Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

A novel approach for solving a capacitated location allocation problem

Masoud Rabbani* **and Helia Yousefnejad**

Department of Industrial Engineering,College of Engineering, University of Tehran, Tehran, Iran

C H R O N I C L E

Article history:
Received October252012
Received in revised format December 28 2012
Accepted February4 2013
Available online 10February2013

Keywords:
Location
Capacitated Location Allocation
Graph Theory
Genetic Algorithm

A B S T R A C T

Location Allocation is one of the most important decision making problems, which attracted many operational researchers during recent decades and many solution procedures are developed so far to cope with this problem. This paper proposes a new graph theory based method to cope with small size capacitated location allocation problems. Additionally, a genetic algorithm is utilized to solve medium and large scale problems. Finally, through some computational experiments, the quality and capability of these algorithms are shown.

1. Introduction

Location Allocation (LA) is a strategic decision making problem concerns with choosing the best set of logistics depots between potential locations and allocating customers to the selected locations. This problem has wide applications in real life issues such as locating health care services, building telecommunication networks and so on. Solving this problem also plays an important role in supply chain network design. Making the best decision, impacts the economics of the organization and is considered as a competitive advantage in national and international markets. An inefficient decision making can lead to poor service quality towards customers, long delivery times, and high investment for the logistics operators, which can badly effect business operations and its profitability. Due to all these reasons LA problem has been studied by many researchers since Cooper (1963) proposed it for the first time. Since then many approaches are developed to tackle this problem. For a detailed and comprehensive review on facility location allocation models and solution methods, interested readers are referred to Owen and Daskin (1998), Mello et al. (2009), Mohammadi et al. (2011) and Yazdianand and Shahanaghi (2011). The definition of the problem, in general, is to choose the optimal subset of facility locations from a set of candidate sites in order to satisfy the demand nodes. To solve a LA

* Corresponding author. Tel.: +98 21 88335605.
E-mail: mrabani@ut.ac.ir (M. Rabbani)

doi: 10.5267/j.ijiec.2013.02.003

problem the following questions should be answered: How many facilities should be opened, on which candidate sites the opened facilities should be located and eventually each opened facility should covers which demand nodes, in such a way that the total cost (including transportation and fixed location cost) is minimized (Cooper, 1963; Krarup & Pruzan, 1983). Cooper (1963) categorized the LA problem into two different classes. One of them is called Uncapacitated LA problem, which is studied in detail by many researchers such as Logendran and Terrell (1988) and Brimberg et al. (2000). In this category of LA problems the capacity of facilities are limitless and it is obvious that each customer should be supplied by the nearest facility. Otherwise, in Capacitated LA problem, which is the second class of LA problems, each facility is able to cover a limited amount of demand and it should be determined how much demand of each node should be covered by each facility. The capacitated LA problem is also considered by many researchers including Ernst and Krishnamoorthy (1999), Barreto et al. (2007) and Zhou and Liu (2007). After this categorization, in order to solve the LA problem many approaches are developed considering different location criteria such as cost, time, coverage, and accessibility between locations.

Many researchers including Love and Moris (1975), Anandand Knott (1986) and Badri (1999) consider exact approaches that involve the use of techniques such as linear programming, integer programming and multi objective optimization. These methods can get the correct answer of the problem, but the calculating scale and store content will be increased nonlinearly with the addition of parameters in the model, so it would be helpful to use other approaches such as heuristic or meta-heuristic methods to solve this problem. Therefore in recent years many of researches such as Charnes et al. (1978), Brimberg et al. (2000) and Bischoff and Dächert (2009) tried to tackle the LA problem in this way. They tried to find the best near optimal solution for the LA problem as a NP-hard one, by proposing different heuristic or meta-heuristic methods. Especially the application of Genetic Algorithm (GA) in location allocation problem has been investigated by several researchers such as Beasley and Chu (1996) and Zhou et al. (2002,2003), Zhou and Liu (2007) and Marian and Luong (2008). Here, as a new approach, a LA problem is solved using Graph Theory methods. As it is important to have a visual understanding of a problem, using graph theory as a capable concept to visualize, model and solve a problem, can be helpful. As the first step, a graph drawing of the problem is presented in Fig. 1 to better understanding, then in next sections the location of facilities on the candidate sites and assignment of demand nodes to each located facility are determined such that the total cost including transportation and fixed location costs is minimized.

