



Efficient Solution Procedure to Develop Maximal Covering Location Problem Under Uncertainty (Using GA and Simulation)

K. Shahanaghi* & V.R. Ghezavati

K. Shahanaghi, Department of Industrial Engineering, Iran University Of Science and Technology, Tehran, Iran

V.R. Ghezavati, PhD student at the same Department, Tehran, Iran

Keywords

maximal covering;
location; genetic algorithm;
simulation;
uncertainty modeling

ABSTRACT

In this paper, we present the stochastic version of Maximal Covering Location Problem which optimizes both location and allocation decisions, concurrently. It's assumed that traveling time between customers and distribution centers (DCs) is uncertain and described by normal distribution function and if this time is less than coverage time, the customer can be allocated to DC. In classical models, traveling time between customers and facilities is assumed to be in a deterministic way and a customer is assumed to be covered completely if located within the critical coverage of the facility and not covered at all outside of the critical coverage. Indeed, solutions obtained are so sensitive to the determined traveling time. Therefore, we consider covering or not covering for customers in a probabilistic way and not certain which yields more flexibility and practicability for results and model. Considering this assumption, we maximize the total expected demand which is covered. To solve such a stochastic nonlinear model efficiently, simulation and genetic algorithm are integrated to produce a hybrid intelligent algorithm. Finally, some numerical examples are presented to illustrate the effectiveness of the proposed algorithm.

© 2008 IUST Publication, All rights reserved. Vol. 19, No. 4

1. Introduction

This paper examines a class of maximal covering location problem which has wide real-world application. The maximal covering location problem (M.C.L.P) was originally developed to determine a set of facility locations which would maximize the total customers' demand serviced by the facilities within a predetermined critical service criterion. Obviously, the model can be directly useful to most facility location planning such as warehouses, health-care centers, fire stations, recreation centers, emergency centers, etc. In addition, this model can be applied to data abstraction problems, to portfolio formation [1]. This model maximizes the number of demand points covered within a specified critical coverage or time by a fixed number of facilities. It does not need that all demand

points be covered. The M.C.L.P and its many developments compose a significant class of problems in location literature.

Brandeau and Chiu [2] presented a survey of representative problems that have been studied in location research. They recognized more than 50 problem types and indicated how those problem types relate to one another. Galvao *et al.* [3] evaluated the performance of two heuristic methods named LaGrange an and surrogate relaxations to solve maximal covering location problem. The first method had the integrality property and the surrogate relaxed problem they resolved was the linear programming relaxation of the original 0–1 knapsack problem. The performances of these two methods were compared with 331 test problems in the literature.

The error of these heuristics were very low and did not differ significantly among themselves. Galvão and ReVelle [4] proposed a LaGrange an heuristic for the maximal covering location planning. In this algorithm, the upper bound was specified by a vertex addition and

*Corresponding author: K. Shahanaghi

E-mail: shahanaghi@iust.ac.ir E-mail: Ghezavati@iust.ac.ir

Paper first received Feb. 22, 2007, and in revised form Jan. 24, 2008.

another heuristic and lower bounds were generated through a sub gradient optimization method. This algorithm was validated by solving 150 instances. For large sized problems a duality gap was computed at the end of the algorithm. Church and ReVelle [5] were the first who introduced the MCLP. In their model, the aim was to maximize the population covered within a desired service distance by optimally locating a fixed number of facilities.

ReVelle *et al.* [6] used some meta heuristics methods to solve large scale problems in maximal covering area. They believed that exact methods may be too unwieldy for real-world applications, and so heuristics can allocate faster solution times with sub-optimal results. In their instances exact algorithms are provided by linear programming and branch and bound. Karasakal *et al.* [7] developed maximal covering location problem as a partial coverage so that coverage can change from full covering to no covering. They also, introduced solution procedure based on LaGrange an relaxation method and compared it with two classical approaches. Richard L. Church *et al.* [8] developed a notation of a model based on the premise that reserve selection or site prioritization can be defined as a classical covering problem usually applied in many location problems.

Specially, they developed a structure of the maximal covering location model to classify sets of sites which represent the maximum possible representation of specific species. Boffey and Narula [9] survey the multi objective covering and routing problems. Karasakal [10] formulated the MCLP in the presence of partial coverage, and developed a solution procedure based on LaGrange an relaxation and showed the result of the approach on the optimal solution by comparing it with the classical methods. Yupo Chan and M. Mahan [11] suggested a variant of the maximal covering location problem to locate up to p signal-receiving stations.

