DESIGN AND SIMULATION OF ALGORITHM FOR THE K-CENTRE PROBLEM

A Thesis

Submitted in fulfillment of the
Requirements for the award of the degree of

Doctor of Philosophy

Submitted By

Rattan Pal Singh (Reg. No. 950803013)

Under the supervision of

Dr. Deepak Garg

Associate Professor & Head
Computer Science and Engineering Department
Thapar University
Patiala - 147004

COMPUTER SCIENCE & ENGINEERING DEPARTMENT

Thapar University, Patiala

November, 2014

CERTIFICATE

Certified that the work contained in the thesis titled "Design and Simulation of Algorithm for the K-Centre Problem", by Rattan Pal Singh (Reg. No. 950803013), has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Rattan Pal Singh (Research Scholar)
Department of Computer Science & Engg.
Thapar University, Patiala

Dr. Deepak Garg (Supervisor)
Department of Computer Science & Engg.
Thapar University, Patiala

November, 2014

Paper Published

- The Analytical Study of K-Center Problem Solving Techniques" "International Journal of Information Technology and Knowledge Management", ISSN: 0973-4414, Issue 2 Vol.1 (pp 527-535), 2008.

- "Heuristic Approaches for k-center problem" IEEE Xplore Advance Computing Catalog N CFP0939F ISBN-978-1-4244-2928-8 (pp: 332-335) April, 2009.

- "An Evaluation of K-Center Problem Solving Techniques towards Optimality" International Journal of Advancements in Technology (IJoAT), ISSN: 0976-4860, Issue 2 Vol. 2, (pp. 206-214) April, 2011.

- "Anticipatory Bound Selection Procedure (ABSP) for Vertex K-center Problem" International Arab Journal of Information Technology (IAJIT), Vol. 11, No. 5, September 2014.
  Indexed in: Science Citation Index Expanded
  Impact Factor: 0.39

- "Algorithm for Obnoxious Facility Location Problem" International Journal of Advancements in Technology (IJoAT), ISSN: 0976-4860, Vol. 5, No. 2, pp. 96-106, July 2014.

- "An Improved Approach for Incremental Variant of Mobile Facility Location Problem" Communicated with Turkish Journal Of Electrical Engineering & Computer Science. October 2013.
  Indexed in: Science Citation Index Expanded

Abstract

Facility location problem addresses the appropriate allocation of facilities in a given region. Facility location problem has become a challenge for public and private sectors. Since, both public and private agencies are committed to provide specified services to their respective customers/clients; within particular range of distance and time. Therefore an appropriate location of facility is matter of priority for all service providers indeed.

Facilities may be of various kinds such as Static Facilities (e.g. Schools, Hospitals, Automobile workstations, College, & Fire Stations etc) and Mobile Facilities (e.g. PCR vans, & Ambulance etc). There are so many important parameters (such as distance, cost, time and demand); which are essential to consider during the allocation of a facility.

In this thesis, we present an approach that is known as Anticipatory Bound Selection Procedure (***ABSP***). ***ABSP*** is utilized in collaboration of Jump Based Scheme (***JBS***) to provide an efficient way to solve the vertex *k*-center problem. This collaboration of these two methods has proven its significance by providing exciting results over the performance of other existing algorithms. It has reduced the number of iterations (which are required to allocate *k*-facilities) drastically. Consequently the time required to determine the appropriate locations for allocating facilities has reduced considerably.

Moreover the presented framework provides an algorithm for obnoxious facility location problem (***OFLP***). The obnoxious material management is

really a considerable issue, since it has adverse effects on environment as well as the health of human being. This simplified method shown wonderful results during the simulation. Further as the results suggest that there is a slight increase in execution time with the increased number of nodes.

Finally this thesis presents an improved approach for the incremental variant of mobile facility location problem. It undertakes the study of increased demand with the passage of time and its impact on existing facilities as well as requirement of new facilities. Presented approach provides an efficient method to solve all the related issues of mobile facility location problem in a dynamic environment.

Acknowledgement

First and foremost, I bow my head humbly before the Almighty God for making me capable of completing my Ph.D. Thesis. With His blessings only I have been able to accomplish this challenge.

My first debt of gratitude must go to my supervisor Dr. Deepak Garg (Associate Professor & Head of Department of Computer Science & Engineering, Thapar University, Patiala). Dr Deepak Garg patiently provided the vision, inspiring guidance, encouragement, and counseling necessary for me to proceed through the doctoral program and complete my dissertation.

I am grateful to Dr. Seema Bawa and Dr. Maninder Singh former Heads of Department of Computer Science and Engineering, Thapar University, Patiala for their constant support, wishes and motivation.

I am extremely grateful to the member of Doctoral Committee Dr. V. P. Singh and Dr. R. K. Sharma for their encouragement, insightful comments and valuable suggestions during these last five years.

I am Thankful to Dr. Dimple Juneja Professor & Principal MMICT&BM, Maharishi Markandeshwar University, Mullana (Ambala, Haryana), Dr. Kanwal Garg Associate Professor, Kurukshetra University, Kurukshetra and Dr. Vijay Kumar Assistant Professor, Haryana Engineering College, Jagadhri for their constant support wishes and motivation during my research work.

I deeply indebted to my parents for their love, prayers, care and sacrifices for educating and preparing me for my future. I am extremely grateful to my wife Mrs. Sonia for her unconditional support and prayers. I feel fortunate to have her in my life.

Also I express my thanks to my sisters, brothers, sisters-in law and brothers-in-law for their constant encouragement, without which I could not have been succeeded.

My loving thanks are due to my son Aditya, and Arnav for their sacrifice and silent wishes.

Last but not the least; I would like to thank all the people associated with me for their moral support in my research work.

This Dissertation is dedicated

To my father Late Sh. Rai Singh Rana

Contents

List of Figures

Page No.

List of Tables

Chapter 1

Introduction

Facility location is a decisive type of problem having an extensive range of sub problems that have been studied within various fields such as Operation Research, Computational Complexity and Graph Theory. This area of research has a long history and persistent activity accompanied by well off literature. The motive is to achieve the best location of facilities in a network in realistic situations.

1.1 Facility Location Problem

From the ancient days, when human was used to live in caves, it was the primary objective to find out a secure and effective location. Hence, the locations, which were capable to provide protection, were selected for residential purposes. In ancient days, the population was scattered in small troops (i.e. villages and small towns), and earning sources were very limited. Initially, they were doing jobs as per their skills such as Carpenter, Mason, Blacksmith, Goldsmith, Vaidya and other professions in local areas to survive. Even though, they were lacking in basic amenities such as medical, transport etc. In present era, the society developed in various fields, such as agriculture, education, infrastructure, roads, transport, manufacturing, retails, medical, sanitation, communication, and technology, etc, which have directly affected the human life. Moreover, government has started to pay attention towards the basic amenities of human being. Obviously, it is responsibility of state government to provide basic facilities to their citizens. In fact, the private sector is also contributing to the society in many ways and exploring business opportunities in a wide region. In other words, it has increased the competitiveness in both public as well as private sectors.

So, it is a challenge for both government welfare agencies and private companies to determine appropriate locations to build up schools, hospitals, fire stations, communication towers, airports, military ammunition depot, bus terminals, warehouses, agriculture service centers, workstations and many more. "How to facilitate a huge population scattered in a wide region with limited number of resources?" this is the very first question that comes in mind. Alternatively, it is required to find out such locations in given region, which can facilitate the surrounding population within minimum distance and access time.

The planning by small entrepreneur to big industrial houses, a fashion designer shop to a manufacturing unit, a five star hotel or any other entrepreneur , and government agencies start with the questions like; Where a garrison should be build?, Where should an automobile company locate its workstations? Which machines in a network should be designated as routers? And other related queries. These questions have intrigued society since the invention of franchises and other such business with distributed production or supply centers. According to M. S Daskin [1], it always remains crucial for both the public and private sectors to take a decision about an appropriate location. For example, any government needed to find out locations for buses, fire stations and ambulances. In such special cases of emergency services, poor decisions about locations may cause loss of property and life both. This is the basic requirement of private sector to locate company offices, manufacturing units, raw material collection as well as distribution centers, and sale points at prime locations. A wrong decision about location in this competitive era may lead to higher costs and subsequently reduced competitiveness. Therefore the facility location decision plays a very significant role in success and failure of any sector.

Additionally, there are so many factors which make the facility location decisions much complicated. The increased population over a time will proportionally increase the demand. Consequently, it will generate the requirement for more facility centers to accommodate the increased demand in future. Since a large investment is involved in opening of a new facility, therefore, it is important to locate the site so that no changes are needed for years to come. There may be some complexity and conflicting factors associated with the site. Therefore, it is also important to resolve such factors prior to the allocation of any facility centre. Otherwise, it may ruin all associated benefits with the site. Such decisions are called as **Strategic Decisions**. These vital decisions have long term implications. It is not very easy to answer a location problem. A few reasons are given as under:

(i) Uncertainty in future.
(ii) Complexity and conflicting factors associated with the site selection problem.
(iii) Constraints and limitations of resources for the production on the site.
(iv) Above all, the distance of access point to client sites, etc.

Let us consider an example of two sites for the location of a new manufacturing unit. Suppose site X is nearer to market but far from the raw material source. While, site Y is nearer to raw material source but far from the market. Site X is a rural location with cheap availability of labor. While site Y is an urban location with better availability of power. Similarly, we can list many factors for each site, some of them may be better for site X while others are much suitable at site Y. Which site to select? When site selection decision is needed, many options are available with relative strengths and weaknesses. A careful consideration is needed on an integrated framework before final site is selected.

Typically, these centers or facilities can be placed at strategic locations for a fixed opening cost and each location can be approached by a customer or client for a fixed connection cost. Here, we use the term **"Facility"** in broader sense. That is, it covers all such points which are provided to facilitate end user such as bus stands , hospitals, electronic switching centers, warehouses, industry, sale points, schools, emergency warning sirens, ATM's, banks, rain gauges and fire brigade stations etc.,

The problem underlying these questions, known as the **Facility Location Problem,** addresses concerns about how many facilities are needed and where they should be placed in order to minimize the cost of serving for a given set of customer demands.

To know more about the concept of "***Facility Centre***" and "***Demand Nodes***" (clients), let us suppose state government wants to place a very limited number of health care centers in a given region to provide the health services to the inhabitant of that region. Suppose A, B, C, D, E, F, G, H, I, J are the connected nodes or places having some weighted distance as shown in Figure 1.1. Now, we have to identify a few nodes among the given nodes as candidate sites for health care centers which are competent to provide services to all connected nodes in minimum access time. Keeping in view that, the poor locations can cost lives.

It suggests that we would like to locate the health care centre, so that the response time can be minimized. The locations which are selected for health centre are known as "***Facilities***" while the other connected nodes are known as "***Client***s" or demand nodes as shown Figure 1.2.

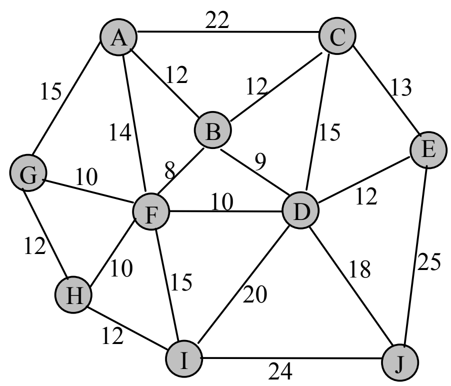


Figure 1.1: Network of Connected Nodes

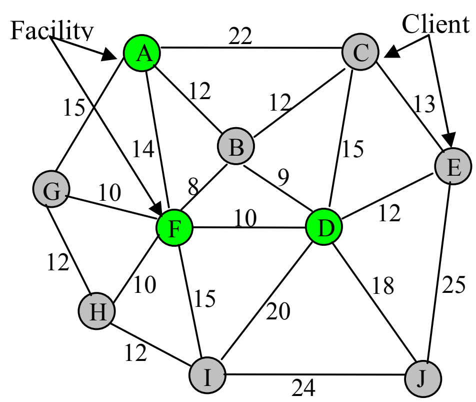


Figure 1.2: Illustrating Facility and Client Nodes

Systematic approaches to the facility location problem have been studied in the operations research literature since last four decades. The model design situations such as deciding placement of basic amenities factories, warehouses, fire stations, schools and hospitals have been explained widely. (For more details readers may refer to [2, 3 and 4].

Location problem can have following characteristics:

- ✓ It is recommended to provide the information about the cost of structure or opening a facility at any particular location. As well as Information about all locations where facilities may be open must be provided.

- ✓ It is required to provide the information about a set of demand nodes connected to a facility to avail services. It is also important to know about the demand of each connected demand node as well as the costs / profits incurred, if it would be served by a particular facility.
- ✓ It is also required that the open facility must fulfill all the requirements of the connected demand nodes.
- ✓ A function that associates to each set of facilities, the cost / profit incurred if one would open all the facilities in the set and would assign the demand points to them such that most of the requirements are satisfied.

The objective of problem is to find the set of facilities to be opened in order to optimize the given function. There are different types of facility location problems corresponding to the features of the four elements mentioned above. Some basic classes of facility location problems are listed below:

If the set of demand points and facility locations are finite, one speaks about a ***discrete facility location problem***, otherwise turn to a ***continuous facility location problem***. If all the data are accurate, models are referred to as ***deterministic***. If some parameter values are given by probability distributions, the model is considered to be ***stochastic***, If parameter values are uncertain, the model is known as ***robust*** and such problems are known as ***NP-hard***. We can further classify a model as ***capacitated*** as opposed to ***un-capacitated*** where the former term refers to upper bounds on the number of clients, a facility can serve. Broadly the facilities can be divided into two categories i.e. ***static*** and ***dynamic*** (mobile) facilities. The static facilities are those which are fixed and can't be moved such as hospitals, schools, airports, bus terminals and warehouses. Alternatively facilities those are not fixed in nature are called ***dynamic*** (as opposed to ***static***). Such models may be used in decision problems, where not only the location of facilities, but also ***"the moment in time"*** is important to establish them.

1.2 K-Centre Problem

Facility location problems proposed in operations research provide mathematical formulations of the common optimization aspects of these problems. The underlying problem i.e. ***k*-centre** problem and this is the sub-model of discrete location model [5].

The main objective of *k*-centre problem is to identify *k* locations among the given sites to locate facilities and minimize the maximum distance between demand node and its nearest facility. Here, *k* is a non negative integer variable representing number of facilities. There are two types of *k*-centre problem, namely "***absolute k-centre problem***" and "***vertex k-centre problem***" [1]. In case of *absolute k-centre* problem, facility can be located anywhere even on the edges of given network. Whereas, *vertex k-centre* problem, implies a condition that the facilities can be located only on vertices. This study focuses on *Vertex k-centre* problem which is part and parcel of the facility location problem.

It can be solved using set covering models but the *k-centre* problem reveals the internal complication of facility location problem i.e. to minimize the largest distance of node to its nearest centre.

1.3 Motivation

We have already heard about globalization and automation that is caused by technological revolution. Consequently, industrial and financial expansion will also be required. The beneficiaries of this industrial expansion should be the most needed areas with fewer endeavors. Further, the frequent changes in market trends suggest that it is essential to think over the increasing demand and population before locating a facility.

The modernization of society persists to provide prompt services to both public as well as private sector. Public sector promises to provide various kinds of services such as health, education, transportation, communication, disaster management, law & order, and many more. If such facility centers are not under the reach of a common man when needed most, then what's the use of these facilities?