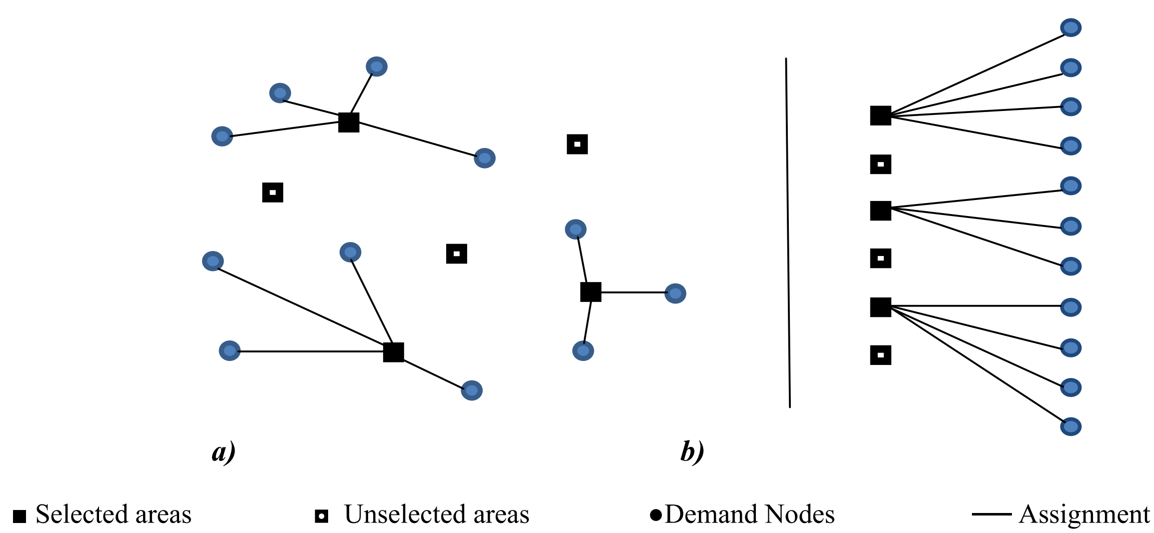


Fig.1. Graph Representation of a Location Allocation problem

In Fig.1, a graph with two different kinds of nodes is defined. The cycle nodes are used to show the demand centers and the square shaped nodes are used to show the candidate sites. When a candidate area turns to black it means that it is selected to locate a facility in it and an edge between these two kind of nodes means that the demand point is covered by the selected area. This graph also can be represented as a *bipartite graph* in which the nodes set is divided into two subsets where there is no edge between the nodes in the same sub set and each node relates one of the nodes from the first subset with another node in the other subset. In this problem

To sum up, we have a complete bipartite graph and we need to find a sub graph of it with the minimum sum of weights.To do so, the remainder of the paper is structured as follows. In Section 2 a more detailed definition of the aforementioned problem is presented, then in Section 3 the solution methodology is described in detail for both small and large size problems. Finally, in Section 4 some experimental results are presented in order to evaluate the quality of proposed algorithms.

2. Problem Definition

As mentioned before, a location–allocation problem is discussed in this research. Having capacitated facilities, the formulation of the abovementioned problem is as follows:

There are ***I*** demand nodes, in which ***P*** products are required and the required amount of each of which is specific. In the other hand, there exist ***J*** candidate sites to be considered in order to locate ***K*** facilities. The transportation cost of unit product from each candidate site to each demand node and the fixed location cost of each facility on each candidate site are known. To cover the demands of each node, facilities use ***S*** different resources that the available level of each of which is known in each candidate site:

$$\min \sum_{p=1}^{P}\sum_{k=1}^{K}\sum_{j=1}^{J}\sum_{i=1}^{I} x_{kj} y_{ji} c_{ji} b_{jip} + \sum_{k=1}^{K}\sum_{j=1}^{J} F_{kj} x_{kj}, \tag{1}$$

subject to

$$\sum_{j=1}^{J} y_{ji} = 1; \qquad \forall i, \tag{2}$$

$$\sum_{k=1}^{K} x_{kj} \le 1; \qquad \forall j, \tag{3}$$

$$\sum_{j=1}^{J} x_{kj} \le 1; \qquad \forall k, \tag{4}$$

$$y_{ji} \le \sum_{k=1}^{K} x_{kj}; \qquad \forall j, i, \tag{5}$$

$$x_{kj} \le \sum_{i=1}^{I} y_{ji}; \qquad \forall k, j, \tag{6}$$

$$b_{jip} \ge y_{ji}; \qquad \forall p, j, i, \tag{7}$$

$$\sum_{j=1}^{J} b_{jip} = D_{ip}; \qquad \forall i, p, \tag{8}$$

$$b_{jip} \le \min\left[D_{ip}, \sum_{k=1}^{K} x_{kj} cap_{kjp} \right]; \qquad \forall j, i, p, \tag{9}$$

$$x_{kj}, y_{ji} \in \{0,1\}. \tag{10}$$

where ***k*** = 1,2, . . . ,***K*** is the index of facility numbers, ***j*** = 1,2,. . . , ***J*** is the index of candidate sites, and ***i*** = 1,2,. . . ,***I*** is the index of demand nodes. The parameters and decision variables of the proposed model are as follows:

