International Journal of Information Technology and Knowledge Management
July-December 2008, Volume 1, No. 2, pp. 527-535

THE ANALYTICAL STUDY OF K-CENTER PROBLEM SOLVING TECHNIQUES

Rattan Rana & Deepak Garg

Abstract

Facility location decisions are a key element in any firm's overall strategic plan. It is one of the main issues for contemporary manufacturing and service firms. Specifically, in the era of global markets and global production, changes in the global geopolitical environment have made the world truly a global factory, allowing companies greater flexibility in their location choices. In these situations the need for special capabilities of choosing the best locations is often far more important than other cost factors. The *k* center problem is concerned with the location of *k*-points (centers) on the network, such that the maximum (weighted) distance of all the nodes to their respective nearest points is minimized. In last few decades a number of approaches have been applied to provide an optimum solution for the *k-center* problem. This paper presents a comparative study of different approaches used to solve the *k-center* problem.

***Keywords*:** Facility location, Site, *k-center*, heuristics, greedy, optimizing,

1. Introduction

Facility location problem is a resource allocation problem that mainly deals with adequate placement of various types of facilities. First is to serve a distributed set of demands satisfying the nature of interaction between the demands and second is Optimizing the cost of placing the facilities and the quality of service. The facility location problem is a well studied problem in the operation research literature and recently has received a lot of attention in the computer science community. For a company, the facility location problem provides more strategic decisions than just giving importance to locate the lowest cost space for storing its products. While identifying the best location for the company's distribution center, it must consider several things like the cost of new structure, new freight, inherent risk and the viability involved in choice of these facilities. Several variations of facility location problems can be formulated depending on the constraints on the nature of the facilities, and objective functions, e.g. the cost reduction, demand capture, fast response time etc. For locating the emergency facilities, such as hospitals, fire-fighting stations etc., covering a region using minimum radius circles is a natural mapping of the corresponding facility location problem where the objective is to minimize the worst-case response time. The placement of multiple facilities (or centers) in such a way that the each center can cover maximum number of demand nodes and the distance of all centers should be as minimum as possible, it is one of the biggest issues for researchers. This problem is also known as *k*-center problem.

1.1. Importance of *k*-centers

The facility location problem involves the optimal placement of *k*-facilities in networks of *N*-nodes. Problem of finding the best location of facilities in network or graphs is crucial in practical situations. One of the best known facility location problems is the vertex *k*-center problem, where given *n* node and distances between all pairs of nodes, the goal is to choose *k* nodes (known as centers) so that the largest distance of a node to its nearest center is minimal. Formally, the *k*-center can be defined as follows. Let $G = (V, E)$ be a complete undirected graph with edge costs satisfying the triangle inequality, and *k* be a positive integer not greater than $|V|$. For any set $S \subseteq V$, and vertex $v \in V$, we define $d(v, S)$ to be the length of a shortest edge from *v* to any vertex in *S*. The problem is to find such a set $S \subseteq V$, where $|S| \leq k$, which minimizes $\max_{v \in V} d(v, S)$. The vertex *k*-center problem is NP-hard.

1.2. Mathematical Formulation

The *k*-center problem can be formulated as: $\underset{X \text{ on } G}{Minimize}\ f(X) = \max_{1 \leq i \leq n} h_i\ d(v_i, X)$

2. LITERATURE SURVEY

The classical facility location problems are known as *k-center* and *k-median* problems, where *k* is a positive integer. Sylvester was the first person who concentrated on 1-*center* problem in 1857 [1]. Similarly Weber has paid attention towards 1-*median* problem in 1909, which is known as Weber's problem [2]. Basically, 1-*median* problem is to find the best location of a single facility in plane and the objective is to reduce the average distance of all the demand nodes to the facility; this is the case of single facility. If we have to locate more than one facilities i.e. ($k > 1$), then it is called as *k-median* problem. While in *k-center* problem the objective is to find a set *S* of *k*-facilities, so that the maximum distance between a demand point and its nearest facility is minimized.

