Solving Capacitated *P*-Median Problem by Hybrid *K*-Means Clustering and Fixed Neighborhood Search algorithm

Rashed Sahraeian
Assistant Professor of Industrial Engineering
Shahed University, Tehran, Iran

Payman Kaveh
Postgraduate of Industrial Engineering
Shahed University, Tehran, Iran

Abstract

Capacitated *P*-median problem (CPMP) is one of the popular discrete location problems. CPMP locates *P* facilities between the candidate points, in order to satisfy the customer demand. This problem is a NP-hard problem. In this paper, a new hybrid algorithm is proposed to solve CPMP. In proposed method, *K*-means clustering algorithm will find a proper solution for Fixed Neighborhood Search algorithm (FNS). Then, FNS algorithm improves the quality of obtained solutions for standard benchmark instances with facilities locations exchange and omit the unsuitable candidates' points. The Computational results show the efficiency proposed algorithm in regard of the quality of solution.

Keywords
Capacitated *P*-median problem, *K*-means clustering, FNS.

1. Introduction

Capacitated *P*-median Problem is a special model of Capacitated Location-Allocation Problem. In this problem, *P* facilities are located between the candidate points, in order to satisfy the customer demand, and of course in a manner that the sum of transportation distances between facilities and customers minimized. We should pay attention: the capacity constraint of each facility should be regarded. This problem has many applications in real world. Some of its applications are: topological design of computer communications networks [1], design of a distribution network, where a set of customers are to be supplied from supply points [2], political districting [3], sales force territories design [4]. Due to CPMP is a NP-hard problem [5], heuristics method is used to solve it. One of these methods is column generation approach [6]. In this method, in each algorithm repetition, a set covering location problem is solved .An other method is Hybrid Scatter Search and Path Relinking algorithm [7]. The reasoning considered to use such a method is when Path Relinking algorithm is used before Scatter Search method; Scatter Search algorithm starts to solve the problem with proper initial solution. Variable Neighborhood Search is also one of the best methods to solve the Capacitated *P*-median Problem [2]. In this method, for decreasing the solution time, naive lower bound and transshipment lower bound were used for the comparison of new solution and the recent best solution that had ever seen. Also, there is a close kinship between CPMP and several other combinatorial optimization problems. If there are no capacity constraints, the problem reduces to the classical *P*-median problem (PMP) [8]. If the set of medians is fixed, the problem reduces to the Generalized Assignment problem (GAP) [9].

In this paper, a new hybrid algorithm is proposed to solve a CPMP. Due to FNS is a local search algorithm and is not able to make the initial solution, k-means clustering algorithm is used to do this task. Then, in FNS algorithm, the quality of obtained solutions for standard benchmark instances is improved by proper facilities location exchange. Other Sections of this paper are organized as follows: section 2 introduces the mathematical model of the problem. Section 3 describes the suggested hybrid method. Section 4 presents the result of computations and their analysis, and section 5 offers the conclusions and future suggestions.

2. Mathematical Model

Suppose $N=\{1,\ldots,n\}$ and $J=\{1,\ldots,m\}$ show the numbers of customers and candidate points for locating facilities, respectively. c_{ij}, d_i, b_j, also show the distance between customer i th $(i \in N)$ and facility j th $(j \in J)$, the demand of customer i th and the available capacity of located facility in the place j, respectively. Also $c_{ii}=0$. The decision variables of problem can be defined as follows:

- $y_j = 1$, if a facility is located at place $j \in J$, otherwise 0.
- $x_{ij} = 1$, if customer i th allocates to the facility which is in location j th, otherwise 0.

So, the mathematical model of CPMP will be [7]:

$$Min \sum_{i \in N} \sum_{j \in J} c_{ij} x_{ij} \tag{1}$$

$$s.t. \quad \sum_{j \in J} x_{ij} = 1 \qquad \forall\ i \in N, \tag{2}$$

$$\sum_{i \in N} d_i \ x_{ij} \le b_j \ y_j \qquad \forall \quad j \in J, \tag{3}$$

$$\sum_{j \in J} y_j = p, \tag{4}$$

$$x_{ij} \in \{0,1\} \quad , \quad y_j = \{0,1\} \qquad \forall\ i \in N \quad , \ \forall\ j \in J, \tag{5}$$

In this model, in constraints (2), each customer will be assigned to exactly one facility. The constraints (3) ensure that the capacity of every located facility is not violated. Constraint (4) guarantees that the P facilities are located exactly. The constraints (5) are binary variables.

3. New Proposed Method

Due to solving the CPMP with exact methods is so time-consuming, so in this part, the new suggested heuristic method will be described to solve the CPMP, which is a combination of *K*-means clustering and FNS algorithms.