*C_{ji} = Transportation cost of unit product from the **j**th candidate site to the **i**th demand node. This unit cost is assumed to be equal for all **P** products.*

*F_{kj} = Fixed cost of locating the **k**th facility on the **j**th candidate site.*

*D_{pi}= Demand of the **p**th product of the **i**th node.*

$$x_{kj} = \begin{cases} 1 & \text{if facility } k \text{ is located on candidate site } j \\ 0 & \text{Otherwise,} \end{cases}$$

$$y_{ji} = \begin{cases} 1 & \text{if demand node } i \text{ is covered by the facility located on candidate site } j \\ 0 & \text{Otherwise,} \end{cases}$$

*CAP_{kjp} = Capacity of the **k**th facility located in the **j**th candidate site to supply the **p**th product.*

*b_{jip} = Amount of the **p**th product of **i**th demand node allotted to the facility located on **j**th candidate site.*

It should be noted that if y_{ji} = 1, all requirements of ***P*** products required in the ***i***th demand node is covered by the facility located on the ***j***th candidate site. The objective function of the presented model minimizes the total cost including transportation and fixed location costs. Eq. (2) implies that all requirements of demand node ***i*** should be served (i.e., consisting of the requirements of all ***P*** products). Constraints Eq. (3) indicate that not more than one facility can be located on one candidate site. On the other hand, Constraints Eq. (4) implies that each facility can be located on at most one candidate site. It is obvious that,first a facility should be located on the ***j***th candidate site and then it can serve the ***i***th demand node. This is satisfied by Eq. (5). Eq. (6) implies that the demand requirement of at least one node should be assigned to the located facilities. In other words, any located facility should not be idle.

Now the problem is how many facilities should be applied, the applied facilities should be located on which candidate sites and should cover which demand nodes. We are given a complete bipartite graph $G = (V,E)$ with bipartition $V = I \cup J$, where ***J*** refers to the set of potential sites to locate the facilities and ***I*** to the set of demand nodes. Establishing a facility ***j*** causes opening cost F_j. Attaching demand node ***i*** to an opened facility ***j*** yields connection cost c_{ji}. The problem is to find a subset $K \subseteq J$ of facilities to open and a mapping $a : I \rightarrow K$ for assigning demand nodes to opened facilities in a way that each demand node is connected to exactly one facility as to minimize the total opening and connection cost. In this problem each one of the demand nodes should be supplied by just one of the opened facilities, it means that every node from set ***I*** have a degree of one, while the facilities in the set ***J*** could be even isolated (not opened) or have a degree of more than one (attached to more than one demand node). In addition, a positive number is written on each edge as its weight, which can show either the distance between nodes, time between them or the cost of transporting objects between them. The Bipartite complete graph related to our problem considering aforementioned assumptions is represented in Fig.2.

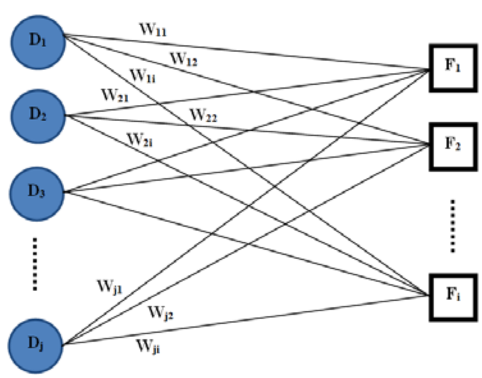


Fig. 2. Bipartite Representation of the Location Allocation Problem

3. Solution methodology

In order to solve the location allocation problem using graph theory methods, some definitions are required:

a. Matching

M is a matching in graph ***G*** if it contains a set of pairwise nonadjacent links. In a matching, the two vertices of each edge of ***M***are said to be matched under ***M***, and each node related to an edge of ***M*** is said to be covered by ***M***. If all of the graph nodes are covered by matching ***M***, it is said to be a perfect matching (Bondy & Murty, 2008).

b. Bipartite Graph

In a bipartite graph $G = (U,V,E)$ nodes are divided into two disjoint sets ***U*** and ***V*** such that each edge (u_i, v_j) connects a node $u_i \in U$ and one $v_j \in V$. If each edge in graph ***G*** has an associated weight w_{ij} , the graph ***G*** is called a weighted bipartite graph (Bondy & Murty, 2008).

c. Matching in Bipartite Graphs

Given a bipartite graph $G = (U,V,E)$ of two node sets ***U*** and ***V*** with positive numbers as weights on its edges. The target is to find a minimum weighted matching in ***G*** i.e., a subset of edges with minimum aggregate weight ($w(M) = \sum_{e \in M} w(e)$), considering the limitation that no two chosen edges share an end-point (Bondy & Murty, 2008).

