Development of Threshold Algorithms
for a Location Problem with Elastic Demand

Tatyana Levanova[1,2(✉)] and Alexander Gnusarev[1]

[1] Omsk Branch, Sobolev Institute of Mathematics,
Pevtsova str. 13, 644043 Omsk, Russia
alexander.gnussarev@gmail.com
[2] Dostoevsky Omsk State University, Prospekt Mira 55A, 644077 Omsk, Russia
levanova@ofim.oscsbras.ru

Abstract. This work is devoted to development of threshold algorithms for one static probabilistic competitive facility location and design problem in the following formulation. New Company plans to enter the market and to locate new facilities with different design scenarios. Clients of each point choose to use the facilities of Company or its competitors depending on their attractiveness and distance. The aim of the new Company is to capture the greatest number of customers thus serving the largest share of the demand. This share for the Company is elastic and depends on clients' decisions. We offer three types of threshold algorithms: Simulated annealing, Threshold improvement and Iterative improvement. Experimental tuning of parameters of algorithms was carried out. A comparative analysis of the algorithms, depending on the nature and value of the threshold on special test examples up to 300 locations is carried out. The results of numerical experiments are discussed.

Keywords: Discrete optimization · Integer programming
Location problems · Threshold algorithms · Simulated annealing

1   Introduction

Nowadays a lot of economic situations are described by the mathematical model of discrete optimal location discrete problems. The situations, where the decision is made by the monopolist only, have been mostly studied. Such situations are common for the normal economic behaviour. Thus there appear the problems aimed to minimise the expenses such as: simple plant location problem [7], p-median problem [13], capacitated plant location problem [16], which have already become classical. In the modern economic situation, it is often demanded to take into consideration the current rivalry at the market. From this point of view there is a great interest to the competitive facility location models, as they describe most complicated situations and require the special methods of solution. The models differ particularly in the competitors' behaviour [10].

The static competitive facility location problems imply that the competitor's decision is known and will not be changed. Let us take into consideration the situation when a Company plans to enter the market of existing products and services, so that its p shops served as big share of the market as it is possible. The facilities produce goods of the same kind; the prices are similar; the customers' choice of the shops depends only on the distance from the located ones [14]. If the customers take into account the size of the shop, its assortment, the quality of the service, then such circumstances are described by the static probabilistic models. The different kinds of which are described e.g. in [3,6,8].

Even more complicated is the situation when the competitor's decision is not known beforehand. The deterministic multilevel sequential problems can be considered among such formulations. The bi-level models distinguish the Leader (the company) and the Follower (the competitor). At first the Leader enters the market and establishes a set of facilities $S1$ which consists of p facilities. Then the Follower, being aware of that decision, establishes the set of facilities $S2$ which consists of r facilities. Each customer chooses the place to be served among the facilities from the sets $S1$ or $S2$ according to his/her own preferences. Each served customer brings definite profit, therefore all the market is divided between the Leader and the Follower. Those problems are called $(r|p)$-centroid problems. The interesting results for that kind of problem have been gained e.g. in [4,9,15].

Besides the probabilistic bi-level sequential problems, competitive facility location with competition of customers and others should be taken into consideration. Some solutions for them can be found in [5,10].

This paper deals with the static competitive facility location problems with the elastic demand. The three types of threshold algorithms [1] have been developed for that task. The computational experiments have been carried out in order to compare them.

The comparative analysis of the algorithms, according to the nature and the value of the threshold, has been carried out for the special test examples with up to 300 locations. The results of the computational experiments are given below.