In their study the “demands,” called geolocations, to be covered by these stations are distress signals and/or transmissions from any targets. Araz *et al.*, [12] proposed a multi-objective MCLP based on an emergency vehicle location. They considered the maximization of the covered customers and minimization of the total travel distance from the emergency services. Younies and George [13] introduced a zero–one mixed integer formulation for a maximal covering problem where points were covered by inclined parallelograms in a plane. A version of maximal expected covering location problem was proposed by Daskin [14] as an extension of the maximal covering location problem (MCLP) formulated by Church and ReVelle [5], to account for the possibility of server unavailability due to a congested system.

This model relates the problem of optimally locating servers so as to maximize the expected coverage of demand, considering the possibility of server

unavailability when a call is received by the server. When formulating this problem the author makes 3 simplifying assumptions: servers operate independently, each server has the same busy probability and server busy probabilities are invariant with respect to their location. Luis Gonzalo Espejo *et al.* [15] proposed a 2-level hierarchical development of the MCLP and also a combined Lagrangean–surrogate (L–S) relaxation to solve the model. Their results were compared with exact results obtained using CPLEX software.

Berman and Krass [16] considered a generalization of the maximal cover location problem which allowed for partial coverage of customers, with the degree of coverage being a non-increasing step function of the distance to the nearest facility. John R. Current and David A. Schilling [17] proposed two bi-criterion routing problems including the median tour problem and the maximal covering tour problem. In both problems the tour must select only p of the n facility on the network. Indeed, both problems have as one of their objectives the minimization of total travel distance. The second objective in both problems is to maximize access to the tour for the facilities not directly on it.

The rest of this paper is as follows: Section 2 presents a new mathematical model for the given problem. Section 3 proposes the solution procedure based on simulation and genetic algorithms. The computational results are illustrated and discussed in Section 4. Finally, the remarking conclusion is given in Section 5.

2. Model Formulation

In this section, we describe in more detail the probability maximal location problem in which we are interested. There is a set of demand points N , at which requests are generated, and a set of locations M , where facilities may be opened. We assume that the requests at a demand point $i \in D$ are generated independent of the processes at other demand points in N .

We develop a formulation based on the classical p -median formulation. However, instead of minimizing the total distance, this model maximizes the expected coverage of the demand points by determining one of the selected facility sites, which ensures maximum coverage confidence for each demand point.

2-1 Parameters:

d_i : The demand for customer i .

q_j : Coverage time for DC j .

f_{ij} : Random variable denoting traveling time between customer i and DC j which has normal distribution function with the mean m_{ij} and variance S_{ij}^2 .

Cap_j : Maximum capacity for DC j .

P: Total number of DCs to be located.

2-2. Decision Variables:

$$X_{ij} = \begin{cases} 1 & \text{If demand of customer is satisfied by DC } j. \\ 0 & \text{Otherwise.} \end{cases}$$

$$U_j = \begin{cases} 1 & \text{If a DC is located at site } j. \\ 0 & \text{Otherwise.} \end{cases}$$

p_i : Total probability of coverage for customer i.

p_{ij} : Probability of coverage customer i by Dc j.

A notable point in this model is that all distribution centers can not service all customers because each DC has special coverage time and if a customer isn't in this coverage, the DC can not service that customer. In this model we consider this parameter that leads to the model that will be more realistic.

We assume that this traveling time is probability and then "covering" or "not covering" a customer by a DC will be probabilistic. So, to compute the objective function, we measure expected value for it. (Amount of demand × Probability of coverage).

So, this constraint must be added to compute probability of coverage for each customer per each DC:

$$P_{ij} - \Pr(f_{ij} \leq q_j) = 0 \Rightarrow P_{ij} - \left(\int_0^{q_j} \frac{1}{s_{ij} \sqrt{2p}} e^{-\frac{x-m_{ij}}{s_{ij}^2}} dx \right) = 0 \quad (1)$$

$$P_i - \sum_j P_{ij} \cdot X_{ij} = 0 \Rightarrow P_i = \sum_j P_{ij} \cdot X_{ij} \quad (2)$$