Therefore, this is a challenge to identify appropriate locations in a wide area, where such facilities can be located with a firm objective to reduce the access time and traveling distance of clients or demand nodes. In operation research, various methods have been discussed to supply chain and logistics. Since last few decades, a great attention has been given to facility location problem as research area under computer science. Various researchers have developed different methods to solve the underlying problem as a contribution to the society. A firm intention to solve a realistic problem as contribution to the society was the driving force to motivate our research in this area.

1.4 Contribution of the Thesis

In this thesis, we have developed an algorithm to solve the facility location problem; more specifically this algorithm provides improved results for the *vertex k-center* problem. This algorithm provides comparatively better results than all existing methods. This algorithm introduces new methods for solving *k-centre* problem i.e. **Anticipatory Bound Selection Procedure** (**ABSP**) and **Jump Based Scheme** (**JBS**). Anticipatory Bound Selection Procedure helped to find out the initial threshold distance to check a feasible solution. While, previously it was done through ***Bisection method***. Bisection method assumes zero as the initial lower bound (initial distance) to calculate the first threshold distance, it also generates such distance values that do not exist in the distance matrix. While the ABSP selects such distance values from the distance matrix, so that the candidate to be at initial threshold distance but not at zero. Jump Based Scheme is advantageous in reducing the total number of iterations to determine an optimal solution for the problem. A simplified approach for **Obnoxious Facility Location Problem** (***OFLP***) is also incorporated as a part of study.

A simulator is developed in C++ to execute the presented algorithm. Recently developed algorithms are also executed on the same simulator and the results obtained from comparative analysis reveal that the presented work provides better results. The major achievement of presented algorithm is that it has reduced the large number of iteration required to find the optimal solution (approximately fifty percent faster than other methods).

In continuation of the presented work, we have also developed an algorithm for the incremental variant of the mobile facility location problem. Basically, this part of thesis addresses to the incremental type of mobile facility only. In other words, this algorithm is designed for the mobile facilities such as ambulance vans, patrol vans and fire brigade etc. Presented algorithm provides a method to facilitate a newly arrived demand node or increased demand after n^{th} period.

Both the algorithms presented in this thesis are very helpful for both the public as well as private sector in allocating the facilities in a given area under the specified constraints.

1.5 An Overview of the Thesis

This thesis provides an improved approach to solve the facility location problem by introducing Anticipatory Bound Selection Procedure and Jump Based Scheme. This thesis has been organized as follow:

Chapter 2 presents literature review. This chapter provides an insight to the origin of facility location problem and highlights substantial contribution of various researchers that promise to solve the problem in different aspects. It also presents the different types of facility location problem along with their available solutions and their drawbacks.

Chapter 3 presents different facility location models. This chapter provides an introduction to facility location models and their relevant techniques.

Chapter 4 presents an efficient algorithm that provides an optimal solution for vertex *k*-centre problem. Anticipatory Bound Selection Procedure is deployed to find the initial threshold distance (or radius) and eradicating the gap between minimum distance (lower bound) and maximum distance (upper bound). The Jump Based Scheme is applied to find the optimal coverage distance. The astonishing aspect of this algorithm is to provide optimal solution with fewer endeavors. This chapter provides a comprehensive description of methods used to implement the presented algorithm. On the basis of results obtained, a systematic analysis is also performed to show the effectiveness of methods implemented. It also provides an introduction to **Obnoxious Facility Location Problem** (*OFLP*) and a simplified approach to solve the ***OFLP*** for *1*-centre problem.

Chapter 5 introduces a new approach to deal with mobile facility location problems in a dynamic environment. This study presents an algorithm to solve the *k*-centre problem for mobile facility location problem. The significant aspect of our algorithm is that it undertakes both possibilities of facility location such as immediate assignment of facility and allocation of facilities to inward demand after n^{th} period.

Finally, in ***Chapter 6*** we draw some general conclusions and present suggestions for future research.

Chapter 2

Literature Review

Location problems have been extensively studied in both computer science and operations research literature due to their vast applicability in real world scenarios. Necessity of emergency services such as fire station and ambulance vans has increased the importance of facility location problems in location theory. All emergency services are the notion of coverage, a concept that is central to several classes of facility location models. It implies whether a demand location is within a pre specified radius (that can be measured by distance, travel time, cost and other parameters) of its assigned facility.

2.1 Background

The classical facility location problems are known as ***k-centre*** and ***k-median*** problems, where ***k*** is positive integer (indicates the number of facilities). Originally, J J Sylvester [6] has endowed with the graceful and concise geometrical interpretation of underlying problem as: find the smallest circle that encloses a given set of ***n*** points. The centre of this circle is precisely the location of *X*. Under literature this problem is known with various names such as ***Euclidean 1-Centre*** problem. Similarly, Weber has paid attention towards ***1-median*** problem which is known as Weber's problem [7]. Basically, the aim of ***1-median*** problem is to find the best location of a single facility in plane and 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. In the ***k-centre*** problem, the objective is to find a set F' (where $F' \subseteq F$, so as to optimize certain objective function) of $\boldsymbol{k}$ facilities, so that the maximum distance between a demand point and its nearest facility (in F') is minimized.

Besides the ***k-centre*** and ***k-median*** problem, there are so many application-specific facility location problems are discussed in the literature depending on the nature of objective functions. Such as obnoxious facility location problems [8, 9, 10, 11] having the objective function to find a set of undesirable facility location that maximizes the distance between a demand node and facility centre.

Cooper [12] reconsidered the Weber problem with the aim to find a single facility location in order to minimize the distance to customers. He assumed that the location of demand points may be random and bi-variant normal distribution for these locations. Finally, Cooper proved that the objective function is convex with respect to the location chosen and provided an iterative algorithm that solved the first order conditions.

Similarly, Hakimi [13, 14] has established a new property for facility location problem that is known as Hakimi property. It states that there exists an optimal solution to a network location problem in which the facilities are located on the nodes of the network not along the edges. Even though, this property holds only for the minisum problems such as ***P-Median Problem*** (PMP) and ***Uncapacitated Fixed-Charge Location Problem*** (UFCLP). On the other hand, Mirchandani and Odoni [15] have proved that Hakimi property holds for PMP on a network with shortest path travel costs in which the cost of a path may be any concave, non decreasing function of its length. Later on, Mirchandani [16] has analyzed the Hakimi property to determine whether it applies for the stochastic versions of PMP and UFCLP under different assumptions.

Likewise Sheppard [17] has proposed a scenario approach to facility location. Sheppard has focused on to minimize the expected location cost, even though he had not considered the length issues. Thereafter, Mirchandani and Oudjit [18] have presented a rigorous attempt to choose facility locations to minimize the expected cost under scenario based approach. Under this study, they have considered ***2-median*** problem on a tree with stochastic edge lengths described by discrete scenarios.

J. Hershberger [19, 20, 21] has presented a new algorithm to minimize the sum of diameters efficiently. He has divided the given set of demand points in two subsets in order to minimize the sum and maximum length of their distances respectively. Capacitated facility location problem is important variant of facility location problem, where all demand nodes are divided into subsets known as clusters and each cluster is served by the same facility. They supposed $\boldsymbol{n}$ as number of demand points and $\boldsymbol{k}$ as the number of clusters (facilities) respectively, then the size of each cluster may be at most a constant $c \geq \lceil n/k \rceil$. Most of these facility location problems are NP-Hard, if number of facilities (say k) are considered as a part of the input [21].

In the basic ***k-centre*** problem, the set of demand points D is discrete in R^d and the set of facility locations F is the entire R^d plane and the distance function is the Euclidean L_2 or L_∞ metric. This is known to be *NP*-Complete if $d \geq 2$ [21]. For small values of $\boldsymbol{k}$, parametric search technique is a useful tool for solving this problem efficiently. In its decision version, a given radius $\boldsymbol{r}$, the problem is to determine whether D can be covered by the union of $\boldsymbol{k}$ balls of radius $\boldsymbol{r}$ in the discrete ***k-centre*** problem, F is required to be a subset of demand points D. Hwang et al. [29] proposed an $n^{o(\sqrt{k})}$ time algorithm in R^2. A contemporary publication by the same set of authors [30] proposed another algorithm for the discrete version of ***k-centre*** problem in R^2 with the same running time. Therefore it makes sense to search for efficient approximation algorithms and heuristics for the general version [23, 24] and for the basic ***k-centre*** problem in two or higher dimensions considering ***k*** as small and fixed constant [19, 20, 26, 31, 32, 33, 34, 35, 36]

J. Elzinga et al. [37] has proposed first algorithmic solution on the Euclidean ***1-centre*** problem and developed an $O(n^2)$ time algorithm. Thereafter, Shamos and Hoey, Preparata and Shamos [38, 39, 40] have focused on this solution and proposed a new algorithmic solution with improved time complexity of the problem to $O(n \log n)$. Finally, Megiddo [41, 42] has considered the same problem in a situation where the centre of the smallest enclosing circle must lie on a given straight line and applied the prune and search technique to solve the problem. Consequently, an optimal $O(n)$ time algorithm is provided to solve the problem.

So far, much has been done on the unconstrained versions of classical ***1-centre*** problem whereas little has been done in constrained cases of classical ***1-centre*** problem. Then F. Hurtado et al. [43] studied the problem and provided an $O(n+m)$ time algorithm for finding minimum enclosing circle whose centre is constrained to satisfy ***m*** linear constraints with the help of linear programming problem.

In case of the ***2-centre*** problem for the point set P, the objective is to cover P by two closed disks whose radius is as small as possible. To solve the ***planar 2-centre*** problem Sharir [36] has proposed a near-linear time algorithm which actually runs in $O(n \log^9 n)$ time. Chan [32] has proposed a couple of algorithms to solve to solve the problem efficiently. First algorithm of Chan is deterministic and runs in $O(n \log^2 n (\log \log n)^2)$ time, while the second one randomized algorithm that runs in $O(n \log^2 n)$ time with high probability. The variant of this problem is the ***discrete 2-centre*** problem, where the objective is to find two closed disks whose union covers the point set P and whose centers are a pair of points in P. Agarwal et al. [44] have proposed an algorithmic solution for the ***discrete 2-centre*** problem, which runs in $O(n^{4/3} \log^5 n)$ time. Kim et al. [45] considered the ***two-centre*** problem where the demand points are vertices of a convex polygon. They proposed efficient algorithms for both the standard as well as discrete ***2-centre*** problem. These algorithms run on $O(n \log^3 n \log \log n)$ and $O(n \log^2 n)$ time respectively. Ben-Moshe et al. [46] have proposed an improved approach on the upper bound of the continuous/discrete weighted ***k-centre*** problem for a tree. Their study presents linear-time algorithm for the weighted ***2-centre*** problem.

There are various possible variations of the basic facility location model. In case, where it allowed to allocate the facility anywhere in the network, the problem is referred to as ***absolute centre*** problem. Second one is where facility locations are restricted to the nodes of the network then it is known as ***vertex centre*** problem. These types of facility location problem can be either weighted or un-weighted. In the weighted version, a special consideration is given to the distance associated with a demand node and facility while in un-weighted, it is assumed equal for all.

In a study Daskin [1] concluded that for a fixed value of ***k***, the ***vertex k-centre*** problem can be solved in polynomial time. This can be done by evaluating each of the $O(N^k)$ possible combinations of ***k*** facilities locations. Evaluating each of these can be done in polynomial time, though it may take considerable amount of CPU time. Kariv and Hakimi [47] have applied an algorithmic approach to network location problems and shown that for variable values of ***k***, the ***k-centre*** problem is NP-hard.

Various authors have used an auxiliary problem to solve the ***k-centre*** problem such as Set Covering Problem (SCP). Set covering problem referred to find the minimum number of facility locations that can provide specified services to their connected demand points within a specified maximum response time and minimum traveling distance. There are various techniques to solve this problem such as linear programming relaxation with occasional branch and bound applications.

In order to solve the set covering problem, first of all Minieka [48] has suggested a rudimentary algorithm that relies on solving a finite sequence of Set Covering Problems. Minieka has implemented his idea by choosing an initial distance value (threshold distance) as radius and checked it whether all demand points are covered within this radius using no more than ***k*** facilities. The set covering model fails to distinguish between large demand nodes and small demand nodes, all nodes simply have to be covered within the coverage distance. Usually, the number of facilities required to cover all demands within the specified distance is excessively large. In such case, we can relax the requirement that all demands be covered within the coverage distance and locate a fixed number (***k***) of facilities to maximize the number of demands that are covered within the coverage distance. This is the maximal covering problem [49, 50, and 51]. Alternatively, we can relax the coverage distance and locate ***k*** facilities to minimize the endogenously determined coverage distance. This is the ***k-centre*** problem [13, 48, 52, and 53]. Basic idea is to find out the smallest radius that can provide an optimal solution of the considered auxiliary problem and a feasible solution to the ***k-centre*** problem.

Several authors [1, 65, and 66] have outlined that how the ***absolute k-centre*** problem can be solved by considering only a subset of the nodes and local centers on the links. They stated that the objective function value for the optimal solution to the ***absolute k-centre*** problem must be defined by a facility that is equidistant between at least two nodes. A set of nodes can be augmented in a network with $O(n^4)$ additional points that are equidistant between each of the $O(n^2)$ pairs of nodes and that go each of the $O(n^2)$ links, where n is the number of nodes. On the basis of this principle they transformed an ***absolute k-centre*** problem into an equivalent ***vertex k-centre*** problem and found that in either case, the optimal solution can be determined by performing a binary search over the coverage distance. For each coverage distance, a set covering problem is solved. If the number of required facilities is less than or equal to k, the trial coverage distance becomes the new upper bound on the objective function value. Otherwise, the trial coverage distance becomes the new lower bound on the objective function value.

Z. Drezner [54] presented two heuristics and an optimal algorithm that solves problem for a given p in polynomial time n. Drezner assumes that there are two equivalent approaches for formulating and solving the ***p-centre*** problem. The first one is where the location of p new facilities that minimize the maximal weighted distance between each demand node and its closet new facility. While, in second approach there are two phases, phase 1: firstly the set of demand points is partitioned in p disjoint subsets and second phase: to find the best location of new facility for each subset is found. Through this study Drezner has proposed an optimal algorithm which provides two way solutions that either it provides a better solution or there is no better solution.

Similarly, Dyer et al. [55] presented a simple heuristic for determining the p-centre of a finite set of weighted points in an arbitrary metric space. This algorithm runs in $O(np)$ time for an n-point set. They have shown that the ratio of objective function value of the heuristic solution to that of the optimum is bounded by $min(3, 1+\alpha)$ where α is the maximum weight divided by the minimum weight of points in the set.

Hochbaum and Shmoys [23] have presented a ***2-approximation*** algorithm for ***k-center*** problem with triangle inequality. Using linear programming theory, they have provided an interesting insight to problem and enable us to derive in $O(|E|log|E|)$ time a solution with value no more than twice the ***k-center*** problem.

Thereafter, J. Plesnik [24] has generalized the results of Hockbaum and Shmoys, a polynomial algorithm with a worst case error ratio of 2 described for the ***p-center*** problem in connected graphs with edge lengths and vertex weights. A slight modification of this algorithm provides ratio 2 also for the absolute ***p-center*** problem. Both these heuristics are the best possible in the sense that any smaller ratio would imply *P=NP*. Plesnik has set the objective to minimize the weighted eccentricity and provided a polynomial heuristic having ratio 2 also for the ***p-center*** problem. In the subsequent publication Plesnik has proved that the ratio can be arbitrarily large in case of multi-centers. J. Plesnik [25] continued his interest and efforts with ***k-center*** problem. He has given two faster heuristics for the absolute ***p-center*** problem in graphs.

Glozman et al. in [26] studied the problem of covering a set of points by a given number of shapes of some specific kind. For the fixed value of ***k*** most of the problems can be solved in polynomial time but some of them still remain intractable [27, 28].