The following is an ILP formulation of the minimum weight perfect matching problem:

$$\min \quad \sum_{(u,v)} w(u,v)x(u,v) \tag{11}$$

$$\sum_{v} x(u,v) = 1 \qquad \forall u \in U \tag{12}$$

$$\sum_{u} x(u,v) = 1 \qquad \forall v \in V \tag{13}$$

$$x(u,v) \in \{0,1\} \qquad \forall u \in U, \forall v \in V \tag{14}$$

In the above formulation nodes$u \in U$ and $v \in V$are matched under ***M*** if and only if $x(u,v)$ gets a value of 1. Constraints Eq. (12,13) guaranty that each node in each subset ***U*** and ***V*** will be matched by one and only one node in the other subset.

d. Bipartite Semi- Matching

Now we discuss representing location-allocation problem as a bipartite matching problem, in which two node sets will be interpreted as facilities and demand nodes. We seek a matching of the two node sets with minimum weight. In the semi-matching problem, we allow nodes on one side to be used multiple times, which means that one of Eq. (12) or Eq. (13) set will be relaxed. This is appropriate here because usually multiple demand nodes may be served by the same facility, but not vice versa.

Bipartite Semi-Matching is similar to ordinary bipartite matching, except that the matching constraint is relaxed on one side, we now seek a subset of edges such that no two chosen edges share an endpoint in ***U***(Lovász & Plummer, 2009). The problem can be formulated for both cardinality and weighted settings. Both the weighted and unweighted matching models can be applied to location-allocation problem. Given a bipartite graph of facilities and demand nodes, the presence of an edge (U_i,V_j) indicates that U_i is capable of serving V_j . A weighted semi-matching model can also be used to solve the allocation problem. In this case, given a weighted bipartite graph of locations and demand nodes, each edge (U_i,V_j) is associated with a corresponding utility value which can be a measure of the outcome which is earned by connecting a demand node to a location. The motivation here is that each node wants to be covered, and different facilities may be capable of covering it, but some might get better.

e. Alternating Path

Considering a semi-matching Min G, an alternating path will be described as a sequence of edges $P = (\{v_1, u_1\}, \{u_1, v_2\}, \ldots, \{u_{k-1}, v_k\})$ with $v_i \in V$, $u_i \in U$, in which we have $\{v_i, u_i\} \in M$ for each i. Note that if $P$is an alternating path relative to a semi-matching Mthen $P \Delta M$, which is the symmetric difference of sets P and M, is also a semi-matching, derived from $M$by switching matching and non-matching edges along P. If the new semi-matching is feasible due to capacities of the facilities and if it has a lowercost, than $P$is called a *cost-reducing path* relative to M. Cost-reducing paths are so named because switching matching and non-matching edges along Pyields a semi-matching $P \Delta M$whose cost is less than the cost of M (Lovász & Plummer, 2009).

Using terms which is defined above, we are going to solve the location allocation problem by modeling the problem as a bipartite semi matching problem in a graph. Each U-vertex represents a demand node, and each V-vertex representsa possible location. For any demand node j and facility i, we set $p_{i,j} = 1$ if the edge $\{u_j, v_i\}$ exists in graph G, and otherwise $p_{i,j} = 0$. Clearly, any feasible semi-matching determines a location for the problem. In particular, an optimal semi-matching determines the optimal allocation that minimizesthe cost or distance ($\sum c_{ij} p_{ij}$). In order to solve the problem a graph theory based procedure is presented based on the following theorem (Ahuja et al., 1993):

Theorem.*A semi-matching **M** is optimal if and only if no cost-reducing path relative to **M** exists.*

In order to solve a location-allocation problem using graph theory methods a heuristic algorithm is presented below in four steps:

1. Generate an initial semi-matching, M.
2. ***While*** There is a cost-reducing path, P,
3. Reduce the cost of M, using$P$ and consider the new semi-matching as M.
4. ***Else*** Suppose M as the optimal answer.