2-3. Model Description

$$MaxZ = \sum_i \sum_j d_i \times P_i \quad (3)$$

Subject to:

$$X_{ij} \leq U_j \quad \forall i \in M, \forall j \in N \quad (4)$$

$$\sum_j X_{ij} \leq 1 \quad \forall i \in M \quad (5)$$

$$\sum_i X_{ij} \cdot d_i \leq cap_j \quad \forall j \in N \quad (6)$$

$$\sum_j U_j = p \quad (7)$$

$$P_{ij} - \left(\int_0^{q_j} \frac{1}{s_{ij} \sqrt{2p}} e^{-\frac{x-m_{ij}}{s_{ij}^2}} dx \right) = 0 \quad \forall i \in M \quad (8)$$

$$P_i - \sum_j P_{ij} \cdot X_{ij} = 0 \quad \forall i \in M \quad (9)$$

$$X_{ij} = (0,1), U_j = (0,1), P_i \geq 0, P_{ij} \geq 0 \quad (10)$$

The objective function maximizes total expected demands which are covered. Set constrain (4) says that a customer can allocate to DC j when that DC is opened. Set constrain (5) ensures that a customer cannot be allocated to more than one DC. Set constrain (6) forces that total demand assigned to DC j must be less than its capacity. Set constrain (7) guarantees that total number of DCs to be located should be equal to p. Set constrain (8) computes probability of coverage customer i per each DC (Based on normal distribution). Set constraint (9) shows total probability of coverage for each customer by all DCs. Set constraints (10) determines type of variables.

3. Solution Procedure

Because this modeling is nonlinear and software such lingo gives us local optimum solution and capacitated location problems are NP-hard, we use hybrid meta heuristics to solve this problem. So, in order to solve general stochastic programming models for capacitated network design problem, we use GA and simulation to produce a hybrid algorithm. Contrasting the general GA, this algorithm will reduce the computation greatly, which makes it possible to deal with problems of quite large size by the algorithm.

3-1. Simulation Process

Because of the complexity of the proposed model, we design a simulation model to compute uncertain functions covering or not covering customers by distribution centers. First, we experiment the simulation process to generate a 0-1 matrix, namely Z_{ij} . Also, we define a lower bound for probability of coverage. If probability of coverage is greater than the lower bound, we assume that the customer can be covered by distribution center completely. For instance, if probability of coverage customer 5 by distribution center 8 is 0.78 and we define the lower bound as 0.70, we assume that $Z_{5,8} = 1$ and it means that Customer 5 can be covered by distribution center 8 [18].

$$Z_{ij} = \begin{cases} 1 & \text{if } Prob(f_{ij} \leq q_j) \geq r \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Due to the complexity, we design some stochastic simulations to calculate uncertain functions shown in Figure 1. (Uncertain function is covering or not covering customers by DCs).

After the mentioned simulation process, we can obtain Z_{ij} matrix. Output of this simulation is input for genetic algorithm. (Output of simulation is Z_{ij} matrix).

After simulation Z_{ij} can be defined and the model will be changed as follows:

$$Z_{ij} = \begin{cases} 1 & \text{If DC at site } j \text{ can cover customer } i. \\ 0 & \text{Otherwise.} \end{cases}$$

3-1-1. Model Linearization

$$\text{Max} Z = \sum_i \sum_j X_{ij} \cdot d_i \cdot Z_{ij} \quad (12)$$

Subject to:

$$X_{ij} \leq U_j \cdot Z_{ij} \quad \forall i \in M, \forall j \in N \quad (13)$$

$$\sum_j X_{ij} \leq 1 \quad \forall i \in M \quad (14)$$

$$\sum_i X_{ij} \cdot d_i \leq \text{cap}_j \quad \forall j \in N \quad (15)$$

$$\sum_j U_j = p \quad (16)$$

$$X_{ij} = (0,1), U_j = (0,1) \quad (17)$$

Set constraint (13) says that customer i can be allocated to DC j if that DC is opened and the customer i can be covered by DC j . The definitions of the other constraints have been said before.

3-2. Genetic Algorithm

GA is a stochastic search and heuristic optimization technique, which has been widely adopted by many researchers to solve various problems. This algorithm was first developed by Holland [19]. It mimics the mechanism of genetic evolution in the biological nature and consists of a population of chromosomes (strings or individuals) that are composed of genes. These genes represent a number of values, called alleles. Each chromosome (genotype) represents one potential solution (phenotype).