There are various applications of covering problem in both public as well as private sectors such as emergency facilities, distribution centers, placement of warehouses etc. [56, 57, 58, 59, 60, 61, 62, and 63]. ReVelle and Hogan [64] defined the α ***-reliable P-center*** problem and the maximum reliability location problem for the vehicle placement. The α **-reliable *P*-center** problem finds the smallest coverage distance such that all demands are covered with probability α by an available vehicle. The maximum reliability location problem is to find the locations of P facilities such that the reliability α is maximized. This can be solved by fixing a feasible value of α and then solving the problem.

Wang and Cheng [72] presented a study on NP-hard ***k-center*** problem. They chosen $\boldsymbol{k}$ vertices as service centers in such a way that the maximum weighted service delivery distance to any vertex is minimized and a special consideration is given to the vertex weight. They implemented a greedy strategy to choose the vertex with maximum vertex weight as the next service vertex. They proved that the proposed algorithm generates results that are guaranteed to be no greater than twice the optimal solution values. They made the comparison of time complexity of greedy strategy on uniprocessor and multiprocessor systems. They found that the time complexity on uniprocessor is $O(n^3)$ and multiprocessor system is $O(n \log^2 n)$, where n is the total number of vertices. They show inspite of having unlimited parallelism the algorithm has higher than poly logarithmic time complexity. Thereafter, Chen and Handler [73] produced some significant results on the conditional ***p-center*** problem in the plane. They developed an algorithm for the conditional ***p-center*** problem namely, the optimal location of one or more additional facilities in a region with given demand points and one or more preexisting facilities. The solution dealt with this involves the minimax criterion and Euclidean distances in two dimensional spaces. They have generalized the present conditional case of a relaxation method previously developed for the unconditional ***p-center*** problems. Interestingly, its worst case complexity is identical to that of the unconditional version and in practice, the conditional algorithm is more efficient.

Hochbaum and Pathria [74] have provided a new study with a generalized version of the ***p-center*** problem. The $\boldsymbol{p}$ servers to be selected are partitioned into $\boldsymbol{k}$ sets and the number of servers selected from each set must be within a specified range. They refer to these problems as the '***Set***' ***p-center*** problems. They established that the triangle inequality versions of these problems in which the edge weights are assumed to satisfy the triangle inequality are also NP-complete. They also provided a polynomial time approximation algorithm for the two triangle inequality ***Set p-center*** problems that is optimal for one of the problems in the sense that no algorithm with polynomial running time can provide a better constant factor performance guarantee unless P=NP.

Similarly S. Chaudhuri et al. [75] has discussed the ***k-center*** problem with triangle inequality where placing ***k-center*** nodes in a weighted undirected graph in which edge weights obey the triangle inequality so that the maximum distance of any node to its nearest center is minimized. In this study they consider three different versions of this reliable ***k-center*** problem depending on which of the nodes can serve as centers and non centers and derive best possible approximation algorithms for all three versions.

Shmoys et al. [76] presented an approximation algorithm for the several facility location problems. They considered the case where the distance between locations are non negative, symmetric and satisfy the triangle inequality. For the un-capacitated facility location, they provided a polynomial time algorithm that finds solution of cost within a factor or 3.16 of the optimal. The results provided are based on filtering and rounding technique of Lin and Vitter [91].

Based on Minieka's idea, Daskin [1] has solved the ***vertex p*-center** problem using set covering model for locating centers and bisection method to find the optimal coverage distance. Although, this algorithm produced wonderful results but the algorithm consumed considerable amount of CPU time. Further the algorithm could resolve the problem in polynomial time only for the fixed values of *P*. Though, ***vertex p-center*** problem is NP hard for the variable values of *P*. Daskin [67] have introduced a new approach to solve the ***vertex p-center*** problem to optimality. Now he has made a few modifications in his previous algorithm like replacement of the set covering model (or set covering problem SCP) with maximal covering model and applied Lagrangian relaxation to solve problem optimally. Daskin has shown how performing a binary search over the maximum distance can solve the problem. For each maximum distance, it solves a maximal covering problem. Similarly Elloumi et al. [68] has applied greedy heuristic and the IP formulation of the sub problem of Minieka's to solve the same problem.

On the other hand, Kamal Jain et al. [77] has presented a simple and natural greedy algorithm for the metric un-capacitated facility location problem with an approximation guarantee of 1.61.

Inspired by an algorithm due to Minieka, Ilhan and Pinar [69] have developed a simple and yet very efficient exact algorithm for the problem of locating *p* facilities and assigning clients to them in order to minimize the maximum distance between a client and the facility assigned. After a lower bounding phase, the algorithm iteratively sets a maximum distance value within which it tries to assign all clients and thus solves integer feasibility sub-problems. They have reported very good computational results on a set of 84 test problems derived from OR-Lib and TSP-Lib problem instances with up to 900 vertices solved to optimality for the first time.

Al-Khedhairi [70, 71] had put forward several improvements to the well-known Daskin's and Ilhan's algorithms. The intention behind these modifications is to reduce the number of iterations needed to find the optimal solution as well as to shrink the gap between lower and upper bounds to solve the ***vertex p-center*** problem. The modified algorithm has shown slightly better results.

D. Chen and Reuven Chen [90] 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.

Facility location problem is one of the typical problems as discussed above and various authors have contributed to solve the problem efficiently in both computer science and operations research fields. In the dynamic environmental where some basic factors changes such as population shifts, market trends evolve and the evolution of new planning challenges justifies the dynamic facility location for relocating, expanding and adapting facilities. However, with the passage of time and occurrence of new development in the concept of facility location problems, ***Static Facility Location*** (***SFL***) models are not necessarily operable and practical. Therefore, keeping in view the long term behavior of facility location problems, models must consider some aspects of future uncertainty in order to be applicable and ready to implement.

In fact, not only the main parameters and criteria in each problem are prone to change during the time planning horizon but also a considerable amount of investment and capital is required for developing and obtaining a new facility so that facility decisions are particularly costly and time sensitive. Consequently, it is expected to perform as beneficial as possible in a long term planning horizon. Therefore, in order to cope with fluctuations, ***Dynamic Facility Location*** (***DFL***) problems seem to be quite indispensable in which cost and time are two main criteria affecting the concerned models at most.

Different authors have paid attention towards the dynamic facility location models and proposed various algorithms to solve the problem. D. Fotakis [79] has presented a study on incremental algorithm for facility location problem and ***k-median***. This algorithm guarantees constant performance ratio using $O(k)$ medians. The novel merger rule is applied on this algorithm. Similarly T. Gonzalez [80] has given attention towards incremental versions of facility location problems and provided a ***2-competitive*** algorithm. Lin et al. [81] have given competitive algorithms for incremental ***k-vertex, k-set cover, k-median, k-spanning tree*** by applying a general framework for cardinality constrained problem to study all these problems. Mettu and Plaxton [82] also contributed to incremental un-capacitated ***k-median*** approach and provided a 29.86 competitive algorithm.

Plaxton [83] provided an (4+ε) *a*-competitive algorithm for incremental facility location problem and gave *a*-approximation algorithm for the un-capacitated facility location ensuing in a 12.16-competitive algorithm. Charikar et al. [84] have developed a deterministic ***8-competitive*** algorithm for incremental clustering and also given a ***2e-competitive*** algorithm for hierarchical ***k-center*** problem. Afterwards, Dasgupta et al. [85] have intensively worked on hierarchical clustering and implemented an idea of finding competitive hierarchical clustering for hierarchical ***k-center*** problem.

The recent literature has strayed from conventional approach to deal with more realistic situations where the input changes over time. Jeff and Alexa [86] have introduced a general model for converting conventional algorithms into the incremental algorithms with only a constant factor loss in approximation power. On the other hand, the mobile facility location problem is one of the prominent issues for the researchers to solve. Zachary Friggstad and R. Salavatipour [87] have worked in order to minimize the movement in mobile facility location problem.

Richard Matthew and Samir Khuller [88] further contributed in the area of streaming algorithms. They have developed first streaming algorithm with constant factor approximation to the cluster radius for two variants of the ***k-***

center clustering problem. A streaming (4+ ε)-approximation algorithm using $O(\varepsilon^{-1}kz)$ memory for the problem with outliers. They have also given another streaming (6+ ε)-approximation algorithm using $O(\varepsilon^{-1} \ln(\varepsilon^{-1}) k+k^2)$ memory for a variation motivated by anonymity consideration where each cluster must contain at least a certain number of input points. M. Albareda-Sambola et al. [89, 126] have introduced a new approach that is the multi-period incremental service facility location problem. The basic idea is to set a number of new facilities over a finite time horizon so as to cover dynamically the demand of a given set of customers.

A critical look at the literature presented above reveals that an effective approach is required to achieve an optimal solution of facility location problem. Therefore, this work aims to achieve an optimal solution for above stated problems by proposing an improved approach for ***vertex k-center*** problem and ***incremental mobile facility location problem***.

Chapter 3

Facility Location Models and Techniques

3.1 Facility Location Models

Facility location models are used in a wide variety of applications. Examples include locating warehouses within a supply chain to minimize the average travel time to the markets, locating hazardous material sites to minimize exposure to the public, locating railroad stations to minimize the variability of delivery schedules, locating automatic teller machines to best serve the bank's customers, and locating a coastal search and rescue station to minimize the maximum response time to maritime accidents [92].

Facility location models [5] concern the provision of a service to satisfy a spatially dispersed demand. A demand for the service exists at a large number of widely dispersed sites. It is impossible to provide the service anywhere. For instance, every household needs a source of groceries but impossible to provide a grocery store at each household. Therefore, for reasons of cost, the service must be provided from a few centralized locations. Facility location problems have proven to be a fertile ground for operations researchers interested in modeling, algorithm development, and complexity theory.

Applications of location modeling include locating Emergency Medical Service (EMS) bases, fire stations, schools, hospitals, reserves for endangered species, airline hubs, waste disposal sites, and warehouses to list only a small subset of the numerous areas in which location models have been applied. Location models have also found applications in nontraditional areas, including medical diagnosis, vehicle routing, alignment of candidates and parties along a political spectrum, and the analysis of archeological sites [93]. Location theory and modeling has its roots in the pioneering work of Weber [7] who considered the problem of locating a single facility to minimize the total travel distance between the site and a set of customers. Location models are application-specific that is uniquely built on their defined objectives and constraints. There does not exist a general location model that is suitable for all potential applications. Although, there are several ways of subdividing the broad spectrum of location models [49] as shown in Figure 3.1.

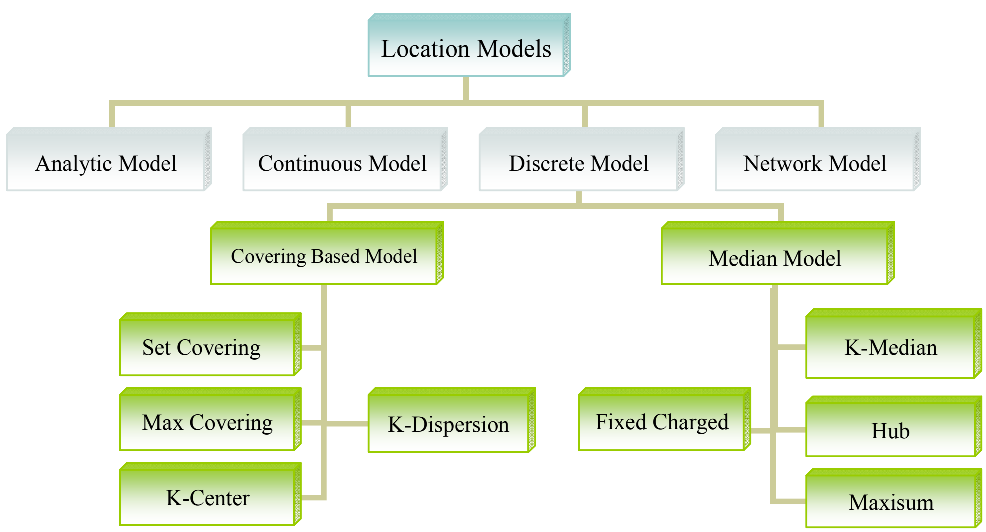


Figure 3.1: Breakdown of location models

3.2 Basic Facility Location Models

There are varieties of facility location models that have been widely researched. This section highlights the several basic facility location models.

These models are generally used in maximum distance problems for example; an automobile company wants to open the workstation (service stations) to facilitate their customers of any given region. The motive is to provide the services to all customers within equal and minimum traveling distance.

The basic eight facility location models are as under:

i) Set Covering
ii) Maximal Covering
iii) K-Center
iv) K-Dispersion
v) K-Median
vi) Fixed Charge
vii) Hub
viii) Maxisum

The general problem is to locate new facilities and optimize the objective function. Distance is the fundamental entity of objective function in such type of problems. Therefore if we want to classify these models on the basis of distance then first four are based on maximum distance and remaining four are based on total/average distance.

3.3 Maximum Distance Models

3.3.1 Set Covering Location Models

According to our literature survey, in category of location problems, first location problem was the Set Covering Problem (**SCP**) discussed introduced by Toregas et al. [49]. The objective was to locate the emergency services in a given region to facilitate all inhabitants. The set covering problem was formulated as given under:

Minimize $$\sum_{j \in J} p_j$$

Subject to: $$\sum_{j \in N_i} p_j \geq 1 \qquad \forall i \in I$$

$$p_j \in \{0,1\} \qquad \forall j \in J$$

Where I = is a set of demand nodes indexed by i.

J = a set of candidate facility locations, indexed by j.

N_i = a set of all candidate locations that can cover demand points i and decision variable is given below:

$$p_j = \begin{cases} 1 & \text{if we located at site } j \\ 0 & \text{otherwise} \end{cases}$$

The objective is to minimize the number of facilities located with assurance of the each demand node is covered by at least one facility.

3.3.2 Maximal Covering Location Models

Simple set covering problem does not impose budget constraints, rather the basic aim is to cover all demand nodes without any conditions. Undoubtedly, budget plays a very crucial role in many facility planning situations. Since sometimes, it's difficult to cover all demand nodes with limited number of facilities. While the set covering problem rule out the budget factor. Church and ReVelle [50] have formulated the maximal covering location problem (**MCLP**) to address planning situations in which have an upper limit on the number of facilities to be sited. The main objective of maximal covering location problem is to locate a predetermined number of facilities (say k), in such a way that the demand to be covered can be maximized. Therefore, the MCLP assumes that there may not be enough facilities to cover all of the demand nodes. If not all demand nodes can be covered, then the model seeks the siting scheme that covers the most demand.

The maximal covering location problem can be formulated as given under:

Maximize $\sum_{i \in I} p_i z_i$

Subject to: $\sum_{j \in N_i} s_j - z_i \geq 0 \qquad \forall i \in I$

$\sum_{j \in J} s_j = k$

$s_j \in \{0,1\} \qquad \forall j \in J$

$z_i \in \{0,1\} \qquad \forall i \in I$

Where p_i = demand node at i

k = the number of facilities to locate

$z_i = \begin{cases} 1 & \text{if demand node } i \text{ is covered} \\ 0 & \text{otherwise} \end{cases}$

The objective function is to maximize the total demand covered. In many cases, where there is a desired coverage distance and some maximum distance beyond which facility (service) is undesirable. It is assumed that covering distance is a fixed or predetermined standard in both set covering location problem as well as maximal covering location problem. Undoubtedly, it's true in many cases but in some situations distance is assumed as a goal or target rather than a fixed standard.