2   Problem Formulation

Berman and Krass [6] and Aboolian et al. [3] develop a spatial interaction model with variable expenditure by introducing non-constant expenditure functions into spatial interaction location models. Aboolian et al. [2] proposed a new model where optimal location and design decisions for a set of new facilities are seeking. Here we develop approximate algorithms for the location and design problem described in [2]. In this problem, Company plans to locate its facilities which differ from one another in design: size, range, etc. Let R be the set of facility designs, $r \in R$. There are w_i customers at the point i of discrete set $N = \{1, 2, \ldots, n\}$. All customers have the same demand, so each item can be considered as one client with weight w_i. Clients of each point choose to use the facilities of Company or its competitors depending on their attractiveness and

distance. The distance d_{ij} between the points i and j is measured, for example, in Euclidean metric or equals to the shortest distance in the corresponding graph. It is assumed that $C \subset P$ is the set of pre-existing competitor facilities and it will not change. The Company may open its markets in $S = P \setminus C$ taking into account the budget B and the cost of opening c_{jr} facility $j \in S$ with design $r \in R$. The Company's goal is to attract a maximum number of customers, i.e. to serve the largest share of total demand.

Let us write out the mathematical model according to [2]. Variables $x_{jr} = 1$, if facility j is opened with design variant r and $x_{jr} = 0$ otherwise, $j \in S, r \in R$. Utility $u_{ij} = \sum_{r=1}^{R} k_{ijr}x_{jr}$, where the supplementary coefficients

$$k_{ijr} = a_{jr}(d_{ij} + 1)^{-\beta}.$$

They depend on the sensitivity β of customers to distance to facility and attractiveness a_{jr}. The total utility for the customers in point $i \in N$ from the facilities controlled by the competitors is $U_i(C) = \sum_{j \in C} u_{ij}$.

The demand function has an exponential form:

$$g(U_i) = 1 - \exp\left(-\lambda_i U_i\right),$$

where λ_i is the characteristic of elastic demand in point i, $\lambda_i > 0$; U_i is the total utility for a customer at $i \in N$ from all open facilities:

$$U_i = \sum_{j \in S}\sum_{r=1}^{R} k_{ijr}x_{jr} + U_i(C) = U_i(S) + U_i(C).$$

It is assumed that the demand of the customers at $g(U_i)$ is a concave increasing function.

The company's total share of facility $i \in N$ is measured by:

$$MS_i = \frac{U_i(S)}{U_i(S) + U_i(C)} = \frac{\sum_{j \in S}\sum_{r=1}^{R} k_{ijr}x_{jr}}{\sum_{j \in S}\sum_{r=1}^{R} k_{ijr}x_{jr} + \sum_{j \in C} u_{ij}}.$$

Then the mathematical model looks like:

$$\max \sum_{i \in N} w_i \cdot g(U_i) \cdot MS_i, \tag{1}$$

$$\sum_{j \in S}\sum_{r \in R} c_{jr}x_{jr} \leq B, \tag{2}$$

$$\sum_{r \in R} x_{jr} \leq 1, j \in S, \tag{3}$$

$$x_{jr} \in \{0,1\}, \quad r \in R, j \in S. \tag{4}$$

Based on above notation, the objective function (1) looks as follows:

$$\max \sum_{i \in N} w_i \cdot \left(1 - \exp\left(-\lambda_i \left(\sum_{j \in S} \sum_{r=1}^{R} k_{ijr} x_{jr} + U_i(C) \right) \right) \right)$$

$$\cdot \left(\frac{\sum_{j \in S} \sum_{r=1}^{R} k_{ijr} x_{jr}}{\sum_{j \in S} \sum_{r=1}^{R} k_{ijr} x_{jr} + \sum_{j \in C} u_{ij}} \right). \tag{5}$$

The objective function (5) reflects the Company's goal to maximize the share of customers demand. Inequality (2) takes into account the available budget. Condition (3) shows that only one variant of the design can be selected.

3   Threshold Algorithms

The number of algorithms known for solution of the competitive facility and design problem is little. In [2], an adapted weighted greedy heuristic algorithm is proposed. Earlier we developed Variable Neighborhood Search algorithms [12]. In this paper, we offer Threshold Algorithms [1] for this problem.