For $i=1$ to No. of customers
For $j=1$ to No. of DCs

Step 1) set $V_k = 0$

Step 2) Generate random numbers Z_{ij} by repeating the following procedure for 100 times:

i) Generate random number m_1, m_2 uniformly from $[0, 1]$.

ii) Let $j_1 = \sqrt{-2 \times \text{Ln}(m_1)} \text{Sin}(2\pi m_2)$

iii) Let $j_2 = j_1 \times S_{ij} + m_{ij}$

iv) If $j_2 \leq q_j$ then

$$V_k + 1 \rightarrow V_k$$

Step 3) if $V_k \geq 75$ then (In definition of Z_{ij} matrix we assume $r = 0.75$)

$$Z_{ij} = 1$$

Else

$$Z_{ij} = 0$$

End if

Next j

Next i

Fig. 1. Pseudo code of simulation process

The process of genetic operators (i.e., crossover and mutation) is carried out in the pool; after that, an evolution is completed by creating new chromosomes (offspring). This offspring is expected to be stronger than the parents, but this may not always be true.

3-1-2. Representation of Chromosome

In our GA-based approach, each chromosome or bit string (i.e., an example solution) consists of a matrix which is made of 0 and 1. The dimension of this matrix is equal to:

GA does not rely on analytical properties of the function to be optimized (Goldberg [20]). In short, GA has two major processes: 1) GA is iteratively and randomly generating new solutions; and 2) these solutions are checked for the optimality according to predefined fitness functions.

This becomes the most powerful principle of GA. It makes them widely suitable for finding an optimal solution in many complex problems, such as the traveling salesman problem (TSP) and any forms of scheduling problems. For more details figure 2 illustrated classical genetic algorithms.

The number of customers in the network \times number of DCs. The matrix represents the assignments of the customers to DCs and gets the value 1 in element $[ij]$ if

customer i is assigned to DC j . sample of chromosome structure has shown in figure 3.

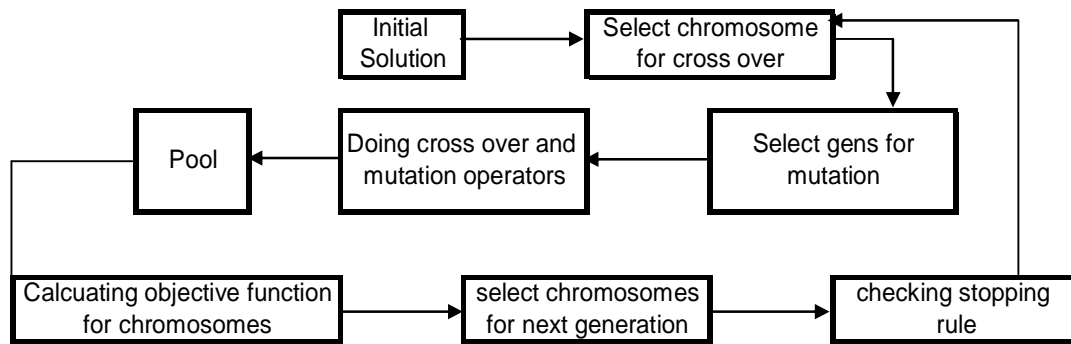


Fig. 2. Structure of genetic algorithms

		Number of DCs			
		1	2	3	4
Number of customers	1	0	1	0	0
	2	0	0	1	0
	3	0	0	0	0
	4	0	1	0	0
	5	0	0	1	0
	6	0	1	0	0

Fig. 3. Sample of chromosome structure

Step1. Initialize pop_size chromosomes $X^k = [x_{ij}^k]$ $k = 1, 2, \dots, Pop_Size$ from the potential region $\{(X) | g_i(x_i) \leq 0, x_i = 1, 2, \dots, n\}$ randomly.

$$[x_{ij}^k] = \begin{cases} 1 & \text{If DC at site } j \text{ covers customer } i \text{ in chromosome } k. \\ 0 & \text{Otherwise.} \end{cases}$$

At first p DCs which should be opened among all potential DCs randomly are selected and then a customer is selected randomly. In the second phase, for each customer the number of DCs which can cover it is determined. Then for this customer, a random DC from those DCs determined previously is selected randomly and the customer in chromosome k is allocated to it considering the capacity of DCs. (Because DCs have capacity, some customers cannot be allocated to DCs).