3.3.3 k-Center Problem

The objective of ***k-center*** problem is to minimize the maximum distance between a demand node and its nearest facility (facility center). The numbers of facilities are pre determined. There are several possible variations of the basic model. The ***vertex k-center*** problem restricts the set of candidate facility sites to the nodes of the network, while the ***absolute k-center*** problem permits the facilities to be anywhere along the arcs. ***Vertex k-center*** problem can be represented as follow:

Let G = (*V*, *E*) is a weighted graph having $V = \{v_1, v_2, v_3 \ldots v_n\}$ as a set of $\boldsymbol{n}$ vertices and $E = \{e_1, e_2, e_3 \ldots e_m\}$ as a set of $\boldsymbol{m}$ edges. For any two vertices v_i and v_j in *V* the distance $d(v_i, v_j)$ is defined to be the length of the shortest path linking v_i and v_j. The objective is to find a subset *S* of *V* which implies that $|S| \leq K$, where *K* is the number of facilities to be identified.

$$\textbf{Minimize} \quad \max_{v \in V} d(v, S)$$

For fixed values of ***k***, the ***vertex k-center*** problem can be solved in $O(N^k)$ time, since we can enumerate each possible set of candidate locations in this amount of time. Even for moderate values of *N* (nodes) and ***k*** such enumeration is not realistic and more sophisticated approaches are required. For variable values of ***k***, the problem is NP-hard [94, 125].

3.3.4 k-Dispersion Problem (KDP)

So far, we have discussed the models revealing importance of distance between clients and facility centers. The major concern was to locate facilities in such a way so that the facility should be as near as possible to the clients. Kubey [95] addressed the ***k-dispersion*** (or *p-dispersion*) problem and stated that the ***k-dispersion*** problem is different than the problems discussed till now. Since, it concerned only with distance between facilities and having the main objective to maximize the minimum distance between any pair of facilities. Locating franchise outlets such as McDonald restaurants are the potential applications of *k-dispersion* problem. Since in such cases, it is desirable to reduce the mutual competition among the restaurants. To formulate the model, it requires an additional input (P) and a decision variable Q.

P = a large constant e.g., $\underset{i \in I, j \in J}{Max}\{d_{ij}\}$

Q = the minimum separation distance between any pair of the facilities

Maximize Q

Subject to:

$$\sum_{j \in J} s_j = k$$

$$Q + (P - d_{ij})s_i + (P - d_{ij})s_j \le 2P - d_{ij} \qquad \forall i \in I, j \in J$$

$$x_j \in \{0,1\} \qquad \forall j \in J$$

The objective function maximizes the distance between the two closest facilities. K facilities are required to be located. Constraints defines that the minimum separation between any pair of open facilities.

3.4 Sum or Average Distance Models

The main objective of average distance models is to reduce the total traveling distance between the facilities and the demand nodes. It is applicable in both private as well as public sector. Life saving services such as public health centers, fire stations, and many more services that is dedicated in society welfare should be located in such a way that can be availed well in time. This approach may be viewed as an efficiency objective.

3.4.1 *k*-Median Problem

The ***k-median*** problem is useful to model many real world situations such as the location of public facilities or industrialization (e.g. warehouses). Hakimi [13, 14] presented a study based on locations of ***k*** facilities to minimize the demand-weighted total distance between demand nodes and the facilities to which they are assigned. Daskin [1] has formulated the *k-median* problem as under:

Minimize $$\sum_{i \in I} \sum_{j \in J} p_i d_{ij} t_{ij}$$

Subject to: $$\sum_{j \in J} s_j = k$$

$$\sum_{j \in J} t_{ij} = 1 \qquad \forall\, i \in I$$

$$t_{ij} - s_j \leq 0 \qquad \forall i \in I, j \in J$$

$$s_j \in \{0,1\} \qquad \forall j \in J$$

$$t_{ij} \in \{0,1\} \qquad \forall i \in I, j \in J$$

Where

d_{ij} = distance between demand node i and candidate sit j

$$t_{ij} = \begin{cases} 1 & \text{if demand node } i \text{ is assigned to facility at site } j \\ 0 & \text{otherwise} \end{cases}$$

The objective function minimizes the demand-weighted total distance traveled. It is also assumed that the facilities are located at nodes of the network. Whereas Hakimi [13] has proved that if these facilities are located at edges (arcs) of the network then, it would reduce the total travel distance. Likewise the ***k-center*** problem, ***k-median*** problem can be solved in polynomial time for fixed values of ***k***, but NP-hard for variable values of ***k***.

3.4.2 Fixed Charge Location Problem

A comprehensive study of ***k-median*** problem reveals that it generally works on three basic assumptions that are not appropriate for certain siting scenarios. Assumptions are as given under:

- ✓ Same cost for siting facility at each potential site.
- ✓ Facilities do not have capacities on the demand that they can serve.
- ✓ A fixed number of facilities to open.

The fixed charge location model [1] relaxes all three assumptions. The objective of fixed charge location model is to minimize the total facility and transportation cost. It also determines the optimal number of location of facilities and the assignment of demands to those facilities.

w_j = fixed cost of locating a facility at candidate site j.

D_j = capacity of the facility at candidate site j.

β = cost per unit demand per unit distance.

The capacitated fixed charge location problem can be formulated as follows:

Minimize $$\sum_{j \in J} w_j s_j + \beta \sum_{i \in I} \sum_{j \in J} p_i d_i t_{ij}$$

Subject to: $$\sum_{j \in J} t_{ij} = 1 \qquad \forall i \in I$$

$$t_{ij} - s_j \le 0 \qquad \forall i \in I, j \in J$$

$$\sum_{j \in J} p_i t_{ij} - D_j s_j \le 0 \qquad \forall i \in I$$

$$s_j \in \{0,1\} \qquad \forall j \in J$$

$$t_{ij} \in \{0,1\} \qquad \forall i \in I, j \in J$$

The objective function minimizes the sum of the fixed facility location costs and the total travel costs for demand to be served. The second set of terms in given objective function is often referred to as demand weighted distance. The constraints prohibit the total demand assigned to a facility from exceeding the capacity of the facility D_j .

3.4.3 Hub Location Model

There are many potential applications based on hub location model. Many logistics systems such as a less than truckload carrier network, airline network, employ hub and spoke systems. These systems are designed to utilize larger capacity or faster vehicles or modes over the long-haul portion of an origin to destination delivery. Accordingly, these systems reduce average per mile transportation cost or total delivery time. Various models [96, 97, 98, 99] have been formulated to locate the hub and delivery routes of hub and spoke systems. The main objective of all these models was to minimize total cost. The ***k-hub location*** model can be formulated using following notation inputs

p_{ij} = number of units of flow between node *i* and *j*.

d_{ij} = unit cost of transportation between node *i* and *j*.

β = distance factor for transport between hubs

And the following are the decision variables

$$S_{ij} = \begin{cases} 1 & \text{if a hub is located at node j} \\ 0 & \text{otherwise} \end{cases}$$

$$t_{ij} = \begin{cases} 1 & \text{ifdemands at node i are assigned to a hub located at node j} \\ 0 & \text{otherwise} \end{cases}$$

Minimize

$$\sum_{i \in N} \sum_{j \in N} p_{ij} \Big(\sum_{k \in N} d_{ik} t_{ik} + \sum_{m \in N} d_{jm} t_{jm} + \beta \sum_{k \in N} \sum_{k \in N} d_{km} t_{ik} t_{jm} \Big)$$

Subject to:

$$\sum_{j \in N} s_j - k$$

$$\sum_{j \in N} t_{ij} = 1$$

$$t_{ij} - s_j \le 0 \qquad \forall i \in I, j \in J$$

$$s_j \in \{0,1\} \qquad \forall j \in J$$

$$t_{ij} \in \{0,1\} \qquad \forall i \in I, j \in J$$

The objective function minimizes the sum of the cost of moving items between a non-hub node and the hub to which the node is assigned, the cost of moving from the final hub to the destination of the flow, and the inter-hub movement cost which is discounted by a factor of β . The model assumes that the hub portion of the network is a complete graph and therefore flows between any pair of nodes *i* and *j* will pass through at most two different hub nodes.

3.4.4 The Maxisum Location Model

The objective of maxisum location model is to determine the locations of ***k*** facilities in such a way that total demand-weighted distance between demand nodes and the facilities to which they are assigned is maximized. The potential applications of maxisum model include the location of prison, power plants and solid waste repositories. It includes the location planning situations, where we have to locate undesirable facilities having at least one objective to locate facilities far from the demand nodes. The model may be formulated as follows:

Maximize $$\sum_{i \in I} \sum_{j \in J} p_i q_{ij} t_{ij}$$

Subject to: $$\sum_{j \in J} s_j = k$$

$$\sum_{j \in J} t_{ij} = 1 \qquad \forall i \in I$$

$$t_{ij} - s_j \le 0 \qquad \forall i \in I, j \in J$$

$$\sum_{k=1}^{m} t_{ik_i} - s_{m_i} \geq 0 \qquad \forall i \in I, m = 1,....,N-1$$

$$s_j \in \{0,1\} \qquad \forall j \in J$$

$$t_{ij} \in \{0,1\} \qquad \forall\, i \in I\, ,\, j \in J$$

This formulation is similar to that of *k*-median problem with two notable exceptions. First, the objective is to maximize the demand weighted total distance and not to minimize it. The unfortunate impact of objective is that it forces demands to be assigned to the most remote facility.

3.5 Location Routing Models

The location-routing model is applicable in location planning situations where distribution centers provide collection and / or distribution functions in which demand is served by multiple drop off and / or pick up routs. The overall effectiveness of such facility sites depends upon the efficiency of vehicle routes needed to serve multiple demands. Various authors [100, 101, 102] have highlighted that the location-routing problems involve three inter related fundamental decisions as given under:

- ✓ Where to locate the facility?
- ✓ How to allocate customers?
- ✓ How to route the vehicles to serve customers?

Therefore the objective function for location-routing problems is somewhat complex due to having multiple decision variables.

3.6 Facility Location-Network Design Models

In the facility location problems it is very important to design its network. Although, in our previous models it was assumed that the network is already given. However in so many problems it is necessary to determine which path should be included in network as well as which location is appropriate for the facility allocation. The potential applications of facility location network design model are electricity distribution systems, computer networks and railway tracks etc. Current and Pirkul [103, 104] have introduced a model for location situations where the desired network is a path and the facilities are entry locations such as stations for demand to enter the path. The objective of model was to minimize the cost of facility and network as well as the cost of arcs needed for demand to reach a facility on the path.

3.7 Multi-Objective Models

There are many facility location models designed to handle multiple objectives. Such problems are having several decision variables which are responsible to select the objective function in different aspects. The final decision may be made on political or pragmatic considerations that are not part of the considerations in the formal analysis. As a result, the decisions can very often be far from the optimal [105].

Cohon [106] has provided in a study that such models can be solved using two approaches as given under:

- ✓ Generating techniques.
- ✓ Preference based techniques.

Generating techniques identify the Pareto-optimal siting of the facilities from the preferences of the decision makers. Preference-based techniques use a ranking method to rank the objectives and then find the solution that optimizes the ranking. Ranking may be done through the use of simple weighting or complex analytical methods.

Multiple distance-related objectives have been developed in some studies to highlight the importance of distance. Schilling [107] included several different maximum covering objectives for fire equipment locations. Church [108] employed a maximum-distance objective and an average-distance objective. Multiple objectives could also include objectives related to cost, risk and the equity of risk.

3.8 Dynamic Location Models

The basic facility location models have not considered the time factor during location analysis, that's why they may be called as *static* facility location models [110, 111]. *Dynamic* models incorporate the time. Current et al. [112] have defined two categories of dynamic models.

- ✓ Implicitly dynamic model
- ✓ Explicitly dynamic model

Basically *implicitly* models are *static*, because all facilities are opened at one time and remain open over the planning horizon. Whereas, the *explicitly* (dynamic) models are those designed for problems where the facilities will be opened over time.

3.9 Stochastic Location Models

All of the basic facility location models presented till now assumes that all parameters of problem are given, in other words there is no uncertainty. Even the dynamic models discussed also assume that the changes over time are known with certainty. However, there are many cases in facility location problems where some parameters of problem are not known with certainty. Hence there exits a considerable uncertainty and such location models are known as ***stochastic location models***. Various authors [15, 16, 109, 113, 114, and 115] have defined several uncertain parameters for the stochastic problems such as demand, travel time, availability of service and the number of facilities to be sited.

The comprehensive literature survey reveals that there are basic four approaches to deal with stochastic problems. The first approach approximates the uncertainty via a deterministic surrogate; Bean et al. [116] formulated the problem and addressed the "*demand*" parameter. The second approach develops chance constraint models [117]. And Daskin [115] has formulated a probabilistic extension of the maximal covering problem in which facilities are assumed to be busy with probability ρ . The main objective of these models is to maximize the number of demands that are covered by an available facility.

The third approach explicitly accounts for the queuing interactions that occur in a spatially distributed queuing system with facilities at multiple locations in a network. The fourth approach is used for scenario planning [117, 119]. Scenarios represent possible values for parameters that may vary over the planning horizon. Initially, Sheppard [17] has discussed the applications of scenario planning to facility location problem, which minimized the expected cost over all scenarios.

3.10 Techniques Used to Solve K-Center Problems

This section describes about different approaches used to solve the facility location problem. 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 parts of this section.

3.10.1 Branch and Bound Method

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.

3.10.2 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 [120]. 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.

3.10.3 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.

3.10.4 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 [1]. 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.

3.10.5 Interchange Heuristic Approach

This interchange heuristic approach is introduced by Teitz and Bart [121]. 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, then 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 ***meta-heuristic***. The basic idea behind this is to break out local optima and search other regions of the solution space.

One of the earliest meta-heuristic 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.

3.11 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. J. Mihelic and B. Robic [122] 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 $\boldsymbol{k}$. 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. J. Mihelic and Borut Robic [122] 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 $\boldsymbol{k}$. Even though the greedy plus version runs much slower because it tries all the vertices for ***1-center***.

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* increases, the performance of algorithm goes slower and results obtained are away from the optimum solution.

So, keeping in view the performance of all techniques discussed above. It may be suggested that it is required to develop an algorithm which can improve the performance with increased number of nodes (***k***).

Chapter 4

Anticipatory Bound Selection Procedure (ABSP) for Vertex *k*- Center Problem

4.1 Introduction

Vertex k-center problem introduces the notion to recognize k locations as centers in a given network of n connected nodes holding the condition of triangle inequality. This paper presents an efficient algorithm that provides a better solution for vertex k-center problem. Anticipatory Bound Selection Procedure is deployed to find the initial threshold distance (or radius) and eradicating the gap between minimum distance (lower bound) and maximum distance (upper bound). The Jump Based Scheme is applied to find the optimal coverage distance. The foremost feature of this algorithm is to provide an optimal solution with lesser number of iterations so that the performance of algorithm can be improved.

This chapter is structured as follow. Section 4.2 provides the formulation of the problem. Section 4.3 describes the algorithm presented. Section 4.4 provides details of methodology employed. Section 4.5 presents the results obtained and section 4.6 provides analytical study of presented work as well as discussion on various aspects of it. Section 4.7 presents an algorithm for obnoxious facility location problem and employed methodology.

4.2 Formulation of Vertex k-Center Problem

As the objectives have been stated clearly in earlier section; now it is required to formulate the ***vertex k-center*** problem in adequate manner; it's indispensable to presume basic entities listed as below:

X and Y be the sets of demand nodes and candidate sites respectively

$$X = \{1, ...N\}$$

$$Y = \{1, ..., M\}$$

d_{xy} = is the distance between demand node and candidate site. $x \in X$, $y \in Y$

K = Total number of facilities to be identified.

$$F_y = \begin{cases} 1 & \text{if the facility is located at site y} \in Y \\ 0 & \text{otherwise} \end{cases}$$

A_{xy} = 1 if demand node $x \in X$ is assigned to candidate site $y \in Y$, where F_y =1, otherwise

A_{xy} = 0

H = the maximum distance between a demand node and its nearest facility in a given network.