3.1 *K*-means clustering algorithm

K-means clustering algorithm [10] is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume *k* clusters) fixed a priori. The main idea is to define *k* centroids, one for each cluster. According to above discussion, the steps of *K*-means clustering algorithm will be described as follows (Figure 1):

1. Select *k* of centroids randomly as initial centroids of *k* clusters. (Initial solution).
2. Allocate each customer to the nearest centroids. *K* new clusters are created.
3. For each *K* of new created clusters, recalculate new centroids.
4. Repeat the steps 2 and 3, so much until new centroids for all clusters in each repetition get constant.

Figure 1: K-means Clustering algorithm Steps

It is noted that, values of *K* and *P* is same. *K*-means clustering algorithm has two shortcomings: first, the *k*-means algorithm does not necessarily find the optimal configuration, corresponding to the global objective function minimum. Second, the algorithm is also significantly sensitive to the initial randomly selected cluster centroids. So, reducing the effect of second shortcoming, *K*-means clustering algorithm will be run multiple times, using the different initial centroids. Due to the above algorithm, uses different initial centroids for clustering the customers, this criterion are used to compare the clusters and select best one.

$$Min\ Z = \sum_{i=1}^{n} \sum_{j=1}^{k} \left\| X_i^{(j)} - C_j \right\|^2 \tag{6}$$

At above function, $X_i^{(j)}$ shows the location of customer i which is allocated by center C_j to the cluster j. The above function minimizes the sum of squared distance between each customer and the centroid of cluster that customer belong to it. In the proposed hybrid algorithm, the *K*-means clustering algorithm is performed 20 runs with different initial centroids and selects best clustering of customers. Then, a Single Facility Location Problem (SFLP) is solved for each of *k* clusters and specifies best facility location between of candidate points. This solution is initial solution of FSN algorithm. It is noted; there is not capacity constraint for each of clusters in *K*-means algorithm.

3.2 Fixed neighborhood search algorithm (FNS)

In this section, the FNS algorithm steps will be described. Firstly, the $k^{'th}$ neighborhood of a solution in a FNS algorithm is defined as follow:
«The $k^{'th}$ neighborhood of a solution contains all the solution that differ from the current one in the location of exactly k' facilities. It means, k' facilities are removed from the current solution and k' new facilities are replaced» .

Therefore, steps FNS Algorithm are described as follows (Figure 2):

1. Identify k' and *maxitr* values. Set r=1.
2. Find an initial solution. Call it S.
3. Create the $k^{'th}$ neighborhood of the solution of S. Call it $N_k'(S)$.
4. Generate a point S' at random from the $K^{'th}$ neighborhood of S ($S' \in N_k'(S)$).
5. IF $F(S') < F(S)$, S=S', r=1 and go to step 3.
6. IF $F(S') > F(S)$, r=r+1. If r > *maxitr*, stop algorithm. Otherwise go to step4.

Figure 2: Steps of FNS algorithm

3.3 The Modified FNS algorithm

There are three main differences between modified FNS algorithm and FNS algorithm: first, in modified FNS algorithm, Unsuitable candidate points are omitted. Second, we use other stop condition to finish algorithm. Third, the new algorithm has a memory.

3.3.1Two new methods of omitting unsuitable candidate points

In this section, two new methods are proposed for omitting unsuitable candidate points. In the former method, first, k' facilities are removed from Current solution(S) randomly. Then, for each of P-K' remained facilities at the current solution, H nearest candidate points is specified. In next step, all of specified candidate points in previous step are omitted. In this method, the value of H will be specified empirically and by trial and error.

In the latter method, there is a rule that according to it, in the most countries, facilities are located in the center and inside area of country, and in the borders area, there are not much facilities. It is because; customers exist in one way of facilities. So they can't service many customers with short distances. So, using this reality, some candidate points are omitted. To do this, a rectangular area is plotted by using candidate points as follow.

$$\left(X^{\min}, X^{\max}\right) = \left(Min\left(X_j\right), Max\left(X_j\right)\right) \quad \forall\, j \in J. \tag{7}$$

$$\left(Y^{\min}, Y^{\max}\right) = \left(Min\left(Y_j\right), Max\left(Y_j\right)\right) \quad \forall\, j \in J. \tag{8}$$

After plotting the rectangular area, for each of $Line\ i,\ i = 1,2,3,4.$, $N_i,\ i = 1,2,3,4,$ nearest candidate points are specified to $Line\ i$. In next step, these points are omitted. For example, in Figure 3, the black circles shows candidate points which were omitted by implementation of second method. Of course, in this method, it is supposed, the candidate points and customers locations are same. Also, the value of N_i will be specified empirically and by trial and error.