In another study it is found that the parametric technique is a useful tool to solve problems having small values of *k* (or the fixed values of *k*). The decision version of parametric technique was used to find whether a set of demand points can be covered by union of *k*-facilities of a given area of radius *r*. J. Elzinga and Hearn proposed first algorithm with an $O(n^2)$ time complexity for Euclidean *1-center* problem [3]. Later, Shamos and Hoey [4], Preparata [5] and Shamos [6] have given an improved algorithm with time complexity $O(n \log n)$. In 1983 Megiddo has used prune and search technique to solve the same problem and found an optimal algorithm with time complexity $O(n)$ [7].

The unconstrained version of classical 1-*center* problem has got a lot of attention of researchers, but the *k-center* problem has got a little bit.

In year 2000 Kim *et al*., [8] has given two efficient algorithms for both standard 2-*center* problem and discrete 2-*center* problem, where the demand points are assumed as vertices of a given convex polygon. These algorithms run in $O(n \log^3 n \log \log n)$ and $O(n \log^2 n)$ time respectively. Recently, in year 2006 B.K. Bhattacharya and Ben-Moshe [9] has proposed a linear time algorithm for the weighted 2-*center* problem. They provide the improved results on the upper bound of the continuous/discrete weighted *k-center* problem on a tree.

The greedy heuristic approach can also be used to solve the *k-center* problem. If, we opt the heuristic approach for *k-center* problem then we need to apply the heuristic method for *n*-times, $n = |V|$, select with a new vertex each time and then select the best solution. These approaches are known as *random*, 1-*center*, and *plus* version, respectively.Gonzalez [10] has applied pure greedy approach to solve the *k-center* problem. Gonzalez has designed an algorithm with approximation factor 2 and the time complexity of algorithm was $O(kn)$. As we go in more details of k-center problem literature, we got very interesting techniques to solve the problem. The decision version of *k-center* problem is described by Shmoys [11] with approximation factor 2. The problem is defined as that we have given an area of radius *r*. The algorithm for the optimization version of problem runs the algorithm for the decision version several times with increasing value of *r*. Time complexity of this algorithm is $O(kn^3)$. Another algorithmic technique called as parametric pruning is introduced by Hochbaum and Shmoys [12] to solve the *k-center* problem. This algorithm works in different fashion. The aim is to find a minimum dominating set in the pruned graph, i.e. the smallest set *S* of vertices such that every vertex not in *S* is adjacent to one of the vertices in *S*. But, to compute the minimum dominating set is NP-hard optimization problem as proved by M. R. Garey and D. S. Johnson [13]. Jurij and Borut [14] has developed a new heuristic algorithm to solve the dominating set problems and given better results.

Minieka [15] solved the *k-center* problem as a series of set cover problems. Daskin has described this approach in more details [16,17]. Jurij and Borut [14] have also made an experimental evaluation of the approximation algorithms for *k-center* problem. In which a quality comparison is made on the clustering algorithm by Gonzalez, the parametric pruning algorithm by Hochbaum-Shmoys, and Shmoys' algorithm. Several variants of greedy approach were also discussed. They concluded that the Gonzalez algorithms were the fastest and solutions are about 50% worse than best known. The pure greedy methods were quite fast. But their execution time was very variable and dependent on the parameter *k*, Shmoys' variants were also fast. The solution quality strongly depends on the parameter *k*, and is much better for low values of *k*.

Only a few researchers have focused on the *k-center* problem in the literature due to NP-hard nature of the problem. The subsequent section highlights the different techniques used to solve the location models or *k*-center problem

3. TECHNIQUES USED TO SOLVE THE *k*-CENTER PROBLEM

This section describes about different approaches used to solve the facility location problems. In the literature of location models, it has been observed that the mixed integer linear programming is used for formulating the problem, but it is the first step of problem solving process. The next important step is to find the optimal solution. To achieve the optimal solution we need some specified techniques. All these techniques are described in subsequent sub section.

3.1. Branch and Bound Technique

Branch and bound is one of the well known algorithm used to solve such kind of problems. But, it has some restrictions. It can be implemented only on small problems. Because, it works on some instances of location models. While the facility location problem is scaled location model. Since the optimization problems consumes unexpected computational time and other resources without any guarantee of optimal solution. Therefore the location models are classified as NP-Hard.