Now the formulation of ***vertex k-center*** problem can be presented as given below:

$$\textbf{Minimize} \quad H \tag{1}$$

Subject to:

$$\sum_{y \in Y} A_{xy} = 1 \qquad \forall x \in X \tag{2}$$

$$A_{xy} \leq F_y \qquad \forall x \in X, y \in Y \qquad (3)$$

$$\sum_{y \in Y} F_y = K \qquad (4)$$

$$\sum_{y \in Y} A_{xy} d_{xy} \leq H \qquad \forall x \in X \qquad (5)$$

Where A_{xy} , F_y can have only binary values such as:

$$A_{xy}\,,\ F_y = \{0,1\} \qquad \forall x \in X, y \in Y$$

And also known as integrity constants.

Equation (1) characterizes the objective function that is to minimize the maximum distance between a demand node and its nearest facility. Subject to 1st constraint as in equation (2), that guarantees that each demand node should be assigned to exactly one facility. Although, 2nd constraint provides a check on each demand node that it should be assigned to only selected candidate site as in equation (3). The 3rd constraint ensures that the total number of facilities located must be equal to *K* as in equation (4). Finally the 4th constraint stipulates that the maximum distance between any demand node and facility should be less than *H* as in equation (5).

4.3 Description of Algorithm

The existing algorithm due to Daskin [4] proposes to solve the vertex *p-center* problem using the bisection (binary search) method to find the optimal distance value. But the practical implication of this method is that it generates a swing in results. Whereas the presented algorithm using a new approach that gives the desired results in lesser number of iterations. Complete steps of proposed algorithm are given as below.

Algorithm 4.1 *Anticipatory Bound Selection Procedure*

Start
Input: A[ ][ ], K;
//where A is distance matrix and K is the number of facility locations to be identified.
1: Initialize *Max* [ ] = {α_1, α_2, α_3, ...α_n ; $\forall \alpha \in A$ };
//where A is distance Matrix and every α represents the maximum distance value from each row.
2: Sort *Max*[]; //Keep unique values only
3: Set ***L***= *Max*[0]; *i*=0; //Assignment of Lower (*L*) & Upper (*U*) Bounds
U= *Max*[*m*]; *i*=*m*; //Where *i*=0,1,2…………………*m*
4: Set *D*=*L*, *i*=0; //where *D* in initial threshold distance.
5: Solve SCP(D);
If *p* > *K* **then** // *p* represents the number of facility locations found with distance *D*.
{Set *D* = *Max*[*i*+2]
Go to Step 5;
}
Else
If (*p* < *K*) **then**
{
Set *U* = *D*, *L* = *D*-1, *D* = *L*,
Go to Step 5;
}
6: Set *temp*=*D*, *L*=*D-1*, *U*=*D and D*=*L*.; //if *p*=*K*
Solve SCP(D); // *temp* is variable used to store value of optimal covering distance.
If *p* > *K* then // infeasible Solution
{**Stop**}
Else
{Go to Step 6**}** //Solution feasible

Output: ***D***, *p* //***D*** is optimal distance value and ***p*** is the number of facilities

First step of algorithm represents the initialization of an array named as *Max*[]. This array contains all distance values collected from each row of distance matrix as the maximum value of respective row. Every α represents $\underset{1 \le i \ge N}{Max}\ d_i^j$ of distance matrix *A*. ***D*** represents initial threshold distance. ***L*** and

U represent lower and upper bounds respectively. The variable *temp* keeps the value of *D* temporarily.

Second step is used to sort the elements of *Max*[] in ascending order. To sort the array, the quick sort method is used. It is set of selected maximum values of distance matrix that helps in smooth search of optimal distance as well as reduces the search efforts unlikely to binary search.

Third step is used to initializes the value of lower and upper bounds. Lower bound is initialized with the minimum value in *Max*[] such as *L*= *Max*[0], and upper bound is initialized with maximum value in *max*[], e.g. *U*=*Max*[m] (where i=1, 2…….*m*). The fourth step is used to assign initial threshold distance D = *L*.

The fifth step is used to solve Set Covering Problem with the help of a subroutine SCP(), *D* is the initial threshold distance that is passed as parameter to the SCP(). Thus it make us able to create required number of clusters on given network, each cluster is having a particular node as candidate site (site may be facility) having maximum distance *D* to all demand nodes. Consequently SCP() function returns ***p***, that is required number of facilities found in network according to distance *D* to all demand nodes. Since *K* is the number of facilities to be allocated, therefore *p* will be compared with *K*.

If ***p*** > *K* then it is not a feasible solution (since we have fixed number of facilities that is *K*). Therefore we have to create clusters with next available distance value (jump from next to next distance value in *Max*[] as suggested by jump based scheme) and set *D*=*M*ax[*i*+2] (initially *i*=0, now *Max*[2]) repeat step 5 until it is feasible. If *p* < *K* then set *U*=*L*, *L*=(*D*-1), *D*=*L* and repeat step 5.

There may be third case i.e. if $p = K$, it shows that the solution is feasible for the distance value D then set $temp$=D, U=L, L= (D-1), D=L and go to the step 6.

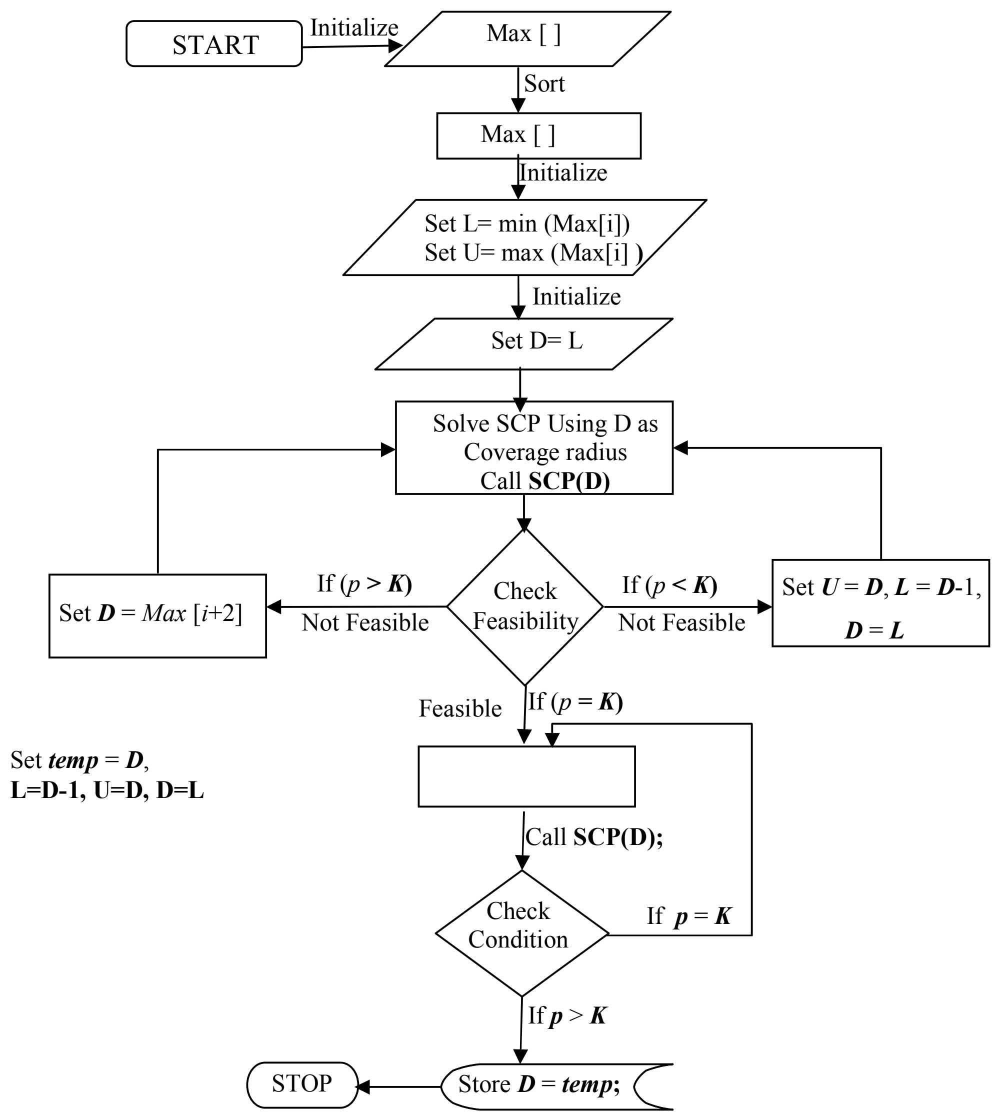


Figure 4.1: Flow Diagram of Proposed Algorithm

Step 6 checks the feasibility at one step backward, Set D= (D-1) and perform SCP(), if $p > K$ then not feasible, Stop and $temp$ is the optimal coverage distance else $temp$=D and $temp$ is the new coverage distance. Thus the presented algorithm provides an improved and effective way to obtain optimal coverage distance to allocate *k*- facilities. The complete flow of algorithm is shown through a flow diagram in Figure 4.1.

4.4 Methodology Employed

This section provides an intact insight to the comprehensive description of the entire models and approaches deployed; basically we have used two approaches namely Anticipatory Bound Selection Procedure and Jump Based Scheme to provide an effective and efficient way to solve the problem undertaken.

4.4.1 Anticipatory Bound Selection Procedure (ABSP)

The Anticipatory Bound Selection Procedure means to anticipate the lower bound or the initial threshold distance before starting any proceeding. The presented approach has used ABSP; consequently it reduces the efforts to determine initial threshold distance. Now, there may be a question in your mind that, why we need the ABSP method?

The critical literature survey reveals that most of the algorithms are using bisection method to compute the initial threshold distance. It also presumes zero as lower bound (minimum distance) as shown in Figure 4.2. While the bisection method has a very serious problem that it generates simple harmonic motion. Additionally, it also generates such distance values that doesn't exist in distance matrix which takes extra iterations to reach the optimal distance value. Eventually, it increases the number of iterations as well as decreases the gap between lower and upper bounds gradually. Hence it's required at the moment to design such a procedure that can generate an initial threshold distance which directs the algorithm to move in either forward or backward direction or nearby the optimal coverage distance as well as the distance values should be selected from distance matrix. To overcome all these shortcomings ABSP is designed.

The ABSP reduces the substantial gap between lower bound and upper bound (lower bound may be defined as the lowest distance value closer to optimal distance value, on the other hand upper bound is a highest distance value closer to optimal distance value) by computing a distance value (i.e. lower bound), that is neither the zero nor the mean value as shown in Figure 4.3.

ABSP provides a set of distance values which are collected in *max*[], further in which sequence these distance values will be passed to SCP() as initial threshold distance is managed by JBS as shown in Figure 4.4.

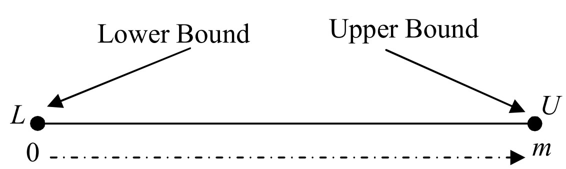


Figure 4.2: Initial Position of Lower /Upper Bound as per Bisection method

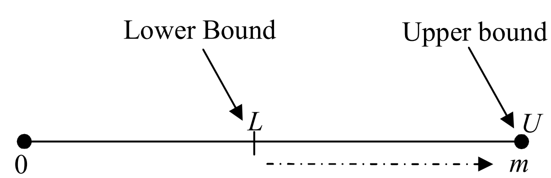


Figure 4.3: Initial Position of Lower and Upper Bound as per ABSP

4.4.2 Operational Details of ABSP

The anticipatory bound selection is a two step procedure employed to shrink the gap between lower and upper bounds. This procedure helps out to change the traditional lower bound selection procedure (i.e. zero). The first step is used to lower distance find the maxima i.e. LUB's (least upper bound of each row of the distance matrix). Second step provides minima i.e. GLB (greatest lower bound) it is used to determine the greatest lower bound of all LUB's. Quick sort technique is applied to sort the elements known as GLB's. The operational details of this procedure are illustrated as below:

Step 1: Finding maxima (or LUB's)

Suppose we have a distance matrix A of order $n \times n$. Also we have an array *max*[].

$$A = \begin{pmatrix} d_1^1 & d_2^1 & d_3^1 & d_4^1 \ldots\ldots\ldots\ldots & d_n^1 \\ d_1^2 & d_2^2 & d_3^2 & d_4^2 \ldots\ldots\ldots\ldots & d_n^2 \\ d_1^3 & d_2^3 & d_3^3 & d_4^3 \ldots\ldots\ldots\ldots & d_n^3 \\ . & . & . & . & . \\ . & . & . & . & . \\ d_1^n & d_2^n & d_3^n & d_4^n \ldots\ldots\ldots\ldots & d_n^n \end{pmatrix}$$

Let $\alpha_1, \alpha_2, \alpha_3, \ldots\ldots\alpha_n$ be the maximum elements of 1st, 2nd ,3rd …………n^{th} rows respectively. Therefore, we have a subset S_d which contains maximum distance value of each row of distance matrix A.

$$S_d = \{\alpha_1, \alpha_2, \alpha_3, \ldots\ldots\alpha_n\}$$

Since, this phase unearth the maximum distance value of each row of distance matrix, that's why it is termed as maxima.

Step 2: Finding Minima (or GLB)

Now, we have to find the minimum element out of these maxima's that's why this is termed as minima. To find the minima following steps are to be followed:

- $max[] \leftarrow S_d$ Remove all duplicate values *max*[] and maintain the unique distance values in ascending order. And we have
- $max[] = \{ \alpha_1, \alpha_2, \alpha_3, \ldots\ldots\alpha_m\}$,

  Where $m \leq n$

  Now Set L=$max[i]$ and U= $max[m]$ Where i = 0, 1, 2…………m

Thus, we can get the lower bound. The entire process exhibits that we have considered only those values, which are present in distance matrix to choose the lower bound as well as initial threshold distance, unlike the bisection method. A tight upper (U) and lower bound (L) is the major strength of proposed algorithm.

4.4.3 Description of Jump Based Scheme (JBS)

The jump based scheme is deployed to reduce the number of iterations needed to find the optimal coverage distance. Consequently, it speeds up the process. To substantiate the above mentioned statement, the functional detail and results produced by JBS are given below as evidence.

As stated above that the JBS is employed to reduce the number of iterations. The JBS implied in collaboration with ABSP. ABSP supplies *max*[] an array of elements (containing distinct maximum distance values of each row of distance matrix) as an input to the JBS. Initially, first element of *max*[] is passed to SCP function as initial threshold distance to find out the required result. When desired result not found then next distance value is selected by JBS from *max*[] and passed to the SCP function as distance parameter to check the feasibility as shown in Figure 4.4.

If solution is not feasible then it jumps to the second next value and repeat the process until feasible solution is reached. As soon as the feasible solution is attained then it checks the feasibility at preceding distance value. Due to jumping nature of method the preceding distance value remains unchecked; therefore it is required to check feasibility at preceding distance value too, it may provide a feasible solution. If feasible then this process is repeated until infeasible state is reached. As soon as it gets infeasible solution algorithm is stopped. A distance value due to which we attained the last feasible solution will be obtained as optimal coverage distance.