Let s be a feasible solution of a combinatorial minimization problem. The general scheme of threshold algorithms is the following:

1. Select an initial solution s_0, compute the initial value of the objective function $f(s_0)$, define the record value as $f^* := f(s_0)$. Set the iteration counter $k = 0$, set the threshold value t_k and the type of neighborhood $N(s_k)$.
2. Until the stopping criterion is not satisfied, do the following:
   2.1 Select the new solution randomly in the neighborhood of the current one: $s_j \in N(s_k)$.
   2.2 If the difference does not exceed the threshold $f(s_j) - f(s_k) < t_k$, then $s_{(k+1)} := s_j$.
   2.3 If $f^* > f(s_k)$, then update the best found solution value $f^* := f(s_k)$.
   2.4 Set $k := k + 1$.

There are three variants of this algorithm depending on the setting of the sequence of the threshold value $\{t_k\}$:

(1) *Iterative improvement*: the sequence $t_k = 0, k = 0, 1, 2, ...$, is a variant of local descent with a monotonous improvement of the objective function;
(2) *Threshold accepting*: $t_k = c_k, k = 0, 1, 2, ..., c_k \geq 0, c_k \geq c_{k+1}$ and $\lim_{k \to \infty} c_k \to 0$ are the local search variants when the objective function deterioration is assumed until some fixed threshold, and the threshold continually goes down to zero;
(3) *Simulated annealing*: the sequence $t_k \geq 0, k = 0, 1, 2, ...$, is a random variable with the expectation $E(t_k) = c_k \geq 0$ which is the local search variant, when the objective function arbitrary deterioration is assumed but the transition

probability is inversely related to the deterioration value. For any two feasible solutions s_i and s_j the probability of accepting s_j from s_i at the iteration k is:

$$P_{s_i s_j} = \begin{cases} 1 & if \quad f(s_j) \le f(s_i), \\ \exp \frac{f(s_i) - f(s_j)}{c_k} & if \quad f(s_j) > f(s_i), \end{cases} \quad s_j \in N(s_i).$$

At the present time the simulated annealing algorithm shows good results for a wide range of optimization problems. The randomized nature of this algorithm allows asymptotic convergence to optimal solutions under special conditions [1].

We adapted threshold algorithms for the maximization problem. These algorithms belong to the class of local search methods. The neighborhood selection plays an important role in their development for individual tasks. The Lin-Kernighan neighborhood have been used in the proposed variants of algorithms. In addition, a special neighborhood is constructed as follows. Let the vector $z = (z_i)$ be such that z_i corresponds to facility i: $z_i = r$ iff $x_{ir} = 1$. Then feasible solution z' is called neighboring for z if it can be obtained with the following moves: (a) choose one of the open facilities p with design variant z_p and reduce the number of design variant up to 0 (close the facility); (b) select the facility q which is closed; then open the facility q with the design variant z_p.

4 Computatinal Experiments

To study the algorithms a series of testing instances similar to the real data of the applied problem [2] has been constructed in [12]. The testing instances consist of two sets with uniform distribution of distances in the interval [0;30] (Series 1) and with Euclidean distances (Series 2). They contain 96 instances for location of 60, 80, 100, 150, 200 and 300 facilities; 3 types of design variants are used, the budget limited is 3, 5, 7 and 9; the demand parameter is $\lambda_i = 1, i \in N$; the customer sensitivity to the distance is high ($\beta = 2$).

It must be mentioned that there is a problem of the choice of the parameter values so that the algorithm could produce good results for a wide range of instances. After the series of preliminary experiments the following parameter values for the simulated annealing algorithm have been found: the temperature interval length $l = 10$, the initial temperature $t_0 = 150$, the cooling (minimal) temperature value $t_{cool} = 5$, the cooling coefficient $r = 0.99$, the number of points in the Lin-Kernighan neighborhood $K = 3$. The threshold value equal to 5 was chosen for the threshold accepting algorithm.