Having a nearly optimal initial assignment is an important issue to get to the best possible answer after few iterations, so it would be helpful to use a heuristic method to find a near optimal initial solution.

3.1. Finding an Initial Semi-Matching:

Considering i demand nodes and j candidate facility locations in a capacitated location-allocation problem, a 6 step algorithm to generate an initial semi- matching, M is as follows:

1. Find F_i = Min $\{F_1, \ldots, F_m\}$ and consider the F_i vertex, which has the lowest opening cost, as the opened facility. If there is a tie choose the one with maximum capacity.
2. Find d_{ij}, the minimum value in the ith row of the incidence matrix of graph G.
3. Put $F_i = F_i - d_{ij}$ and set $P = \{D_k \mid D_k \leq F_i \ , k \neq i\}$.
4. If $P \neq \varnothing$ go to step 3 otherwise omit F_j from the network and go to step 1.
5. Do this till there is no unassigned demand node.
6. Compute the cost of this initial semi-matching.

3.2. Improve the Answer

Now after finding the initial solution, we use the concept of iteratively find and remove cost reducing paths to improve our solution.The method for finding cost-reducing paths is to apply a depth-first search (DFS) algorithm to grow a forest of alternating paths where each tree root is chosen to be an unused F -vertex with lowest opening cost for example $\boldsymbol{F_1}$. By switching matched and non-matched edges along the alternating path, if the new solution is feasible due to capacity constraints and if it leads to a better objective function, it would be accepted as the new best answer of the problem and the alternating path is considered as a cost reducing path. If such an answer is not found, we can say that there exists no cost reducing path from $\boldsymbol{F_1}$. Hence, we move to another unused vertex in $\boldsymbol{F}$ and construct a new alternating search tree from the vertex. If all vertices of $\boldsymbol{V}$are visited then the algorithm terminates. If no cost-reducing path exists then the optimal semi matching is found.

3.3. Large Size Problems

The above mentioned procedure can be used to solve small size problems. In order to solve medium and large size problems a Genetic Algorithm (GA) is utilized in this paper. In order to apply GA to aforementioned problem, the representation scheme for the chromosome is a***n***-bit string where ***n***represents the number of demand nodes. A non-zero value for the ***ith***bit implies that a facility is allocated to that demand node. If a facility is not present in the string, it implies that this facility was not openeddue to non-feasibility reasons (allocation of zero customers). Let us consider a network comprising of 10 demand nodes and 5 logistics facilities. The representation of an individual chromosome (solution) is illustrated as Fig.3.

Demand Nodes	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*

Fig. 3. Solution Representation of Location Allocation problem

Using case Fig.4 (a), we can say that facility 1 is allocated to demand nodes (1,2,3), facility 2 to demand nodes (4,5,6) and so on. By analyzing results for Fig.4 (b), we see that facility 1 is allocated to demand nodes (1,5,10), facility 3 to (4,6,8). However, in case (b) the logistic facility 5 is absent which means it was not opened for the reasons of zero allocation of demand nodes.

a)

Demand Nodes	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***
Facilities	***1***	***1***	***4***	***2***	***3***	***2***	***3***	***4***	***5***	***5***

b)

Demand Nodes	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***
Facilities	***1***	***4***	***4***	***3***	***1***	***3***	***3***	***3***	***2***	***1***

Fig.4. Examples of Solution Representation

The pseudo code for implementing GA is presented as follows (Gen, 2000):

1. Set iteration counter t = 0.

2. Generate the initial population, P(t)using the heuristic method described in Section 4.1.

3. Evaluate P(t) using the objective function.

4. While the termination condition is not met, do

4.1. Set t= t+ 1

4.2. Select pairs P_1 and P_2 from the feasible best ranked populations.

4.3. Replenish population by applying genetic operators to P_1 and P_2

4.3.1. Apply crossover operator to integrate P_1 and P_2 using single point crossover.

4.3.2. Apply mutation operator to the population with the best fitness.

4.3.3. Check the feasibility of the new child and evaluate the fitness of the feasible one using the objective function

5. Stop.

4. Numerical Result

4.1. Small Size Example

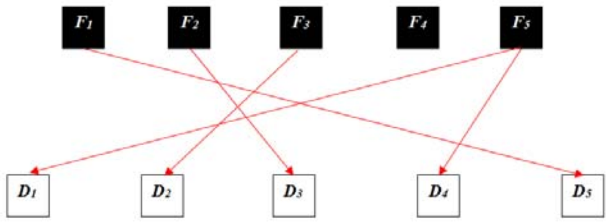
To illustrate our proposed framework, we present a numerical example. Consider a food processing company which is evaluating five potential plant location sites (F_1 to F_5) in five different cities. The production plants have to serve five distribution centers (D_1 to D_5). The investors need to determine which location site to open and how much transport from each location to each distribution center should be done. The resource data are tabulated in Table1 and Table 2. Because of the characteristics of the company and perishability of products it is important to deliver demands as soon as possible, so the distance between each candidate site and distribution center is tabulated in scale of time unit. Fig.5 shows the solution progress using the heuristic method described in Section 4.1.