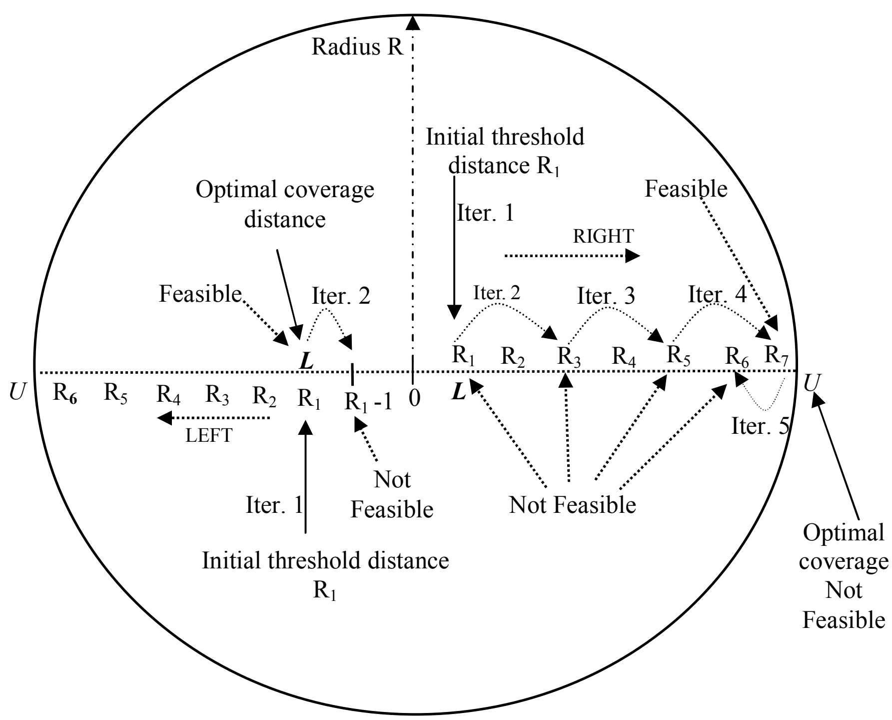


Figure 4.4: Illustrating Working of Jump Based Scheme

Figure 4.4 illustrates the working culture of JBS. It shows a big circle having LEFT half and RIGHT half. The LEFT half showing the situation where the initial threshold distance R_1 or *L* is passed to the SCP() for feasibility check. Since it is feasible, now it'll check for the preceding distance value i.e., R_1-1. Figure 4.4 shows that it is not feasible for R_1 -1, therefore the R_1 is the optimal coverage distance. Similarly, the RIGHT half is presenting another look of same scheme as shown in Figure 4.4, initially *L* is set to be the threshold distance and passed to the SCP() to check the feasibility of solution. If it is feasible then the former case will be repeated till it is not feasible. Otherwise, *D* is set to be to the second next distance value and procedure is repeated until it is feasible. As soon as it reaches to the feasible state then the preceding distance value is set to *D* and repeat the process until it is not feasible. Where it is infeasible; the former distance value is optimal coverage distance as shown in Figure 4.4.

It has been already proved by Al-Khedhairi [70, 71] through an analytical study of Jump Based method and generalized it as; if any algorithm needs five or more iterations to perform a particular task then Jump Based method takes lesser number of iterations to complete the assigned task. Consequently some improvement has shown by the Jump Base method [70]. At the same time Al-Khedhairi said that the jump based method does not shows any improvement in cases where less than five iterations are required. In general wherever large numbers of iterations are required this method has proven its worth to reduce the number of iterations.

4.5 Results Obtained

The major achievement of this algorithm is substantial reduction in number of iterations required to locate the requisite facilities. Presented algorithm is implemented on more than 30 problems having range of nodes 9 to 90 and facilities 3 to 58. Program executed on dual core 1.6 GHz Microprocessor with 1GB RAM and code is written in Turbo C++. Results obtained are described in Table 4.1.

Table 4.1 illustrates the results obtained through execution of proposed algorithm using ABSP and most famous Daskin's algorithm. Column 1 shows the number of nodes in network; Column 2 presents the number of centers to be located, column 3 and 5 showing the number of iterations taken by proposed algorithm and Daskin's algorithm to locate the required number of centers while the column 4 and 6 shows optimal distance value. Excluding a few cases, proposed algorithm has completed most of cases within fewer numbers of iterations than existing algorithm. Figure 4.5 exhibits the results of Table 4.1 in graphical form.

Table 4.1: Comparative Results Obtained by ABSP and DASKIN'S Algorithm

N	K	Proposed Algorithm		Daskin's Algorithms	
		#Iter.	Optimal value	#Iter.	Optimal value
9	3	3	6	6	6
9	4	2	4	5	4
9	6	3	3	5	3
13	3	4	13	6	20
13	4	2	10	6	10
13	7	4	8	6	8
13	9	7	5	9	5
13	10	8	4	6	4
20	6	12	12	6	12
20	7	3	7	6	12
20	10	4	6	6	6
20	12	2	5	5	5
20	17	3	4	6	4
25	7	16	28	7	28
25	8	23	21	8	21

25	9	5	18	6	18
25	10	6	17	7	17
25	12	7	16	7	16
25	15	2	15	6	15
25	18	4	13	6	13
25	22	7	10	6	10
50	22	4	22	6	22
50	23	2	20	7	20
50	28	3	19	6	19
50	29	4	18	7	18
50	32	5	17	7	17
50	34	7	15	7	15
50	42	8	14	7	14
50	45	10	12	6	12
75	45	11	12	13	12
75	48	11	13	13	13
90	50	13	15	15	15
90	55	12	16	15	16
90	58	13	16	14	16

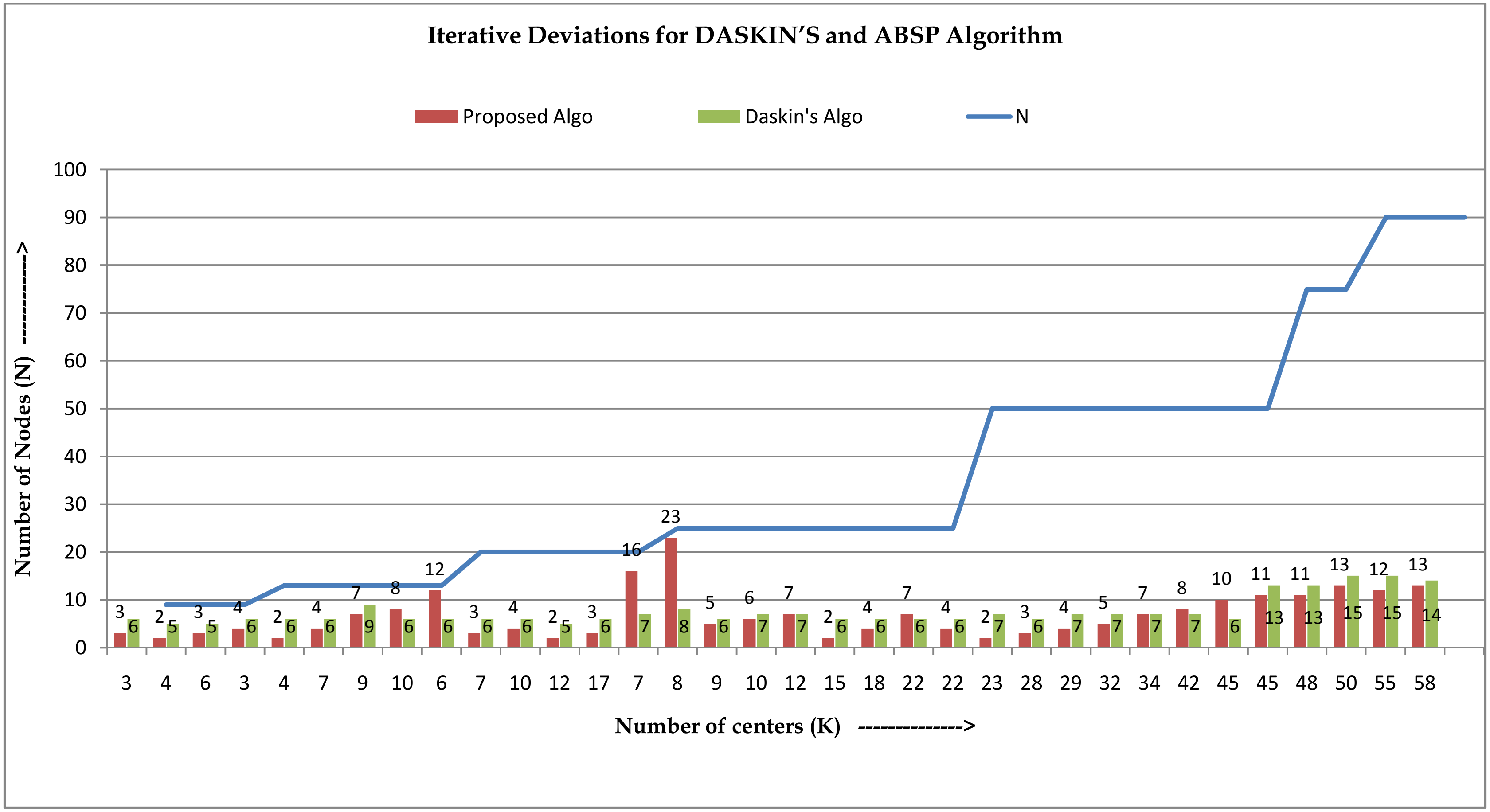


Figure 4.5: Illustrating Iterative Deviations for DASKIN'S and ABSP Algorithm

The overall impact of using Anticipatory Bound Selection Procedure in proposed algorithm makes it robust as well as efficient than the existing methods. Although the value of optimal coverage distance determined by both the algorithms in most of cases is equal but in a few cases, our approach showing better results. In other words the results of proposed algorithm substantiate the significance of improvements made in original algorithm for vertex *k-center* problem.

4.6 Results Analysis and Discussion

It is essential to perform some analytical technique to justify the results obtained and methodology employed. An iteration frequency analysis is performed on the results obtained from both the algorithms as shown Table 4.1. The analysis results are shown in Table 4.2 and Table 4.3.

This iteration frequency analysis reveals that the proposed algorithm takes at least two iterations to execute any assignment while the Daskin's algorithm takes at least five iterations to execute the same assignment. Moreover, most of cases are executed by proposed method taking 4 or 5 iterations except a few cases.

Table 4.2: Iteration Frequency for DASKIN'S Algorithm

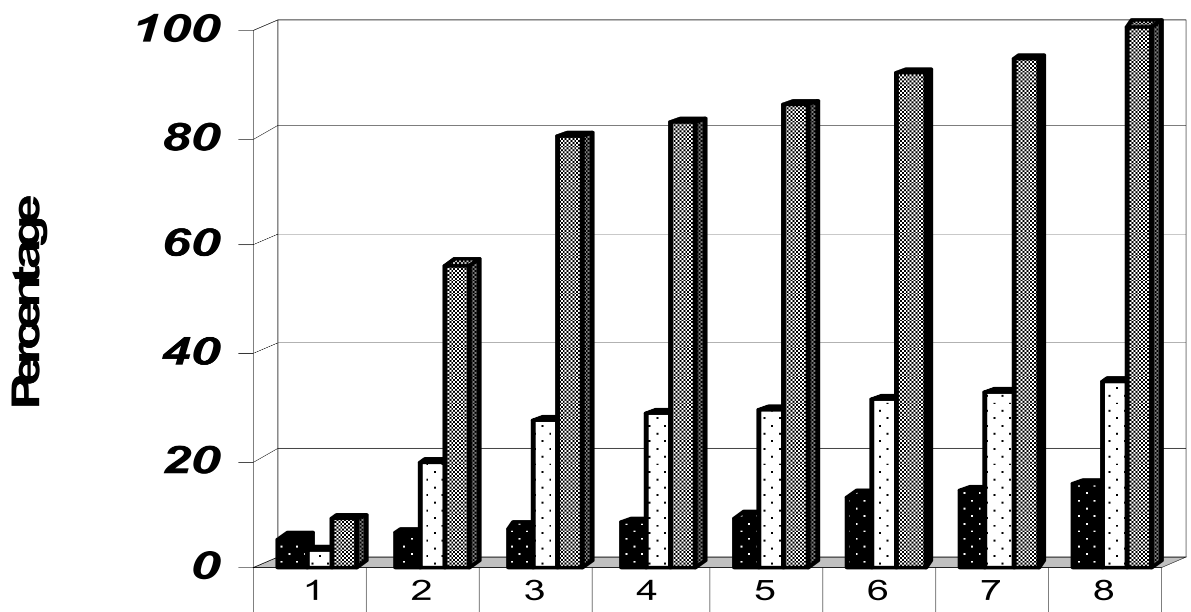
***DASKIN'S* Algorithm**				
Iterations	**Freq.**	**Freq. %age**	**Cumulative Frequency**	**Cumulative Frequency Percentage**
5	3	8.82	3	3
6	16	47.06	19	55.88
7	8	23.53	27	79.41
8	1	2.94	28	82.35
9	1	2.94	29	85.29
13	2	5.88	31	91.18

14	1	2.94	32	94.12
15	2	5.88	34	100.00

Table 4.3: Iteration Frequency for PROPOSED (ABSP) Algorithm

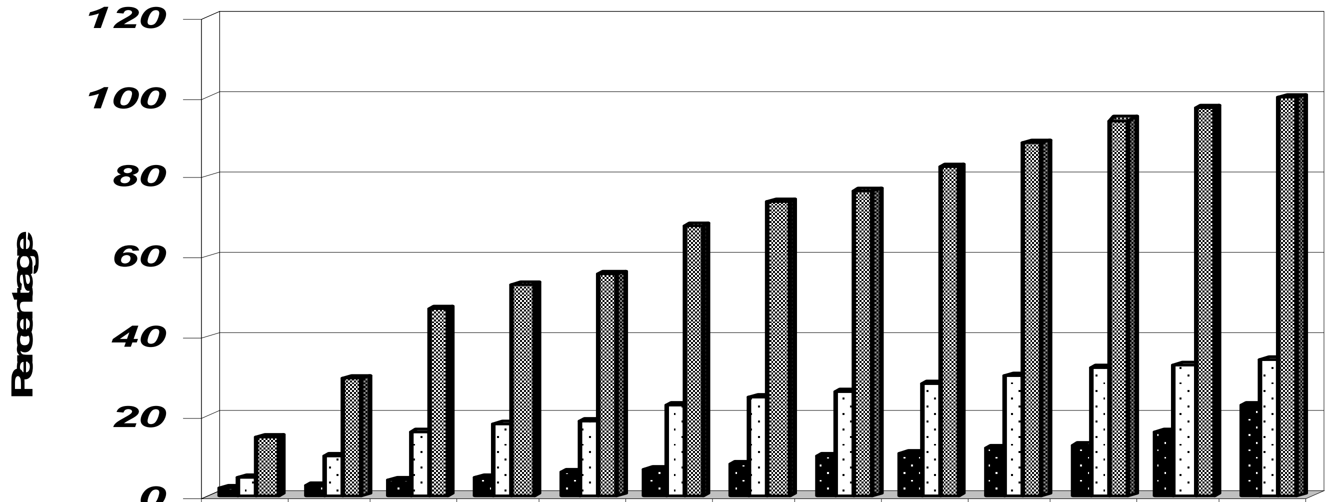
***PROPOSED* Algorithm**				
Iterations	**Freq.**	**Freq. %age**	**Cumulative Frequency**	**Cumulative Frequency Percentage**
2	5	14.71	5	14.71
3	5	14.71	10	29.42
4	6	17.65	16	47.06
5	2	5.88	18	52.95
6	1	2.94	19	55.89
7	4	11.76	23	67.65
8	2	5.88	25	73.53
10	1	2.94	26	76.47
11	2	5.88	28	82.36
12	2	5.88	30	88.24
13	2	5.88	32	94.12
16	1	2.94	33	97.06
23	1	2.94	34	100.00

It is clearly shown in column cumulative frequency percentage of Table 4.2 and Table 4.3 that the Daskin's algorithm completes only 8.82 percent of jobs within the range of 5 iterations while the proposed algorithm ***ABSP*** completes 52.95 percent. It is noticeable in both the tables that there is no iteration less than 5 in the first column of Table 4.2 while in Table 4.3 there is the minimum iteration number that is 2 with frequency 5. Moreover the cumulative frequency of iteration 4 is 16 in Table 4.3 it's an evidence that more than 47 percent of work is done by the proposed algorithm by exhausting at most of 4 iterations.