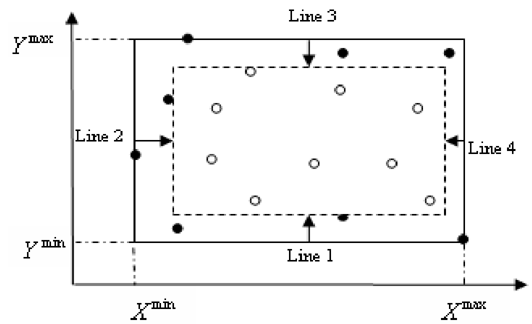

Figure 3: Omitting of candidate points

Now, omitting unsuitable candidate points in two proposed methods and using all remained candidate points, the k'^{th} neighborhood of current solution (S) is created.

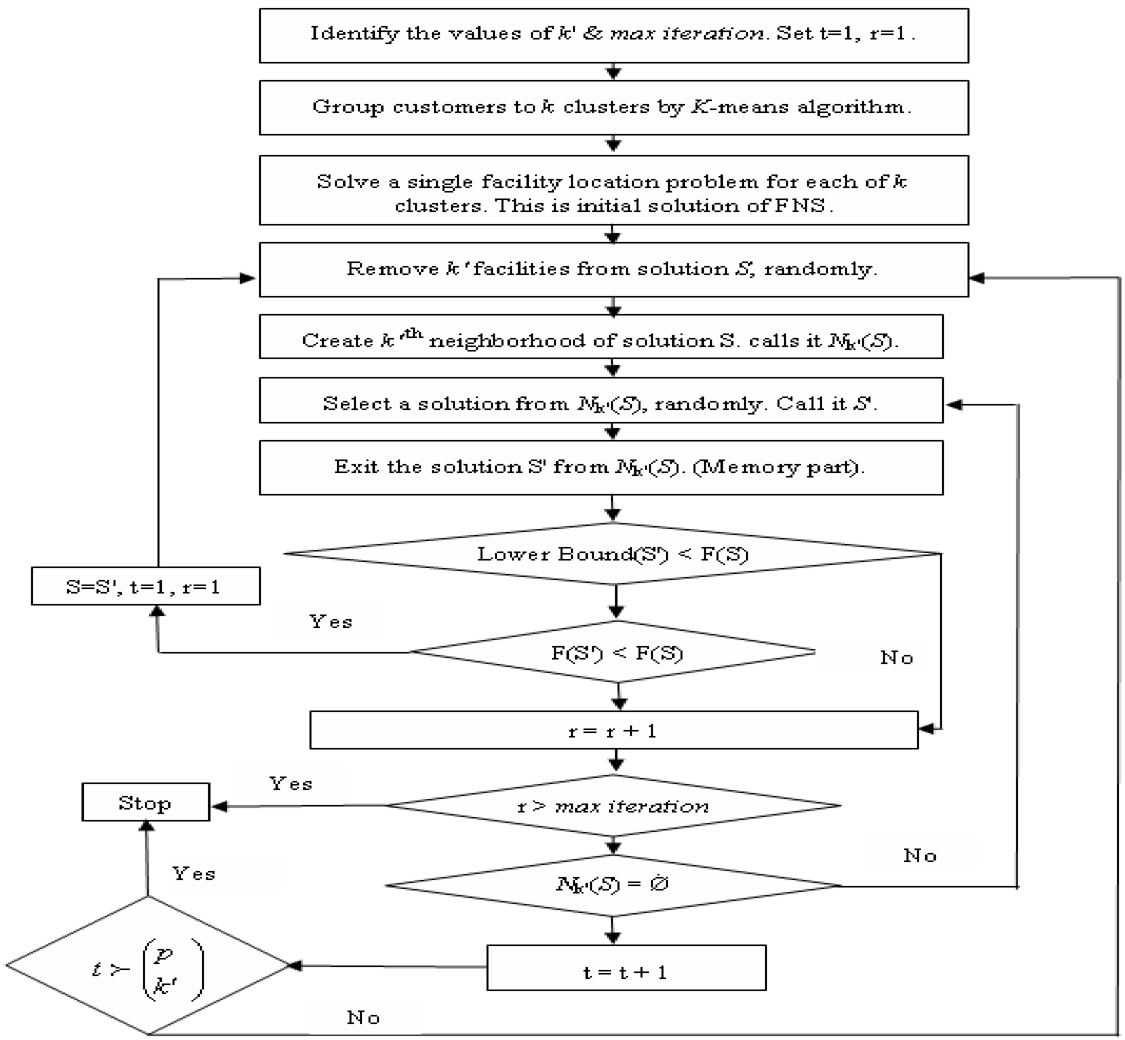

Figure 4: new hybrid algorithm

3.3.2 Stop condition

In the proposed hybrid algorithm, in addition to stop condition of maximum number of iteration, another stop condition is also used. In this new condition, if for any of the *k'* different facilities which are removed from the current solution, the best available solution dose not improves, the algorithm will be finished. In other words, in this new condition, all of available solutions in the neighborhood are considered. Then, algorithm is stopped. According to above discussion, steps of proposed new hybrid algorithm are shown in Figure 5. Also, property of having the memory of Tabu Search algorithm [11, 12] is used to avoid the re-evaluation of repeated solutions. The naive lower bound [2] was used to make the solution time of proposed algorithm faster also. It means; firstly, lower bound of the each of produced solution is calculated. If the value of the lower bound be worse than value of the best available solution, that produced solution is not examined and a new solution is produced, but if it is better, that solution will be considered.

4. Computational Results

In order to evaluate the efficiency of proposed algorithm, it was tested on two sets of standard benchmark problem instances. The first instances set contains 10 problem instances of size *50 customers* and *5* facilities, (P_1 *to* P_{10}), and second instances set contains 10 