Step 2. Compute the fitness of all chromosomes $V_k, k = 1, 2, \dots, Pop_Size$. The rank-based evaluation function is defined as the objective function for chromosome k .

Step 3. It is very important to create new chromosomes (i.e., offspring) from the selected chromosomes (called parents) with the current population. This process is carried out by the use of genetic operators, namely crossover and mutation.

Renew the chromosomes $V_k, k = 1, 2, \dots, Pop_Size$ by crossover operation. In order to determine the parents for crossover operation, we repeat the following process from $k = 1$ to pop_size : generating a random real number r from the interval $[0, 1]$, the chromosome V_k will be selected as a parent provided that $r < Pc$; where the parameter Pc is the probability of crossover. Then we group the selected parents V'_1, V'_2, V'_3, \dots to the pairs, $(V'_1, V'_2), (V'_3, V'_4), \dots$ without loss of generality; let us illustrate the crossover operator on each pair by (V'_1, V'_2) . At first, we make a matrix with $i \times 2j$ dimension. ($i =$ No. of customers and $j =$ No. of DCs). In other words, we accrete parent matrixes and make one matrix.

Then in the new matrix, in each row, if number of X_{ij} with value 1 is more than one, select one of them randomly and set it to be 1 and set the other to be 0. By this, a customer is not allocated to different DCs. In the new matrix we have at most $2 \times p$ opened DCs. For

generating offspring we select p DC from opened DCs in the new matrix, randomly.

In fact we select p columns from the new matrix. (The note in this procedure is, if column r ($r \leq j$) is selected then column $r + j$ must not be selected and if column r ($r \geq j$) is selected then column $r - j$ must not be selected). By doing the mentioned procedure we can get offspring by crossover operator. An example of designed crossover is illustrated in figure 4.

Step 4. Update the chromosomes $V_k, k=1,2,\dots,Pop_Size$ by mutation operation. Similar to the proves of selecting parents for the crossover operation, we repeat the following steps from $k=1$ to Pop Size: generating a random real number r from the interval $[0, 1]$, then chromosome V_k will be selected as a parent provided that $r < Pm$; where the parameter Pm is the probability of mutation. For each selected parent: $X^k = [x_{ij}^k]$, we mutate it in the following way. Select randomly from opened DCs and name it j and then select randomly from closed DCs and name it j' . Then with below procedure close DC j and open DC j' . First for customers allocated to DC j , we set $X^k = [x_{ij}^k] = 0$. Then start to allocate unassigned customers to DC j' considering coverage radius and

capacity constraint, randomly. Example of designed mutation is illustrated in figure 5.

Step 5. Compute objective function for all chromosomes where the chromosomes $V_1, V_2, \dots, V_{Pop_Size}$ are assumed to have been rearranged from good to bad according to their objective values.

Step 6. Select the chromosomes for a new population. The selection process is based on selecting 50% from the best chromosomes and 50% randomly. Thus we obtain pop size copies of chromosomes, denoted also by V_k ;

Step 7. Repeat the second to seventh steps for a given number of cycles.

Step 8. Report the best chromosome $X^* = [x_{ij}^*]$ as the optimal solution.

A notable point in the mentioned GA's operators is to generate feasible offspring according to the capacity and the other constraints. So, in this process, it needn't to repair or reject offspring since they are generated feasible.