Table 1
Data about the time between candidate locations and demand nodes

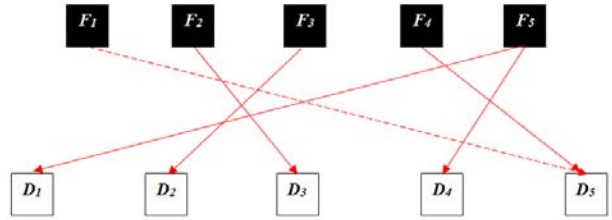
	Demand Nodes				
Candidate Locations	D_1	D_2	D_3	D_4	D_5
F_1	5	14	19	4	9
F_2	7	16	21	6	12
F_3	9	8	12	5	12
F_4	5	3	10	6	2
F_5	10	17	23	13	14
Monthly Demand	4200	4500	3350	900	2500

Table 2
Data about the production capacity and opening cost of candidate locations

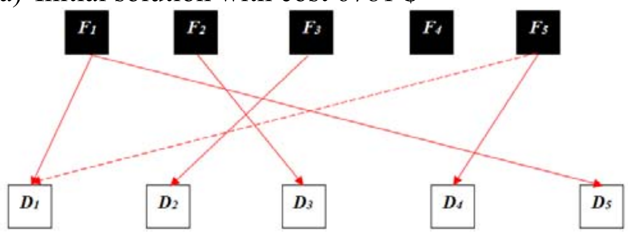
Candidate Locations	F_1	F_2	F_3	F_4	F_5
Production Capacity	5200	5000	5800	3250	5650
Fixed Cost	1450	2100	1720	2580	1450



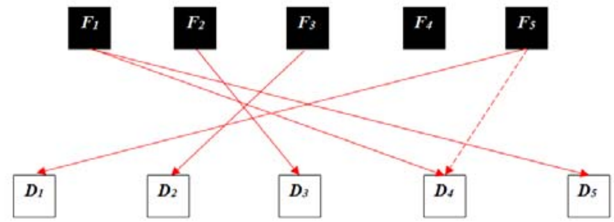
a) Initial solution with cost 6781 $



b) Second iteration with cost 7894 $



c) Second iteration : the solution is infeasible



d) Best solution with cost 6772 $

Fig.5. Step By StepSolution Progress of the Example

4.2. Medium and large Size problems

In this section, we evaluate the proposed algorithms in terms of solution quality and the required computation time. The programming model which is presented in Section 1 is implemented in GAMS and the associated results are compared with those of our graph theory based heuristic and meta-heuristic approach. Many problems with different parameters and values are considered and solved. The required input data of the test problems are randomly generated and all of the experiments are performed on a Pentium 4 with a Core i7 CPU processor. In order to show the proper performance of GA, different small sized problems are solved by GAMS software. As it is tabulated in Table 3, the results indicate that optimal values from GAMS and the proposed GA are the same in small sized problems. However the CPU times of the GA are considerably less than GAMS.

Table 3
Obtained Results of Small sized Test Problems from GAMS and GA

Problem Size	GAMS		GA		GAP (%)
(J×D)	OFV ($)	CPU Time (Sec.)	OFV ($)	CPU Time (Sec.)	
3×5	7130	1.438	7130	0.011	0
4×5	5176	2.406	5176	0.021	0
5×5	6760	17.656	6760	0.01	0
5×7	15448	4.5	15448	0.17	0
6×8	9200	17.985	9200	0.21	0