problem instances of size *100 customers* and *10* facilities, (P_{11} *to* P_{20}). This two standard benchmark instances are available in the OR library (http://mscmga. ms. ic. ac. uk/info. html) [13]. The propose hybrid algorithm was encoded in *MATLAB.7.4.0* (*R2007a*) and carried out on a Pentium *IV* with 3.2 *GHZ* and 1GB *RAM*. Also, the Generalized Assignment Problem (GAP) was solved by the *bintprog* function of MATLAB software.

Table 1: Comparison of proposed method with other heuristics

Problem	OPT	NO. opt	PR+SS	VNS	Proposed Method		dev			Time(S)	
					K-means	FNS	PR+SS	VNS	FNS	Hybrid	Lingo
P_1	713	10	713	713	714	713	0. 00	0.00	0.0	1.69	18
P_2	740	10	740	740	741	740	0. 00	0.00	0.0	1.1	15
P_3	751	10	751	751	754	751	0. 00	0.00	0.0	6.2	17
P_4	651	10	651	651	651	651	0. 00	0.00	0.0	1.2	15
P_5	664	10	664	664	664	664	0. 00	0.00	0.0	3.11	16
P_6	778	10	778	778	782	778	0. 00	0.00	0.0	3.66	17
P_7	787	10	787	787	847	787	0. 00	0.00	0.0	21.5	74
P_8	820	10	820. 9	820	823	820	0. 11	0.00	0.0	348	1010
P_9	715	10	715	715	717	715	0. 00	0.00	0.0	3.75	23
P_{10}	829	10	831. 4	829	842	829	0. 29	0.00	0.0	73.9	228
P_{11}	1006	10	1006	1006	1025	1006	0. 00	0.00	0.0	130	5700
P_{12}	966	10	966	966	984	966	0. 00	0.00	0.0	26	1896
P_{13}	1026	10	1026	1026	1046	1026	0. 00	0.00	0.0	876	942
P_{14}	982	10	983. 7	982	996	982	0. 17	0.00	0.0	2400	9480
P_{15}	1091	9	1092. 2	1091	1100	1091.3	0. 11	0.00	0.02	1232	6480
P_{16}	954	10	954	954	957	954	0. 00	0.00	0.0	7.48	1298
P_{17}	1034	9	1034	1034	1090	1035.4	0. 00	0.00	0.13	2603	4020
P_{18}	1043	10	1043. 2	1043	1063	1043	0. 02	0.00	0.0	2172	3540
P_{19}	1031	10	1032	1031	1039	1031	0. 10	0.00	0.0	757	2842
P_{20}	1005	10	1006	1005	1048	1005	0. 10	0.00	0.0	1600	97200