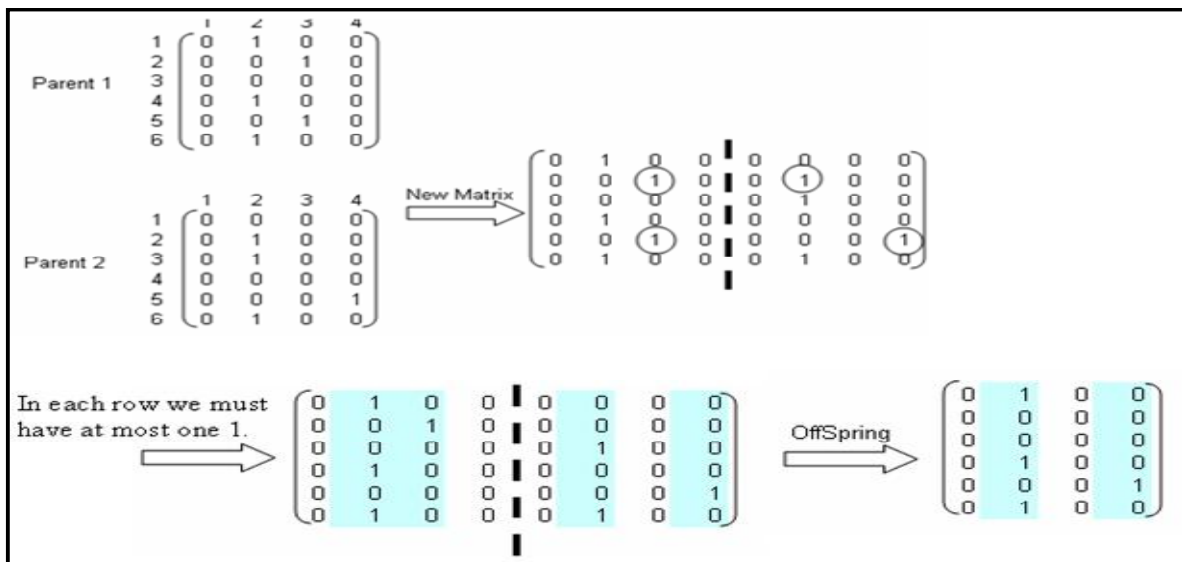


Fig. 4. Sample of crossover operator in GA

4. Computational Results

4-1. Validation Genetic Algorithm

In order to illustrate the effectiveness of the genetic algorithm, we give some random numerical examples that are performed on a personal computer. All algorithms considered in this study were coded in Visual Basic 6 and run on a Pentium IV PC with 1.5 GHz CPU and 256MB RAM. For this purpose we solve some examples and compare our solutions with global solutions obtained from lingo 8 software. In this

section, we just try to validate genetic algorithms alone without using simulation. In other words, in this section, genetic algorithm is applied to solve the second model and we assume that the Z_{ij} matrix is a deterministic parameter in order to measure just the effectiveness of the genetic algorithm. In the next section we will measure robustness of hybrid genetic. Consider a manufacturer decides to locate new distribution centers. Assume that there are 25 customers. Indeed, suppose that a decision maker

needs to select distribution centers from 6 potential distribution centers to cover customers. In this study, we consider 30 population sizes and 200 generations for each test problem solved by the proposed algorithm. Customers demand has been generated uniformly from interval [10, 20]. Also, DC capacity

has been generated uniformly from interval $[0.3 \sum_i d_i, 0.5 \sum_i d_i]$ where $\sum_i d_i$ is the total demand of all customers.

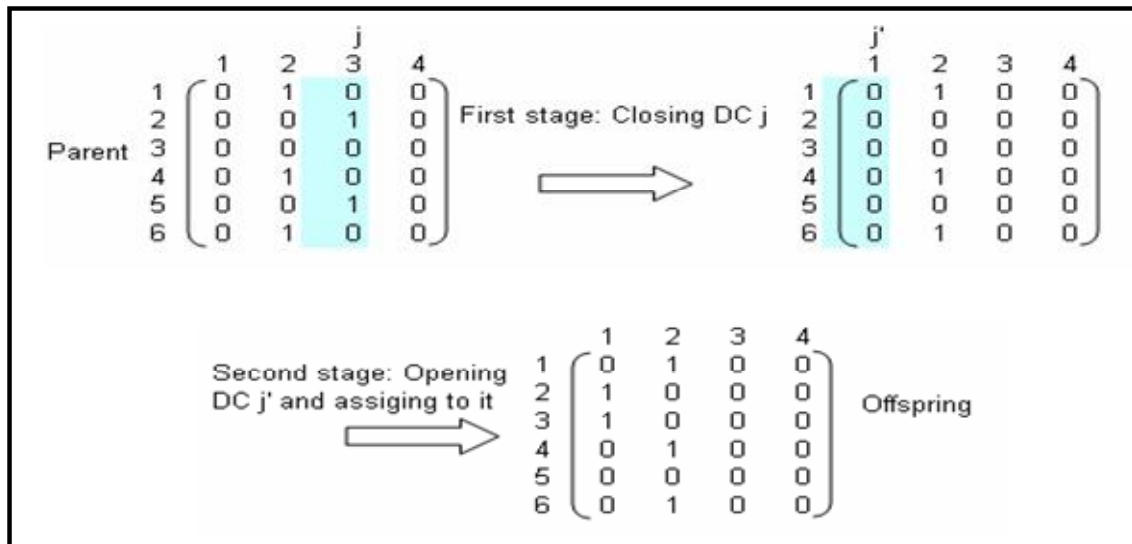


Fig. 5. Sample of mutation operator in GA

Tab. 1. Comparison between the GA and global solutions.