Moreover, 12 instances are considered for medium and large sized problems. The results are shown in Table 4 in terms of CPU time and obtained objective value. To improve the quality of the presented GA in medium and large sized problems, there are three parameters that should be tuned before the algorithm is used. These three parameters, which are related to the implementation of GA, are crossover probability, mutation probability and the population size. For computations of Table 3 and Table 4 these parameters are considered to be 0.8, 0.1 and 100, respectively. It should be noted that in each run of the algorithm the initial population is generated using the heuristic method of Section 3.1. In order to find the best set of parameters, a sensitivity analysis is done. In the first step the mutation rate and the population size are set to be 0.1 and 100, respectively and the crossover rate is varied between three levels. The obtained results are shown in Figure 6, where we make use of a common performance criterion, named relative percentage deviation (RPD) as the vertical axis. RPD is calculated as follows:

$$RPD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \times 100$$

where the Algsol is the obtained objective function for each problem by combining the parameters and Minsol is the minimum objective function value in all considered combinations. It is obvious from Figure 6 that the best crossover probability is 0.8. In addition the same process is done for the mutation rate and population size. For mutation rate two levels is considered and for population size four different levels are compared with the assumption of other parameters to be fixed. The obtained results are shown in Fig. 7 and Fig. 8, respectively. From these three charts it could be concluded that the best set of GA parameters are 0.8, 0.1 and 100, which were considered as our algorithm parameters in computations of Table 2 and 3. Besides, it follows from the charts that calculated RPD does not exceed 2.76% for each set of selected parameters, so it could be concluded that the proposed genetic algorithm is somehow robust to the parameter changing.

Table 4

Obtained Results of Medium and Large Sized Test Problems from the proposed GA

Problem Size (J×D)	OFV ($)	CPU Time (Sec.)	Problem Size (J×D)	OFV ($)	CPU Time (Sec.)
9×15	31493	25.5	15×30	218700	162.9
10×13	35739	34.95	20×30	127890	128.1
10×15	40011	16.3	20×45	73640	30.82
10×18	176500	77.4	30×40	150230	47.62
11×21	185600	33.5	30×55	83680	56.12
15×20	71002	74.75	50×70	210980	69.25

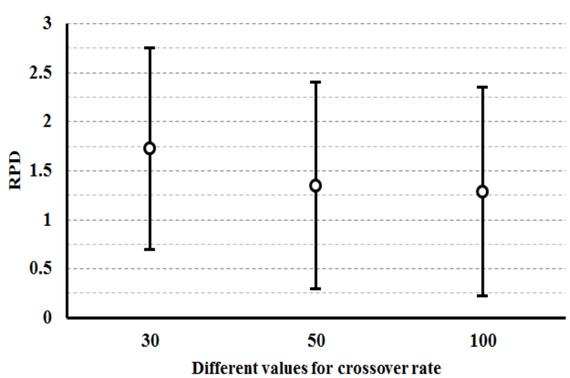


Fig.6. RPD Diagram with Respect to Combinations of Fixed Mutation Rates and Pop size and Different Crossover Rates

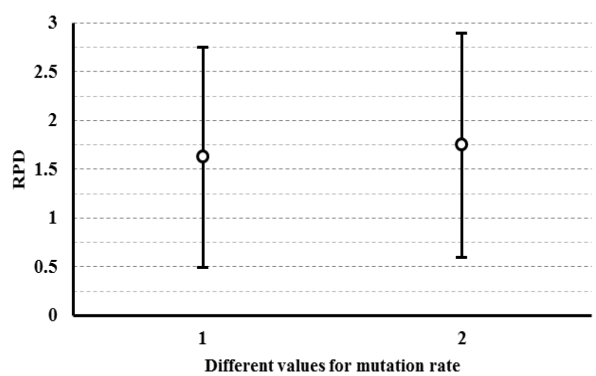


Fig.7. RPD Diagram with Respect to Combinations of Fixed Crossover Rates and Pop size and Different Mutation Rates

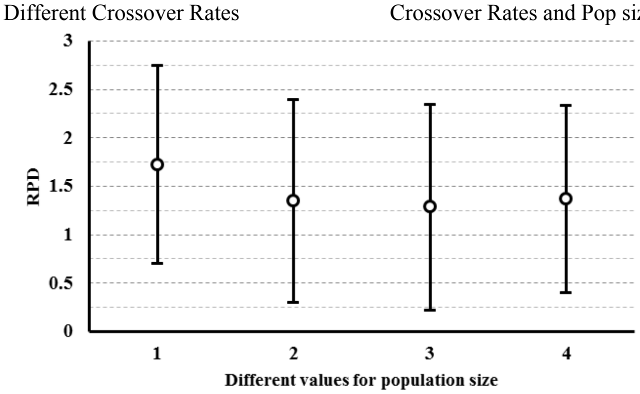


Fig.8. RPD Diagram Regarding the Population size

5. Conclusion

This paper addressed capacitated location allocation problem,which can be categorized into two sub-problems. Firstly, the location problem, that is which logistics facilities should be opened and where, and secondly, the allocation problem, that is how to perform customer allocations to logistics depots to ensure timely service for customers. The problem is studied under the assumption of having opening costs for facilities and only one criterion (time) is considered. As the solution methodology a graph theory based algorithm is presented.For large-scale problems, a genetic algorithm is proposed, then. Finally, implementation of the proposed model in some real casesis reported and sensitivity analysis is done to show its applicability in real world circumstances.To extend current direction of this paper, it can be practical to consider the case that a demand node can be served by more than one facility. In addition, hybrid meta-heuristic methods can be applied and the results can be compared with those obtained from this study.