In all problems, *k'*=1.To see if the proposed algorithm is suitable or not, we make a comparison between its results and the results of other algorithms, it means *PR+SS*, algorithm [7] and *VNS* [2]. Table 1, shows the result of this comparison. Except *VNS* algorithm, other considered algorithms performed in 10 runs. In table 1, the column of *Problem***,** the number of each problem, the *OPT* column, the optimum solution of each problem, and the *NO.opt* column show the number of times, out of the 10 runs, that the proposed algorithm found an optimal solution. Fourth and Fifth columns show results of hybrid Path Relinking and Scatter Search approach (PR+SS) and Variable Neighborhood Search (VNS) algorithm. The *Proposed Method* column shows the results of *K*-means clustering algorithm and the average of the best obtained values at 10 runs of the proposed algorithm. The *dev* column also shows the percent of deviation from the best-known solution which is calculated by $dev = \frac{FNS - optimal}{optimal} \times 100$.The *time* column shows the solution time of problems in new proposed algorithm and software of *LINGO.8*. Comparing the results of table 2, we will understand that, the proposed algorithm has had good performance for each of problems with size *n=50, 5* facilities. although in the proposed hybrid algorithm, the K-means clustering algorithm only could find optimum solution in P_4 and P_5, in other problem with size *n=50* and *5* facilities, the FNS algorithm was able to find optimum solution in each of 10 runs by proper exchanging and omitting the unsuitable candidate points. In the first instances set, the deviation percent of results of proposed algorithm from the best-known solution is %0.00. It is noted that in P_8 and P_{10} problems *PR+SS* algorithm is not able to find optimum solution in all 10 runs. This algorithm has bad performance than proposed algorithm and in two above-mentioned problems has had deviation*%0.11 and %0.29* respectively. Of course, *VNS* algorithm finds an optimum solution for each of P_1 to P_{10} problems also. For the problems P_{11} to P_{20}, PR+SS algorithm has a worse result than the result of proposed method also. According to the results of proposed hybrid algorithm, for the problems P_{11} to P_{20}, it could be understood that the new method is able to find a solution in 10 runs (except problem P_{17}), which is so better than PR+SS. of course, For problem P_{15}, in 10 runs of new method, there is a result which has a deviation of %0.02, but this deviation is less than the deviation of PR+SS algorithm. But, comparing with the results of VNS algorithm, the proposed algorithm just only had worse results in problems P_{15} and P_{17}. Of course, this new proposed method for the problems P_{15} and P_{17} in 9 out of the 10 runs finds the optimum solution.

Due to each considered algorithms in table 1 was encoded in different softwares and computers so, we avoid comparing their time. Also, considering the solution time of the proposed algorithm and LINGO software, we will get that this algorithm has a better times in all of different problems.

5. Conclusions

In this paper, a new hybrid algorithm is proposed to solve the capacitated *P*-median. We used two sets of standard benchmark problems instances to evaluate the efficiency of the proposed method. The results show that the power of the proposed algorithm is due to; first, omitting of unsuitable candidate points, second, decreasing number of available solutions in the neighborhood and third, avoiding to re-evaluation of repeated solutions.

References

1. Pirkul, H., 1987, "Efficient algorithm for the capacitated concentrator location problem," computers and operations research, 14 (3), 197-208.
2. Fleszar, K., and Hindi, K, S., 2008, "An effective VNS for the capacitated P-median problem," European Journal of Operational Research, 191, 612-622.
3. Bozkaya, B., Erkut, E., and Laporte, G., 2003, "A tabu search heuristic and adaptive memory procedure for political districting," European Journal of Operational Research, 144(1), 12-26.
4. Mulvey, J, M., and Beck, M.P., 1984, "Solving capacitated clustering problems," European Journal of Operational Research, 18, 317-336.
5. Garey, M. R., and Johnson, D.S., 1979. Computers and Intractability: A Guide to the Theory of np-Completeness. W.H. Freeman and CO.
6. Lorena, L. A. N. and Senne, E. L. F. 2004, "A column generation approach to capacitated P-median problems," Journal of Computers & Operations Research, 31, 863-876.
7. Diaz Juan A., Elena Fernandez., 2006, "Hybrid scatter search and path relinking for the capacitated P-median problem," European Journal of Operational Research, 169, 570-585.
8. Hansen Pierre, Mladenovic Nenad., 1997, "Variable Neighborhood search for the *p*-median," Location Science, 5, 207-226.
9. Yagiura, M., Ibaraki, T., and Glover, F., 2006, "A path relinking approach with enjection chains for the generalized assignment problem," European Journal of Operational Research, 169(2), 548-569.
10. McQueen, J. B., 1967. Some methods for classification and analysis of multivariate observations. In: Proceedings of the Symposium on Mathematics and Probability, 5th, Berkeley, Vol. 1, AD 669871, University of California Press, Berkeley, CA, 281-297.
11. Glover. F., 1989. Tabu search – part 1, ORSA j. Compute. 1. 190-206.
12. Glover. F., 1990, Tabu search – part 2, ORSA j. Compute. 2. 4-32.

13. Beasley, J.E., 1990. OR – library: distributing test problems by electronic mail. Journal of the Operational Research Society, 41 (11), 1069-1072.