	No. of Customers	P	No. of DCs	Pop_Size	Pm	Pc	Iteration	No. of Opened DC	Objective Function	Global Objective Function	Error	CPU Time (Seconds)
1	12	3	4	30	0.95	0.55	200	2,3,4	160	160.00	0.00%	4
2	15	3	5	30	0.98	0.5	200	1,2,3	318	318.00	0.00%	5
3	18	3	5	30	0.95	0.45	200	1,3,4	465	465.00	0.00%	7
4	20	3	4	30	0.90	0.5	200	2,3,6	487	497.23	2.06%	7
5	22	2	5	30	0.95	0.5	200	2.4	310	315.58	1.77%	8
6	24	2	6	30	0.90	0.55	200	1.4	368	376.68	2.31%	8
7	25	3	5	30	0.96	0.45	200	2,3,4	476	486.47	2.15%	9

In Table 1, we compare solutions when different problems are taken with the same generations as a stopping rule. It appears that all the objective function and global optimum differ little from each other. In order to account for it, we present a parameter, called the percent error, i.e. (global optimum – objective function)/global optimum, where the global objective value is obtained from lingo 8 software. The last column named by ‘error’ in table 1 indicates this parameter. It follows from table 1 that the percent error does not exceed 2.31 % when different problems are selected, which implies that the genetic algorithm is effective enough to solve model.

4-2. Sensitivity Analysis for Simulation

To illustrate the effectiveness of the hybrid genetic algorithm, we solve a special problem for eight times

with numerous different runs for the simulation model. We show that this hybrid algorithm is also robust to the simulation settings. Also set parametering on GA and simulation was shown in table 2.

Consider a manufacturer who wants to locate new distribution centers. Assume that there are 45 customers. Suppose that a decision maker needs to select 5 distribution centers from 12, potential distribution centers to serve customers.

In Table 2, we compare solutions when different simulation runs are taken with the same generations as a stopping rule.

It appears that all the maximal covered demand differ little from each other. In order to account for it, we present a parameter, called the percent error, i.e. (the best objective value - objective value)/the best objective value, where the best objective value is the

maximum of all the ten maximal covered demand obtained above.

The last column named by 'error' in Table 2 is just this parameter. It follows from Table 2 that the percent error does not exceed 2.18 % when different simulations are selected, which implies that the hybrid

genetic algorithm is robust to the simulation runs. In other words, differences in simulation runs don't have any major impact on the final solution. Thus, the hybrid method is robust to the simulation settings and therefore is also effective to solve the model.

Tab. 2. Measuring the robustness of the HGA to the simulation run

	No. of Customers	P	No. of DCs	Pop_Size	Iteration	No. of Opened DC	Objective Function	The Best Objective	Error
1	45	5	12	25	300	2,4,8,10	725	735	1.36%
2	45	5	12	25	300	1,3,8,10	731	735	0.54%
3	45	5	12	25	300	2,3,4,8	719	735	2.18%
4	45	5	12	25	300	2,4,8,11	724	735	1.50%
5	45	5	12	25	300	4,6,8,9	732	735	0.41%
6	45	5	12	25	300	2,3,4,10	720	735	2.04%
7	45	5	12	25	300	2,3,6,8	735	735	0.00%
8	45	5	12	25	300	4,7,8,10	728	735	0.95%
9	45	5	12	25	300	2,4,8,10	721	735	1.90%
10	45	5	12	25	300	1,2,4,8	732	735	0.41%

5. Conclusion

In this paper, we defined notation of location-allocation problem while we have stochastic traveling time between service centers and customers this time is stochastic with normal distribution function. In the proposed model, we assumed that service centers have coverage radius restriction.

We formulated the problem as a non-linear integer programming. The advantages of the proposed study are as follows: considering probabilistic coverage radius which yields more flexibility for the model, presenting hybrid method including simulation for realizing uncertainty and genetic algorithm to solve the model. Performance of the solution procedure was verified by randomly generated test problems. These experiments show that the proposed algorithm is efficient to generate high quality solutions in a short period of time. We suggest some new areas to develop the mentioned model:

By using this contribution in the integrated supply chain systems, it assumes that such inventory and transportation can be a suitable development as future research for the presented model. Also, considering service level constraint to cover each customer can be an interesting future studies. Because in this model we had traveling time between service centers and customers, considering the time window constraint for customers to get service is another suitable development.