	1	2	3	4	5	6	7	8
No of Iterations	5	6	7	8	9	13	14	15
Commulative Frequency	3	19	27	28	29	31	32	34
Commulative %age	8.82	55.88	79.41	82.35	85.29	91.18	94.12	100.0

Figure 4.6: Cumulative Frequency Analysis of DASKIN'S Algorithm



	1	2	3	4	5	6	7	8	9	10	11	12	13
No of Iterations	2	3	4	5	6	7	8	10	11	12	13	16	23
Commulative Freq.	5	10	16	18	19	23	25	26	28	30	32	33	34
Commulative %age	14.71	29.4	47.0	52.9	55.8	67.6	73.5	76.4	82.3	88.2	94.1	97.0	100.

Figure 4.7: Cumulative Frequency Analysis of PROPOSED ABSP Algorithm

The difference in performance of both the algorithms is visible. Approximately 50% of work has been completed by the proposed algorithm taking four iterations at most for each case; on the other hand the Daskin's algorithm could not execute even a single case taking four iterations. Daskin's algorithm takes at least five iterations to execute any of given problem. A graphical representation of this analysis is also presented through the Figure 4.6 and 4.7.

As already mentioned, that the success of presented algorithm is in the reduced number of iterations. It is just because of methodology opted to select the initial threshold distance i.e. **ABSP** & **JBS** as described earlier. The overall impact of presented methodology is that many cases are able to complete in smaller number of iterations. Since the bisection method iteratively half the distance values to reach towards optimal distance while the presented methodology selects the maximum distance values and checks the possible feasible solution in just selected values, additionally the Jump Based Scheme speed up the searching process. Consequently, ABSP and JBS collectively provide results in lesser number of iterations.

The proposed algorithm eliminates the swings in distance values (*D*) with the help of Anticipatory Bound Selection Procedure and provides a simple approach to move towards the optimal coverage distance. Jump Based Scheme played a remarkable role in reducing iterations. Eventually, the results produced by algorithm have shown some improvement. Even though, all the results produced by the proposed algorithm are well simulated and reliable.

4.7 Algorithm for Obnoxious Facility Location Problem

This study presents an endeavor to determine an optimal solution for the obnoxious facility location problem. Obnoxious materials are those which may cause harm to the health of human beings and pollute the environment. The obnoxious material management is a significant environmental issue. Since the social economical growth encourages to the establishment of industrialization. While the waste material of industry or such plants necessitate a careful placement to avoid any kind of damage to the inhabitant of concerned area. During the last few decades, a significant work has been done in operation research, graph theory and computation complexity for the placement of desirable facilities due to increasing demand and production of goods. But the mentioned problem could not get a considerable attention of researchers. Hence this study provides a simplified approach to solve the obnoxious facility location problem on network.

4.7.1 Problem Model

Facility location problem is a most significant and sub problem of commerce optimization problem. Facility location problems emerged as a challenge for both public and private sectors. Since it is the moral responsibility of any state government to provide all basic facilities to their citizens like hospitals, schools, colleges, ration depots and fire stations etc. Similarly, it is essential for all big and small production houses to reach up to last user of their goods.

Location of such facilities is very significant issue and needs to consider impact of various relevant parameters (such as distance, population and access time) on a location of facility. Broadly we can categories these facilities in two categories like desired facilities and undesired facilities, since all kind of facilities have some impact on quality life. Desired facilities are those which are desired by inhabitants to be placed in closer areas, such as schools, hospitals, company outlets. On the contrary, obnoxious facilities are those, which are never desired to be placed nearby areas by the inhabitants such as garbage dumps and chemical plants etc. due to their adverse effects.

Obnoxious facility location problem deals with the proper placement of such materials, which are preferred to be placed far from the populated area to prevent the inhabitants from health related issues as caused by such materials. There is a wide list of obnoxious materials such as waste dumps, nuclear power plants, chemical plants, electricity power plants, waste released by industry, airports, corrosive substances, gas plants, flammable liquids and solids, oxidizing substance, radioactive material, poisonous and infectious substances. If such materials are located closer to any populated area it may be dangerous for the life of mankind.

Keeping in view all adverse effects of the obnoxious facilities over the environment and population, it is crucial to locate these facilities away from the populated area. This study focuses on to develop an algorithm that can provide an optimal solution for obnoxious facility location problem. The subsequent section provides a simplified procedure to allocate obnoxious facility.

4.7.2 A Simplified Procedure to Allocate Obnoxious Facility Location

This section endows with the comprehensive methodological details of the presented approach as well as formulation of the problem. It is assumed that the facility to be located on network.

The primary objective of the proposed approach is to maximize the minimum distance between a demand node and its nearest obnoxious facility. Let us suppose that we have a network of nodes that can be represented as a graph $G = (V, E)$, V is a set of vertices (nodes) and E is as set of edges (path) that connects different vertices. Every edge is having a weighted distance from one vertex to another.

P and Q are the sets of demand nodes and candidate sites respectively

$$P = \{p_1, p_2, \ldots\ldots, p_n\};$$

$$Q = \{q_1, q_2, \ldots\ldots, q_m\};$$

d_{pq} = is the distance between demand node and candidate site where $p \in P$, $q \in Q$

G = the minimum distance between a demand node and it's nearest obnoxious facility in a given network.

Therefore the objective function can be presented as given below:

$$\textbf{Maximize } G \qquad (1)$$

Subject to:

$$Min_q \ d_{pq} \geq G \qquad \forall p \in P \qquad (2)$$

Equation (1) states that we have to maximize the minimum distance between a demand node and its closest facility. Constraint as shown in equation (2) states that after allocation any of the node connected to facility should have distance less than G.

Assuming that, we have to allocate single obnoxious facility location on any given network. Distance is a crucial parameter that is highly significant for deciding the location of facility. Therefore, first of all it is indispensable to find out such nodes those are located at highest distance to all other present nodes in network. This can be attained by using the distance matrix. This distance matrix helps us to find out the maximum and minimum distance of each row. Suppose we have D as distance matrix of $[n \times n]$ order. We are also having two arrays *Max*[] and *Min*[] that represents maximum and minimum distance values respectively.

$$D=\begin{pmatrix} d_1^1 & d_2^1 & d_3^1 & d_4^1 \dots\dots\dots\dots & d_n^1 \\ d_1^2 & d_2^2 & d_3^2 & d_4^2 \dots\dots\dots\dots & d_n^2 \\ . & . & . & . & . \\ . & . & . & . & . \\ d_1^n & d_2^n & d_3^n & d_4^n \dots\dots\dots\dots & d_n^n \end{pmatrix}$$

Let α_1, α_2, α_3, …….α_n be the maximum distance value elements and β_1, β_2, β_3, …….β_n be the minimum distance value (but not zero) elements of 1st, 2nd ,3rd …………n^{th} rows respectively. Therefore, we have two sub sets Max_d and Min_d having all the elements with highest distance value and lowest distance values respectively.

$Max_d = \{\alpha_1, \alpha_2, \alpha_3, \dots\dots\alpha_n\}$

$Min_d = \{\beta_1, \beta_2, \beta_3, \dots\dots\beta_n\}$

Initialized both the arrays as shown below

$Max[] = Max_d$ $0>\alpha<\infty;$

$Min[] = Min_d$ $0>\alpha<\infty;$

Thereafter we have to find a pair $\alpha i\ \beta i$, where αi is having the maximum value in *Max*[] and βi is maximum distance value in *Min*[].

Now let us have an example to understand the procedure opted to implement presented approach. Suppose we have a network of nodes as shown in Figure 4.8. A, B, C, D, and E are the connected nodes and the weighted distance between nodes is mentioned above each edge.

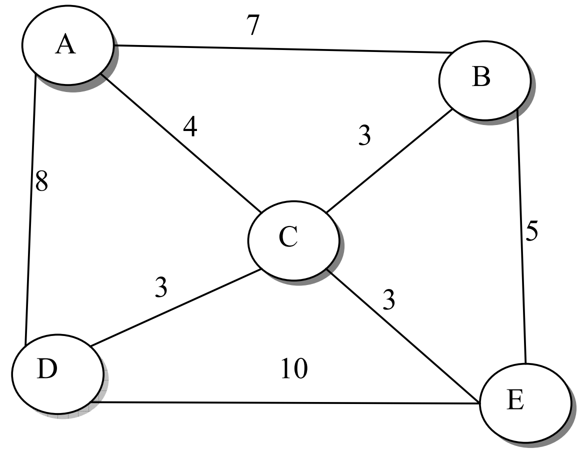


Figure 4.8: Network of Nodes

Table 4.4 presents the distance matrix for the network shown in Figure 4.8

Table 4.4: Distance Matrix

Nodes	**A**	**B**	**C**	**D**	**E**
A	0	7	4	8	∞
B	7	0	3	∞	5
C	4	3	0	3	3
D	8	∞	3	0	10
E	∞	5	3	10	0

Now we have to proceed to find out the maximum (Neither ∞ nor 0) distance value from each row of distance matrix to initialize the *Max*[]. Similarly minimum distance values (neither 0 nor ∞) from each row of distance matrix to *Min*[].

Max[] = {8, 7, 4, 10, 10};

Similarly *Min*[] = {4, 3, 3, 3, 3};

Then we have applied Sorting technique on *Min*[] to identify the maximum distance value. A particular index "*i*", which holds the maximum distance value in *Min*[], will be considered to select a distance value from *Max*[].

As we have shown in our example that 4 (i.e. at index '0', as encircled in Table 4.4) is the maximum distance value of *Min*[]. Similarly the *Max*[0] will be selected as maximum distance value of *Max*[]. Hence we got a distance pair (maximum-8, minimum-4). It reveals that the candidate site for the obnoxious facility is node A (since 8, 4 both distances belongs to node A).

The main objective is to provide priority to the maximum distance in *Min*[] array is only to increase the distance between connected nodes and obnoxious facility. In view of the fact, that everybody wants, that the obnoxious facility should be located as far as possible. Therefore we have tried to maximize the distance between nodes and obnoxious facility. Then the selected maximum distance from *Min*[] array is passed to FLoc() procedure to identify the appropriate node for obnoxious facility.

Algorithm to Allocate Obnoxious Facility

Input: A[][]; //A is distance matrix

Step 1: **Initialize** *Max*[] = { $\alpha_1, \alpha_2, \alpha_3, \ldots\ldots\alpha_n$; $\forall\alpha\in A$ };

Min[] = { $\beta_1, \beta_2, \beta_3, \ldots\ldots\beta_n$; $\forall\beta\in A$ };

//Every α is the maximum distance value (but not infinity) of each row of A,

//Similarly every β is the minimum distance value (but not zero) of each row of A.

Step 2: **Search** Minimum=Maximum (*Min*[]); //*i*=0, 1, 2……….N

//searching maximum element of *Min*[]

k= *i*; *// i is the index at which we got maximum of Min[]*

Maximum= Max[*k*]

Step 3: **Run** FLoc(Maximum, Minimum);

//Floc() is process to find out resultant node

4.7.3 Results and Discussion

As the presented approach has provided results to It is quite important to remind that the presented approach has considered a case where we have to allocate only one obnoxious facility on given network. Since such facilities are never desired to have in a huge number. Additionally, such facilities are required to be allocated as far as possible from the populated areas. As per the objective function or our problem, it is required to increase the minimum distance from a node to its nearest obnoxious facility.

More than 50 cases with different numbers of nodes has been undertaken to test the presented approach. The number of nodes covered in a set from 8 to 98. During the computation a few percent relative errors are reported as usual. But the major achievement of algorithm is that it has speed up the overall processing, consequently the time taken by the algorithm is comparatively low, as shown in Table 4.5. Computation times are expressed in CPU seconds Program executed on dual core 1.6 GHz Microprocessor with 1GB RAM and code is written in Turbo C++.

Table 4.5: Results Obtained from Presented Approach (OFLP)

Total Nodes	**No of Iterations**	**Time Taken by Algorithm (in CPU seconds)**	**Average Time (In CPU seconds)**
15	1	2.45	2.45
	2	2.46	
	3	2.45	
20	1	3.30	3.30
	2	3.30	

	3	3.30	
25	1	4.50	4.52
	2	4.55	
	3	4.52	
35	1	5.93	5.93
	2	5.93	
	3	5.92	
42	1	6.80	6.80
	2	6.80	
	3	6.80	
57	1	7.99	7.99
	2	7.98	
	3	7.99	
66	1	8.70	8.70
	2	8.71	
	3	8.70	
83	1	10.35	10.36
	2	10.36	
	3	10.36	
98	1	11.26	11.25
	2	11.25	
	3	11.25	

Table 6.2 presents the execution time variation of a few cases out of total 50 tested cases. It reveals that the execution time slightly increases with the increased number of nodes. But the increase in execution time is very marginal. During the testing phase we have observed a significant increase and decrease in execution time in a few cases which usually occurs.

In order to verify the execution time for such cases, it has been performed repeatedly. When the consistent time is reported by the algorithm in consecutive execution then it has been noted as final execution time for a particular case (such exceptions are not shown in Table 4.5). But for majority of cases as the presented algorithm has shown very consistent behavior as described in Figure 4.9.