3.2. Greedy Heuristics

The greedy heuristic can be implemented in the situations where we have to select a subset of objects. This is helpful in optimizing some objectives. The most common is the sequential approach in which individual site is evaluated and that site is selected that provides the highest impact on objective [22]. This location is fixed. Then the next location is selected from all remaining sites. Now choose the location that is more close to fulfilling the objective. Repeat the process till required numbers of sites are identified. For this reason this approach is called as Greedy Heuristic, specifically it is known as Greedy-Add. Another version of Greedy Heuristic is known as Greedy-Drop, since it removes the site which has least impact on the objective during the process of site selection. We continue the removing process till the required number of facilities or sites remain. Both of these algorithms provides good solutions but can't optimize.

Improved Heuristic Search

Although the Greedy-Add and Greedy-Drop provide good (or feasible) solution for location model, but these couldn't provide consistently good solutions for all problems. That's why some of new algorithms developed that starts from the results given by heuristic algorithms to improve the solution. These are known as Improved Heuristics

Neighborhood Search

The neighborhood search algorithm is one of the improvements heuristic. In this technique we start search from any feasible solution given by any of the greedy heuristics. Here, we assign a set of demand nodes to a nearest facility. These demand nodes constitute neighborhood around the facility site. Then each potential site is evaluated in neighborhood and selects the best one. If, any site is relocated, then new neighborhood is defined and algorithm is repeated. This process is repeated until there is no change found.

But there is one limitation of the neighborhood search algorithm that, in evaluating the impact of any relocation decision, only the effect on those nodes in the neighborhood is considered. The potential benefit to nodes outside of the neighborhood is not considered in deciding whether or not relocation should be made. This makes the researchers to think about the exchange or interchange algorithm as an alternative improvement procedure.

Interchange Heuristic Approach

This interchange heuristic approach is introduced by Teitz and Bart (1968). In this approach the location of facility is moved to any unused site and each site is tried in turn, which site gives improvement in objective function, that new site is selected as facility location. The problem with many search heuristics is that, instead of yielding the required optimal solution, they become stiff in local optima. Then researchers planned to apply the heuristics in more intelligent manner that is called as *metaheuristic*. The basic idea behind this is to break out local optima and search other regions of the solution space. inhibit

One of the earliest metaheuristic is Tabu search. The Tabu Search heuristic involves defining what type of exchanges to restrict and nature of the aspiration criteria and short-term memory to utilize. The subsequent section will provide a comparative analysis of all the above given techniques. In fact, it will show different technical complexities of their implementation.

4. COMPARATIVE STUDY OF THE TECHNIQUES USED TO SOLVE *k*-CENTER PROBLEM

This section presents a brief description and the analysis on different heuristic techniques used in solving *k*-center problem. Here, the experimental results of different heuristic techniques are also shown [18].The greedy heuristic and its different versions are commonly used to solve *k*-center problem. First is greedy (Gr) method [31] in which first facility is located in such a way as to minimize the maximum cost. Facilities are then added one by one until the required number of facilities are located. Each time the total cost is reduced. The improved versions of greedy method are greedy first random, greedy 1-center and greedy plus. *Greedy random* (GrR) chooses first center randomly. *Greedy* 1-*center* (Gr1) used to solve the 1-center problem. While the *greedy plus* (Gr+) Gonzalez, Dyer and Frieze have described another greedy heuristic for the *k*-center problem. In which, they have proven the approximation factor 2. The average and deviation ratio of different algorithms is shown in Chart 1.

The solutions returned by greedy algorithms are typically of better quality than randomly generated solutions, yet, a disadvantage of greedy heuristics is that they can generate only a limited number of different solutions. Additionally, their results in early stages of process strongly constrain the available possibilities at later stages, often causing very poor moves in the final phases of the solution. As far as the complexity of algorithm is concern, that could not be reduced at any level of heuristic techniques.