References

- [1] Chen-Cua Chung: *Recent Application of the Maximal Covering Location Planning Model*, Journal of operations research society. Vol.37, No.8, 1986, pp. 735-746.
- [2] Brandeau, M.L., Chui, S.S., *An Overview of Representative Problems in Location Research*. Management Science. 35, 1986, pp. 645-74.
- [3] Roberto, D., Galvão, Luis Gonzalo Acosta Espejo and Brian Boffey, A comparison of Lagrangean and surrogate relaxations for the maximal covering location problem, European Journal of Operational Research Vol. 124, Issue 2, 16 July 2000, pp. 377-389
- [4] Roberto Diéguez Galvão, Charles ReVelle, *A Lagrangean Heuristic for the Maximal Covering Location Problem*: European Journal of Operational Research Vol. 88, 1996, PP. 114-123, Location Science, Volume 5, Issue 1, May 1997, P. 72.
- [5] Church, R.L., ReVelle, C.S., *The Maximal Covering Location Problem*. Papers of the Regional Science Association; 32:101-18, 1974.
- [6] Charles, ReVelle, Michelle Scholssberg, Justin, *Solving the Maximal Covering Location Problem With Heuristic Concentration*, Computers & Operations Research, Vol. 35, Issue 2, Feb. 2008, pp. 427-435.
- [7] Orhan Karasakal, Esra, K., Karasakal, *A Maximal Covering Location Model in the Presence of Partial Coverage*, Computers & Operations Research, Vol. 31, Issue 9, August 2004, pp. 1515-1526.
- [8] Richard, L., Church, David, M., Stoms, Frank, W., Davis: *Reserve Selection as a Maximal Covering Location Problem*, Biological Conservation, Vol. 76, Issue 2, 1996, pp. 105-112.
- [9] Boffey, B., Narula, S.C., *Multi Objective Covering and Routing Problems*. In: Karwan MH, Spronk J, Wallenius J. editors. Essays in decision making: a volume in honour of Stanley Zionts. Berlin: Springer, 1997.

- [10] Karasakal, O., Esra, K., *A Maximal Covering Location Model in the Presence of Partial Coverage*. Computers & Operations Research. 31, 2004, pp.1515-1526.
- [11] Chan, Y., Mahan, J.M., Chrissis, J.W., Drake, D.A, Wang, D., *Hierarchical Maximal-Coverage Location-Allocation: Case of Generalized Search-and-Rescue*. Computers & Operations Research. 35, 2008, 1886-1904.
- [12] Araz, C., Selim, H., Ozkarahan, I., *A Fuzzy Multi-Objective Covering-Based Vehicle Location Model for Emergency Services*. Computers & Operations Research. 34, 2007, 705-726.
- [13] Younies, H., Wesolowsky, G.O., *A Mixed Integer Formulation for Maximal Covering by Inclined Parallelograms*. Eur. J. of Operational Research, 159, 2004, pp. 83-94.
- [14] Daskin, M.S., *A Maximal Expected Covering Location Model: Formulation, Properties and Heuristic Solution*. Transportation Science, 17, 1983, pp. 48-70.
- [15] Luis Gonzalo Acosta Espejo, Roberto, D., Galvão, Brian Boffey, *Dual-Based Heuristics for a Hierarchical Covering Location Problem- Computers & Operations Research*, Vol. 30, Issue 1, Jan. 2003, pp. 87-9.
- [16] Berman, O., Krass, D., *The Generalized Maximal Covering Location Problem*. Computers & Operations Research, 29, 2002, pp. 563-581.
- [17] John, R., Current, David, A., *Schilling: The Median Tour and Maximal Covering Tour Problems: Formulations and Heuristics*, European Journal of Operational Research, Vol. 73, Issue 1, 24 Feb.1994, pp. 114-126.
- [18] Hwang, H.S., *Design of Supply Chain Logistic System Considering Service Level*. Computers & Industrial Engineering. 43, 2002, pp. 283-297.
- [19] Holland, J.H., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, 2nd Ed., MIT Press, Cambridge, 1992.
- [20] Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wiley, MA, 1989.