References

Ahuja, R. K. A., Magnanti, T. L., &Orlin, J.B. (1993).*Network Flows: Theory, Algorithms and Applications*. Prentice Hall.

Anand, S., &Knott, K. (1986). Computer assisted models used in the solution of warehouse location-allocation problems. *Computers & Industrial Engineering*, 11(1-4), 100-104.

Badri, M.A. (1999). Combining the analytic hierarchy process and goal programming for global facility location-allocation problem. *International Journal of Production Economics*, 62 (3), 237-248.

Barreto, S, Ferreira, C, Paixão, J, &Santos, B.S. (2007).Using clustering analysis in a capacitated location-routing problem.*European Journal of Operational Research*, 179 (3), 968-977.

Beasley, J.E., &Chu, P.C. (1996).A genetic algorithm for the set covering problem.*European Journal of Operational Research*, 94, 392-404.

Bischoff, M., &Dächert, K. (2009).Allocation search methods for a generalized class of location–allocation problems.*European Journal of Operational Research*, 192 (3), 793-807.

Bondy, J.A., &Murty, U.S.R. (2008).*Graduate Texts in Mathematics. Graph Theory*. London: Springer.

Brimberg, J., Hansen, P., Mladenovic, N., & Taillard, E.D. (2000).Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem. *Operations Research*, 48 (3), 444-460.

Charnes, A., Cooper, W. W., &Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2, 429–444.

Cooper, L. (1963). Location-allocation problems.*Operations Research*, 11, 331-344, 1963.

Ernst, A.T., & Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86, 141-159.

Gen, M., (2000). *Genetic Algorithms & Engineering optimization*. NewYork: John Wiley.

Krarup, J., &Pruzan, P.M., (1983). The simple facility location problem: Survey and synthesis. *European Journal of Operational Research*, 12, 36–81.

Lovász, L., & Plummer, M. D. (2009).*Matching Theory*.AMS Chelsea Publishing.

Logendran, R., & Terrell, M.P. (1988).Uncapacitated plant location-allocation problems with price sensitive stochastic demands. *Computers and Operations Research*, 15 (2), 189-198.

Love, R. F., & Morris, J. G. (1975).A computation procedure for the exact solution of location–allocation problems with rectangular distances.*Naval Research Logistics*, 22, 441–53.

Marian, R. M., Luong, L. H. S., & Akararungruangkul, R. (2008).Optimisation of distribution, networks using genetic algorithms. part 1 - problem modelling and automatic generation of solutions. *International Journal of Manufacturing Technology and Management*, 15(1), 64-83.

Marian, R. M., Luong, L. H. S., & Akararungruangkul, R. (2008).Optimisation of distribution networks using genetic algorithms. part 2 - the genetic algorithm and genetic operators. *International Journal of Manufacturing Technology and Management*, 15(1), 84-101.

Melo, M.T., Nickel, S. & Saldanha-da-Gama, F. (2009). Facility location and supply chain management – A review. *European Journal of Operational Research*, 196, 401–412.

Mohammadi, M., Tavakkoli-Moghaddam, R., & Rostami, H. (2011).A multi-objective imperialist competitive algorithm for a capacitated hub covering location problem.*International Journal of Industrial Engineering Computations*, 2(3), 671-688.

Owen S.H. & Daskin, M. S. (1998). Strategic facility location: a review. *European Journal of Operational Research*, 111, 423–447.

Yazdian, S. A., & Shahanaghi, K. (2011).A multi-objective possibilistic programming approach for locating distribution centers and allocating customers' demands in supply chains.*International Journal of Industrial Engineering Computations*, 2(1), 193-202.

Zhou, G., Min, H., & Gen, M. (2002). The balanced allocation of customers to multiple distribution centers in the supply chain network: a genetic algorithm approach. *Computers and Industrial Engineering archive*, 43(1-2), 251 – 261.

Zhou, G., Min, H., & Gen M. (2003).A genetic algorithm approach to the bi-criteria allocation of customers to warehouses.*International Journal of Production Economics*, 86, 35–45.

Zhou, J., & Liu, B. (2007). Modeling capacitated location–allocation problem with fuzzy demands. *Computers & Industrial Engineering*, 53, 454–468.