Table 1 shows the values of deviations from the upper bounds (UB) [12] for test instances with uniform distribution of distances in a single run of the algorithms. For example, the average deviations for the uniform distribution of distances instances of 300 locations are: 6.627% for the iterative improvement; 8.308% for the threshold accepting; 1.523% for the simulated annealing algorithm. The test instances with Euclidean distances proved to be difficult for all considered algorithms. The deviations in this case are: 15.325% for the iterative improvement; 18.296% for the threshold accepting; 14.231% for the simulated

Table 1. Deviations from upper bounds in case of uniform distribution of distances

N	Iterative improvement			Threshold accepting			Simulated annealing		
	min	max	av	min	max	av	min	max	av
60	0.010	5.196	1.562	0.066	11.043	3.511	0.008	4.727	0.818
80	0.019	5.698	1.017	0.027	9.030	3.452	0.014	1.810	0.342
100	0.000	6.596	1.043	0.050	22.014	3.737	0.000	3.463	0.648
150	0.000	12.279	3.434	0.013	13.492	6.638	0.000	4.804	0.668
200	0.000	10.176	3.402	0.000	24.627	6.796	0.000	9.589	1.796
300	0.000	19.050	6.627	0.000	21.510	8.308	0.000	11.340	1.523

annealing algorithm. In 8 test instances the best found goal function values of simulated annealing was within 0.001% from UB (in 6 and in 2 test instances for iterative improvement and threshold accepting respectively). It was noted that in some instances, vector solutions of the algorithms coincide with the vectors of UB: in 42 cases for iterative improvement, in 23 test instances for threshold accepting and in 75 cases for simulated annealing algorithm. For instances with Euclidean distances this values are 23, 11 and 27 respectively.

Since algorithms are of a probabilistic nature, they are tested repeatedly. We ran 1000 each of the algorithms for each of the test instances. For this computational experiment, the following results are obtained. The average deviations for the instances with uniform distribution of distances of 300 locations are: 6.771% for the iterative improvement; 8.36% for the threshold accepting algorithm; 1.044% for the simulated annealing algorithm. The deviations in case of the Euclidean distances are: 16.017% for the iterative improvement; 18.71% for the threshold accepting algorithm; 14.734% for the simulated annealing algorithm. This indicates that Series 2 is more complicated for the proposed algorithms. On the other hand, such deviations may display that the upper bounds for the second series are inaccurate. For Series 1, the 95% confidence interval for the probability of obtaining the deviations less than 0.001% for simulated annealing is between [0.111;0.115], for iterative improvement is [0.044;0.047] and for threshold accepting is [0.030;0.032]. For Series 2, the 95% confidence interval for the probability of obtaining the deviations less than 12% for simulated annealing is [0.112;0.116], for iterative improvement is [0.075;0.078] and for threshold accepting is [0.030;0.032]. The Wilcoxon test [11] showed statistically significant differences between the values of the objective functions of the investigated algorithms with significance level 0.05. The simulated annealing algorithm in both series of test instances gives the best value of the objective function. At the same time, an iterative improvement algorithm has an advantage over the threshold accepting algorithm, which is confirmed by statistically significant differences in both series.

Table 2 contains the information about minimal (min), average (av) and maximal (max) CPU time (in seconds) for the proposed algorithms until the stopping criterion was met. The experiments have been carried out using a PC Intel

Table 2. CPU time (sec)

N	Iterative improvement			Threshold accepting			Simulated annealing		
	min	av	max	min	av	max	min	av	max
60	0.085	0.109	0.142	0.078	0.104	0.123	0.905	1.190	1.462
80	0.132	0.161	0.203	0.130	0.162	0.226	1.544	1.866	2.281
100	0.177	0.210	0.235	0.176	0.217	0.260	2.228	2.228	2.853
150	0.332	0.411	0.504	0.342	0.416	0.523	4.120	4.876	5.783
200	0.525	0.603	0.660	0.554	0.618	0.668	6.367	7.213	7.824
300	1.242	1.330	1.581	1.227	1.325	1.529	15.197	16.176	18.004

i5-2450M, 2.50 GHz, memory 4 GB. The time for the instances with Euclidean distances and with the uniform distribution of distances is approximately the same. Note that well-known commercial software is rather time-consuming. For instance, for one of the instances of 60 locations the CPU time of CoinBonmin (GAMS) was 63 h and the objective function deviation from the upper bounds was 12.919%. All proposed algorithms yielded equal objective function values with deviation 11.796%, running time was less than 1.638 s for it.

