* 179 (2007) 927-939.

[39] J. Moreno, C. Rodrguez, N. Jimnez, Heuristic cluster algorithm for multiple facility location-allocation problem, *RAIRO-Operations Research-Recherche Oprationnelle* 25 (1991) 97-107.

[40] T. Oncan, Heuristics for the single source capacitated multi-facility Weber problem, *Computers & Industrial Engineering* 64 (2013) 959-971.

[41] M. N. Neema, K. M. Maniruzzaman, A. Ohgai, New genetic algorithms based approaches to continuous p-median problem, *Networks and Spatial Economics* 11 (2011) 83-99.

[42] A. R. Dehkordi, The optimal solution set of the multi-source Weber problem, *Bulletin of the Iranian Mathematical Society* 45 (2019) 495-514.

[43] G. Reinelt, TSPLIB A traveling salesman problem library, *ORSA journal on computing* 3 (1991) 376-384.

[44] M. G. Resende, R. F. Werneck, A hybrid heuristic for the p-median problem, *Journal of heuristics* 10 (2004) 59-88.

[45] M. G. Resende, R. F. Werneck, A fast swap-based local search procedure for location problems, *Annals of Operations Research* 150 (2007) 205-230.

[46] S. Salhi, M. D. H. Gamal, A genetic algorithm based approach for the uncapacitated continuous location allocation problem, *Annals of Operations Research* 123 (2003) 203-222.

[47] M. B. Teitz, P. Bart, Heuristic methods for estimating the generalized vertex median of a weighted graph, *Operations research* 16 (1968) 955-961.

[48] E. Weiszfeld, Sur le point pour lequel la somme des distances de n points donns est minimum, *Tohoku Mathematical Journal, First Series* 43 (1937) 355-386.

[49] Z. M. Zainuddin, S. Salhi, A perturbation-based heuristic for the capacitated multi-source Weber problem,*European Journal of Operational Research* 179 (2007) 1194-1207.

Saeed Jahadi received his MSc degree from Urmia University of Technology in 2014. His research areas of interest include facility layout and location, transportation planning, optimization, and distribution systems.

Maghsud Solimanpur is a Professor of Industrial Engineering at the Department of Industrial Engineering, Urmia University, Urmia, Iran. He has received his BSc and MSc degrees from Sharif University of Technology in 1995 and 1997, respectively. He received his PhD degree from Indian Institute of Technology (IIT) Delhi, India in 2003. His research areas of interest include production and operations management, operations research, cellular manufacturing systems, facilities planning and layout, etc.