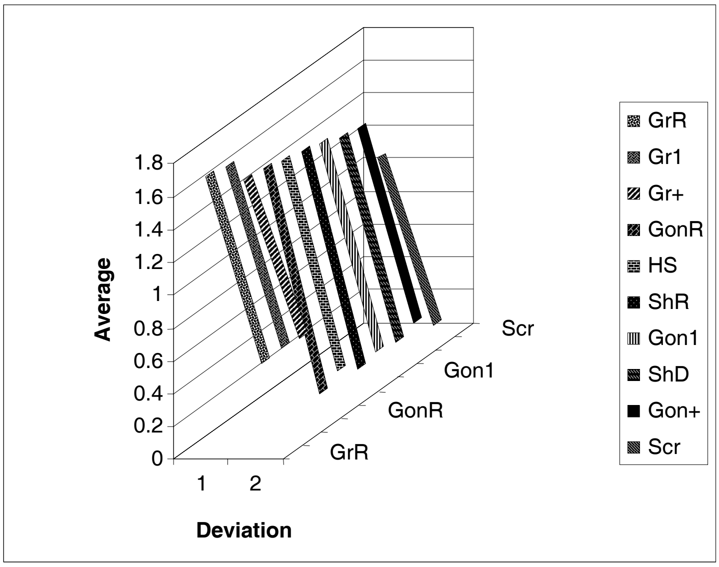

Figure 1: Average and Deviation Chart of Different Algorithms

In the 1990 Wang and kam Hoi Cheng has shown parallel time complexity of a heuristic algorithm for the *k*-center problem. In which, the greedy strategy is used to choose the vertex with maximum usages weights as the next service vertex. They found that the results of greedy strategy are no greater than twice the optimal solution value. They made the comparison of time complexity of greedy strategy on uniprocessor and multiprocessor systems. The time complexity on uniprocess is $O(n^3)$ and on multiprocessor system is $O(n \log^2 n)$, where n is the total number of vertices. It shows that, inspite of having unlimited parallelism the algorithm has higher than polylogrithmic time complexity.

Hence, it has been proven that the greedy heuristic couldn't provide optimal solution for *k*-center problem even on implemented with multiprocessor system. D. Chen and Reuven Chen have introduced a new relaxed based optimal algorithm for the *p*-center problem, in which they added some new variants. This algorithm works somewhat differently than the traditional iterative algorithms. The traditional iterative algorithms required solving a few large problems while the relaxation algorithms need to solve many small problems. The experimental results of D. Chen shows that the relaxation based algorithms are best suited for problems with small values of *p*. The performance graph of IIhan *et al*.'s algorithm and relaxation algorithm is shown in figure 1. It shows that relaxation algorithms performs very well for small values of *p*, but as value of *p* increases it loses its advantage over traditional iterative algorithm. The figure 2 also shows that the behavior of both the algorithms is unpredictable

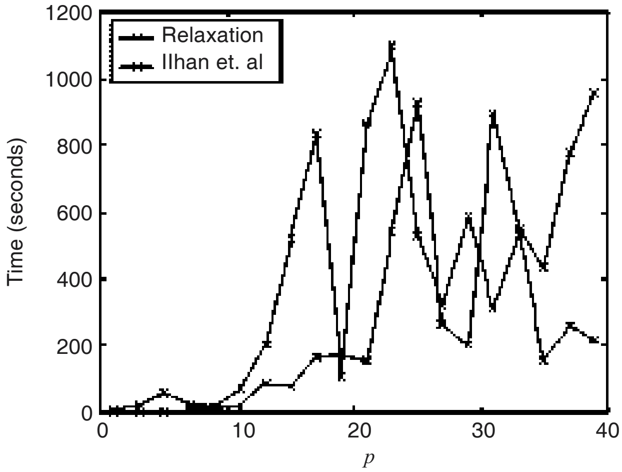

Figure 2: Performance Comparison of Relaxation Algorithm and IIhan *et al.* Algorithms

Thereafter, Chen and Handler proposed an algorithm to solve the *p*-center problem by first solving 1-center, 2-center and eventually the (*p*-1)-center problem. But the essential requirement to start solving problem is that we must know in advance the tight upper bound on solution.

5. Result Discussion

The working custom of different techniques has been described in previous subsection. Now, it is necessary to highlight the efficiency and accuracy of these techniques according to their performance. Jurij Mihelic and Borut Robic [18] have shown some practical aspects of these techniques. If all of these algorithms are performed on a particular set of vertices, then how these algorithms perform?