5   Conclusion

This paper is devoted to the development of approximate algorithms for one variant of the static probabilistic competitive facility location and design problem. Its mathematical model is based on a nonlinear objective function, and the share of the served demand is elastic. That complicates the task of finding an optimal solution.

The threshold algorithms for the search of approximate solutions have been built, their parameter setting has been carried out. It should be noticed that the iterative improvement and the simulated annealing algorithms are comparable in the objective function for the instances up to 100 locations. The maximum counting duration of the built algorithms does not exceed 20 s. and the minimal deviations from upper bounds was less then 0.001%. The analysis of the algorithms for the instances with a large number of locations has proved the advantage of the simulated annealing algorithm over the threshold algorithms and its applicability to complex problems.

References

1. Aarts, E., Korst, J., Laarhoven, P.: Simulated annealing. In: Aarts, E., Lenstra, J.K. (eds.) Local Search in Combinatorial Optimization, pp. 91–120. Wiley, New York (1997)
2. Aboolian, R., Berman, O., Krass, D.: Competitive facility location and design problem. Eur. J. Oper. Res. **182**(1), 40–62 (2007)

3. Aboolian, R., Berman, O., Krass, D.: Competitive facility location model with concave demand. Eur. J. Oper. Res. **181**, 598–619 (2007)
4. Alekseeva, E., Kochetova, N., Kochetov, Y., Plyasunov, A.: Heuristic and exact methods for the discrete $(r|p)$-centroid problem. In: Cowling, P., Merz, P. (eds.) EvoCOP 2010. LNCS, vol. 6022, pp. 11–22. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12139-5_2
5. Beresnev, V.L.: Upper bounds for objective function of discrete competitive facility location problems. J. Appl. Ind. Math. **3**(4), 3–24 (2009)
6. Berman, O., Krass, D.: Locating multiple competitive facilities: spatial interaction models with variable expenditures. Ann. Oper. Res. **111**, 197–225 (2002)
7. Cornuejols, G., Nemhauser, G.L., Wolsey, L.A.: The uncapacitated facility location problem discrete location problems. In: Mirchamdani, P.B., Franscis, R.L. (eds.) Discrete Location Problems, pp. 119–171. Wiley, New York (1990)
8. Drezner, T.: Competitive facility location in plane. In: Drezner, Z. (ed.) Facility Location. A Survey of Applications and Methods, pp. 285–300. Springer, Berlin (1995)
9. Davydov, I.A., Kochetov, Y.A., Mladenovic, N., Urosevic, D.: Fast metaheuristics for the discrete $r|p$-centroid problem. Autom. Remote Control **75**(4), 677–687 (2014)
10. Karakitsiou, A.: Modeling Discrete Competitive Facility Location. Springer, Heidelberg (2015)
11. Kerby, D.S.: The simple difference formula: an approach to teaching nonparametric correlation. Compr. Psychol. **3**(1) (2014). http://journals.sagepub.com/doi/full/10.2466/11.IT.3.1. Accessed 20 Dec 2017
12. Levanova, T., Gnusarev, A.: Variable neighborhood search approach for the location and design problem. In: Kochetov, Y., Khachay, M., Beresnev, V., Nurminski, E., Pardalos, P. (eds.) DOOR 2016. LNCS, vol. 9869, pp. 570–577. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44914-2_45
13. Mirchamdani, P.B.: The p-median problem and generalizations. In: Mirchamdani, P.B., Franscis, R.L. (eds.) Discrete Location Problems, pp. 55–118. Wiley, New York (1990)
14. Serra, D., ReVelle, C.: Competitive location in discrete space. In: Drezner, Z. (ed.) Facility Location. A Survey of Applications and Methods, pp. 367–386. Springer, Berlin (1995)
15. Spoerhose, J., Wirth, H.C.: (r, p)-centroid problems on paths and trees. Technical report No. 441, Institute of Computer Science, University of Wurzburg (2008)
16. Sridharan, R.: The capacitated plant location problem. Eur. J. Oper. Res. **87**, 203–213 (1995)