In case of any particular problem the pure greedy shows worst results while the greedy plus gives slightly better results. The basic constraint of pure greedy algorithm is that it is highly dependent on the parameter *k*. In case of low number of centers it provides better results, otherwise it shows worst performance.

Gonzalez algorithm provides good results but Gonzalez plus version shows much better results. Although, the results of Gonzalez's algorithm is approximately 32% above the optimal. Similarly, HS, ShR behaves and returns same results.In 2005 Jurij Mihelic and Borut Robic assert that their Scoring algorithm (Scr) can provide best results till date. But on an average the results of Scr are 6% above the optimal.

The pure greedy algorithm is quite fast, but its execution time is highly dependent on the value of parameter *k*. Even though the greedy plus version runs much slower because it tries all the vertices for 1-center.

6. Conclusion and Future Work

It has been observed that different researchers have followed different approaches to solve the k-center problem. Each approach has its own advantages and disadvantages. Undoubtedly, most of them provide better results. But a common problem which is observed in each approach is their execution time. The results are strongly dependent on the value of parameter *k*, as the value of *k* increase the performance of algorithm goes slower and results obtained are so far than the optimum solution. So, it can be suggested for future work that it is the required to develop a such algorithm which performs equally with any value of *k*.

References

[1] J. J. Sylvester, *A Question in the Geometry of Situation*, Quarterly *Journal of Mathematics*, (1857), 1-79.

[2] A. Weber, *er den Standort der Industrien*, Tuebingen, (1909).

[3] J. Elzinga and D. W. Hearn, Geometrical Solutions to Some Minimax Location Problems, *Transportation Science*, **6**, (1972), 379–394.

[4] M. I. Shamos and D. Hoey, Closest-point Problem, *Proc. of the 16th Annual IEEE Symposium Foundation Computer Science*, (1975), 151–162.

[5] F. P. Preparata, Minimum Spanning Circle, Technical Report, Univ. Illinois, Urbana, IL, in *Steps into Computational Geometry*, (1977).

[6] M. I. Shamos, Computational Geometry, *PhD. Thesis.* Department of Computer Science, Yale University, New Haven, CT, (1978).

[7] N. Megiddo, Linear-time Algorithms for Linear Programming in R3 and Related Problems, *SIAM Journal on Computing*, **12**, (1983), 759–776.

[8] S. K. Kim, and C. S Shin, Efficient Algorithms for Two-center Problems for a Convex Polygon, *Proc. of the 6th International Conference on Computing and Combinatorics*, LNCS 1858, (2000), 299–309.

[9] B. Ben-Moshe, B. K. Bhattacharya, Q. Shi, An Optimal Algorithm for the Continuous/discrete Weighted 2-center Problem in Trees, *Proc. of 7th Latin American Theoretical Informatics Symposium*, LNCS 3887, (2006), 166–177.

[10] T. Gonzalez. Clustering to Minimize the Maximum Intercluster Distance. *Theoretical Computer Science,* **38**, (1985), 293–306.

[11] David B. Shmoys. Computing Near-optimal Solutions to Combinatorial Optimization Problems. *Technical Report*, Ithaca, NY 14853, (1995).

[12] Dorit S. Hochbaum and David B. Shmoys. A Best Possible Heuristic for the *k*-center Problem. *Mathematics of Operations Research*, **10**, (1985),180–184.

[13] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., San Francisco, 1979.

[14] Jurij Mihelic and Borut Robic. *Approximation Algorithms for the k-center Problem*: an Experimental Evaluation.

[15] E. Minieka. The *m*-center Problem. *SIAM Rev.*, **12**, (1970),138–139.

[16] Mark S. Daskin. *Network and Discrete Location: Models Algorithms and Applications*. Wiley, New York, (1995).

[17] Mark S. Daskin. A New Approach to Solving the Vertex *p*-center Problem to Optimality: Algorithm and Computational Results. *Communications of the Operations Research Society of Japan*, **45** (9), (2000), 428–436.

[18] Jurij Mihelic and Borut Robic. Solving the *k*-center Problem Efficiently with a Dominating Set Algorithm. *Journal of Computing and Information Technology-CIT* **13** (3), (2005), 225–233.

Rattan Rana & Deepak Garg
Computer Science & Engineering Department
Thapar University
Patiala (India)