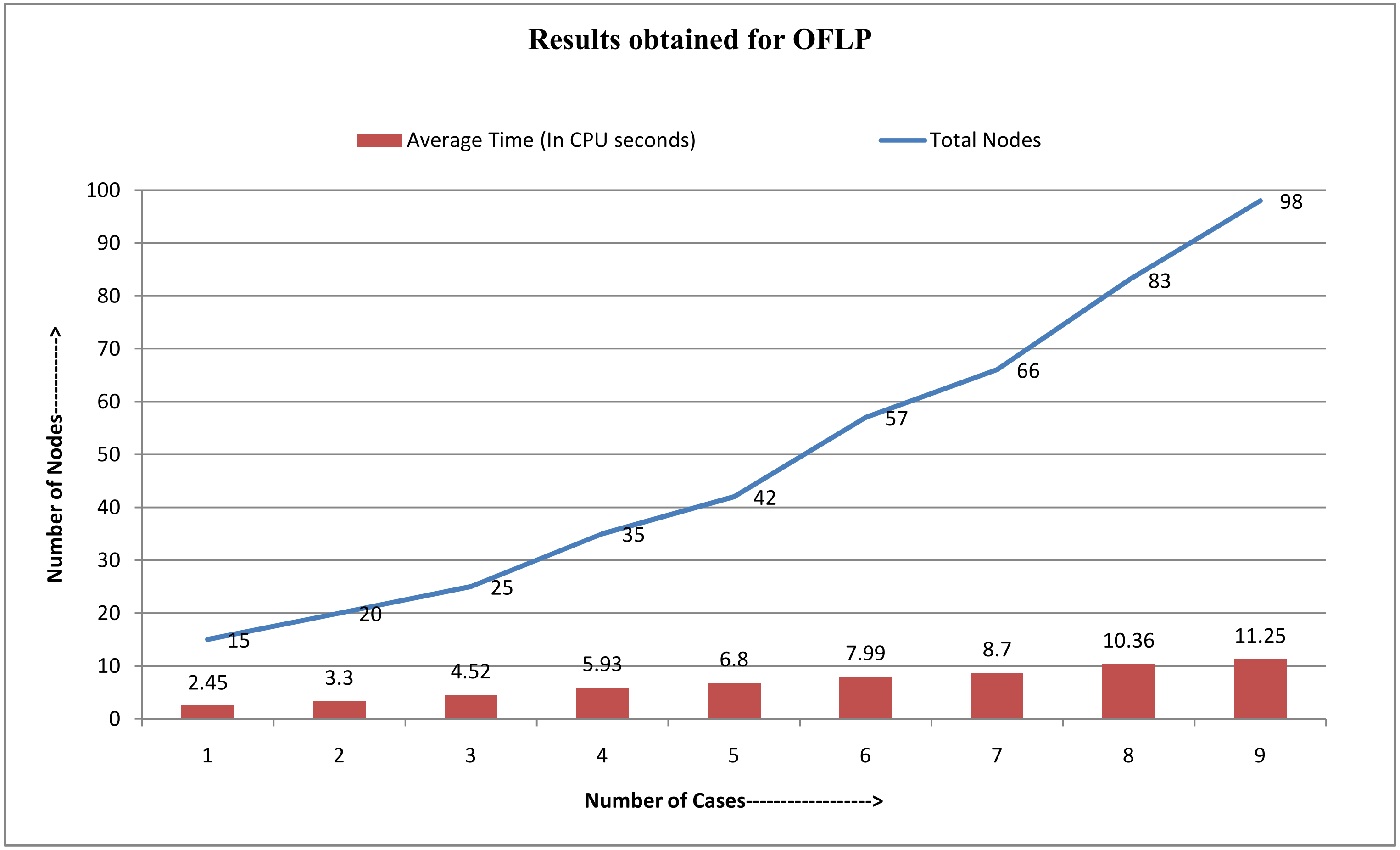


Figure 4.9: Graphical Presentation of Results Obtained for OFLP Algorithm

The problem undertaken has a great significance and impact on the structure of modern society as well as environment. Keeping in view all the facts of matter, it is essential to locate the obnoxious facility away from the residential areas for the sake of healthy environment and health of inhabitants. The presented approach provides an effective and simplified method to allocate obnoxious facility locations (for 1-center only). The major achievement of presented approach is that it increases the minimum distance between obnoxious facility and connected nodes. A few related issues such as transportation of obnoxious material and their routing is still an open problem for research.

Chapter 5

An Improved Approach for Incremental Variant of Mobile Facility Location Problem

Facility location decisions are utterly dependent on the emerging market trends and other factors such as population and demand. The aim of decision makers is to find a robust location with variable demand on changing time horizon. This chapter introduces a new approach to deal with mobile facility location problem in a dynamic environment. This algorithm endows with a solution of *k*-center as well as *k*-median problem simultaneously in perspective of mobile facility location allocation. The significant aspect of proposed algorithm is that it undertakes both the possibilities of facility location, i.e. immediate assignment of facility and after n^{th} period allocation of facilities to inward demand.

5.1 Problem Model

Facility location problem deals with the location allocation issues. Since last three decades intensive efforts have been made by several researchers in the area of operation research and computer science to reach optimal solution of this problem. There are various significant aspects of facility location problem pertaining to the objective functions such as minimization of facility establishment cost, distance between clients and facility site and capacity of facility center, etc. Uncertainty of parameters is one of the major issues in facility location problem. However, a substantial research work has been done in this direction to find an optimal solution for facility locations but a major part of the work represents static natured facilities such as distribution centers, hospitals, schools, fire stations, etc. Facility location problem deals not only with the static facilities rather it also covers mobile facilities. In case of mobile facility various parameters are there which are used to change very frequently (such as population of region, increased or decreased demand at any demand node) and play a very vital role in deciding location of any facility.

For example, the placement of Ambulance vans, Fire Brigade stations, which directly refer to the dynamic facility location problem. The continuous changes in market trends suggest that it is essential to satisfy the consumer for both private as well as public sector. A few common parameters such as increased population, demand and timeliness directly or indirectly force the service providers to improve their services.

The impact of these parameters plays a very vital role for strategy makers in deciding the right locations for facility centers. For instance public health department wants to provide several ambulance vans (Mobile Facility) for emergency services to the needy inhabitants of a given region. It is intended that each ambulance van provides services within a specified minimum distance $\boldsymbol{r}$. The aim is to provide the fewest number of ambulance vans so that all expected customers (beneficiaries) within radius r can avail the facility within minimum traveling distance and time. As with many conventional problems, the aim of facility location problem is to provide a static solution to a static set of customers. While in reality it is expected that the customer-base will grows vigorously over time. Therefore, we need a solution that is capable to accommodate a newly arrived customer. In order to satisfy the requirement of ever increasing customers it is necessary to locate more and still more facility centers.

First of all, it is required to know that how much demand is increasing periodically, and then the important aspect is that whether the increased demand can be accommodated through existing facilities. If it can be accommodated then a proper assignment of demand nodes to the existing facilities should be completed immediately. As far as the incremental version of facility location problem is concerned, the main objective is to allocate new facilities and assign demand upon arrival. This study proposes an improved approach for incremental variant of mobile facility location problem for planning facility locations as desired by both private as well as public sectors.

5.2 Incremental Model for Mobile Facility Location Problem

The incremental model of mobile facility location problem deals with some specific issues related to facility location problem in dynamic environment. It is also important for decision makers to analyze all such issues before making a strategy. The presented model provides an efficient solution for vertex *k*-center problem, and its incremental variant for dynamic facility location problem. Since distance is one of the major and sagacious parameter which makes the allocation of any facility more significant. The objective of our algorithm is to minimize the distance between facility points and all associated demand nodes. There may be two possibilities of entertaining the inward demand, either by allocating the facility to each new demand node immediately as soon as it arrives or by allocating facilities to all new demand nodes after a specific period. The proposed model covers both the possibilities, and provides an effective method to deal them as per the situation.

The primary objective of presented algorithm is to provide immediate allocation of facility to inward demand. It's presumed that there exist some facility points in nearby areas. Our algorithm provides graceful solution to this problem. It checks the distance of these demand nodes to all existing facility points and compare these distance values with previously settled optimal distance value ***d***. The facility point having distance less than or equal to ***d*** will be the best possible nearest facility point for inward demand. If no match is found, then the relocation of facilities will take place.

If relocation of facilities is required then, a few additional parameters are to be analyzed. Such as the time period after which this relocation is taking place, increased demand/ population, requirement of extra facilities and then find out the best locations for facility points in present state.

5.3 Methodology Implemented

Since, we are discussing the incremental model for mobile facility location problem, where the demand arrives in incremental order, it is necessary to analyze the impact of increase in population on demand. Obviously the increase in population will result in increase in demand. Consequently it necessitates finding out the number of facilities required to fulfill the increased demand at any point of time. Now the most important question is how to find the total increase in population/demand as well as number of facilities required? Let us find the answer to this question as under: Suppose

n_i –be the number of demand nodes, where $1 \leq i \leq k$

p_{n_i} –be the present population of each node, where $1 \leq i \leq k$

u – be the number of facilities already allocated

Then a facility can accommodate population at a time $= \dfrac{\sum p_{ni}}{u}$

Let r be the rate of annual increase of population, then total population after

n^{th} year $$\sum P_{ni} = \sum p_{ni} \times (1 + \frac{r}{100})^n$$

At the end of n^{th} year, we have extra population as under

$$= \sum P_{ni} - \sum p_{ni}$$

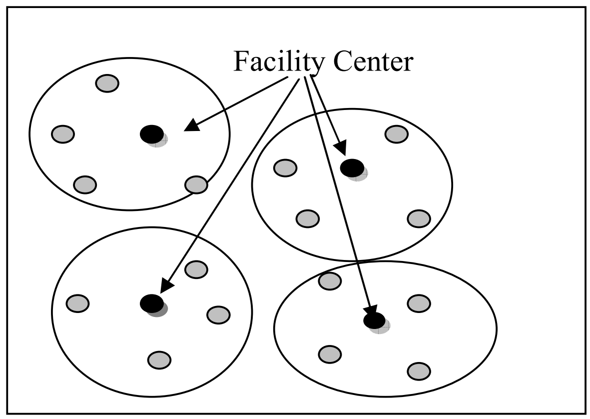


Figure 5.1: Illustrating Current Population and Facility Center Allocated

Figure 5.1 presenting the current scenario where facilities are allocated as per the demand as well as the facilities allowed by the authorized body. Figure 5.2 demonstrates the scenario of incremental demand according to increase in population over a time.

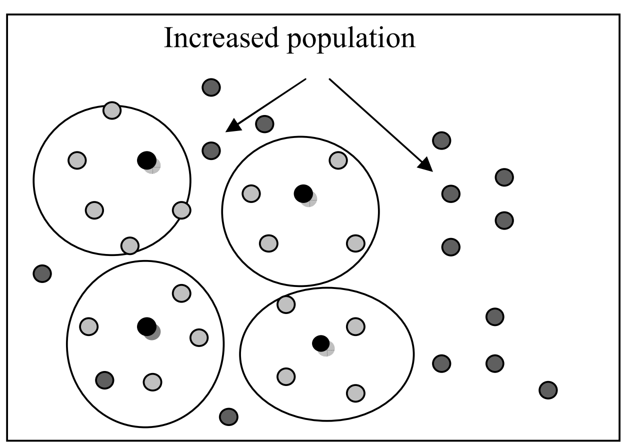


Figure 5.2: Illustrating Increased Population over n^{th} year

Consequently, it is required to find out the number of extra facilities needed to accommodate the increased demand after n^{th} year as given under:

$$= (\sum P_{n_i} - \sum p_{n_i}) \times (\frac{u}{\sum p_{n_i}})$$

i.e. $$= u(\frac{\sum P_{n_i}}{\sum p_{n_i}} - 1)$$

At the end of n^{th} year, as shown below K (including existing facilities) is the total number of facilities required to accommodate the current population.

$$K = u + u(\frac{\sum P_{n_i}}{\sum p_{n_i}} - 1)$$

$$= u(1 + \frac{\sum P_{n_i}}{\sum p_{n_i}} - 1)$$

$$K = u(\frac{\sum P_{n_i}}{\sum p_{n_i}})$$

Thus we attained the required number of facilities to accommodate the augmented demand over a time. Further it is required to identify such locations in a given area, which are capable of serving all connected demand nodes with some intrinsic characteristics such as less traveling distance, approximately equal accessibility from all adjacent sites and takes minimum time to approach. Primarily, it is essential to create such clusters of given demand nodes that are connected to each other and have marginal distance among them. So that the facility access time and distance between demand node to its nearest facility can be minimized.

In order to attain a feasible and optimal solution we have used Anticipatory Bound Selection Procedure (**ABSP**) and Jump Based Scheme (**JBS**) as described in Section 4.4 of chapter 4. To understand the proper working of algorithm let us have an example as given under:

To show the performance of proposed algorithms let us have an example of placement of police vans (known as PCR). Suppose the state administration wants to place police vans (PCR) in a city to maintain the law and order. Let us suppose, initially city is divided into six divisions named as **A**, **B**, **C**, **D**, **E**, **F** and we have only three vans to place in these six divisions as shown in Figure 5.3. Now the problem is to identify such three locations in a given network of nodes where these PCR vans can be placed, so that the purpose can be served effectively.

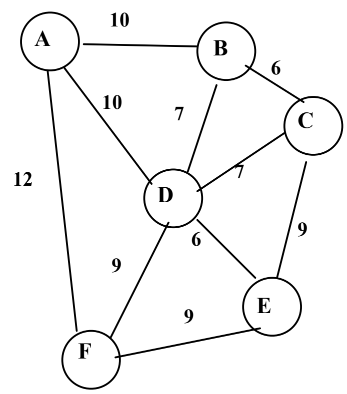


Figure 5.3: Showing Location of Nodes in network

The network shown below provides a site map different locations in the city and their respective distances. The objective is to find the desired number of locations to justify the objective within lesser traveling distance and minimum access time. For an instance after n^{th} period the population of city increases and more nodes like **G**, **H**, **I** are also added to the network as illustrated in Figure 5.4.

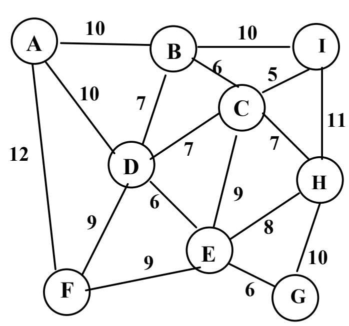


Figure 5.4: Showing increased population in network after n^{th} year

Assumptions

Let N =Number of nodes in given network, K=Number of facilities to be located, P=Number of sets found using any distance value, LB= Initial threshold distance (lower bound), UB= maximum distance value (upper bound)

Solved the problem using proposed algorithm as shown below:

Create distance matrix to initialize the *max*[], *max*[] is an array containing the maximum distance value of each row of distance matrix shown in Table 5.1.

Table 5.1: Distance matrix based on Figure 5.3

Nodes	**A**	**B**	**C**	**D**	**E**	**F**
A	0	10	0	10	12	0
B	10	0	6	7	0	0
C	0	6	0	7	0	9
D	10	7	7	0	9	6
E	12	0	0	9	0	9
F	0	0	9	6	9	0

Table 5.2: Distance matrix based on Figure 5.4

Nodes	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
A	0	10	0	10	0	12	0	0	0
B	10	0	6	7	0	0	0	0	10
C	0	6	0	7	9	0	0	7	5
D	10	7	7	0	6	9	0	0	0
E	0	0	9	6	0	9	6	8	0
F	12	0	0	9	9	0	0	0	0
G	0	0	0	0	6	0	0	10	0
H	0	0	7	0	8	0	10	0	11
I	0	10	5	0	0	0	0	11	0

Facility allocation using proposed algorithm for Figure 5.3 is given as under:-

Initialize the *max*[ ] ={12, 10, 9, 10, 12, 9}, Now perform sorting on *max*[ ] and keep only the unique values: *max*[ ]= {9, 10, 12}. For the given set of data algorithm is executed and following results are provided in Table 5.3. The highlighted row of Table 5.3 is showing final set of nodes.

Table 5.3: Results of Proposed Algorithm for Figure 5.3

Iter. No	LB	UB	D	K	Sets Found
1	9	12	9	3	A, BCD, EF
2	8	9	8	4	A, BCD, E, F

Figure 5.4 illustrating that after n^{th} period of time population of given region has been increased and G,H, I are the new nodes added to the given network. Now the problem is to accommodate the newly arrived demand. Let us see how the proposed study undertakes the problem.

Initialize the *max*[] ={12, 10, 9, 10, 9, 12, 10, 11, 11}, Now perform sorting on *max*[ ] and keep only the unique values: *max*[ ]= {9, 10, 11, 12}.Consequently, following results are provided by the execution of algorithm. The results are shown in Table 5.4 as given below.

Table 5.4: Results of Proposed Algorithm for Figure 5.4

Iter. No	LB	UB	D	K	Sets Found
1	9	12	9	4	A, BCD, EFGH, I
2	11	12	11	4	ABD, CEHI, F, G
3	12	12	12	3	ABDF, CEHI, G
4	11	12	11	4	ABD, CEHI, F, G

Results provided by the algorithm reveals that the three vans are not sufficient to accommodate the increased demand after n^{th} years, since allocation is done with maximum distance value in the distance matrix. Therefore, it concludes that more vans are required to provide efficient services. As much as the traveling distance will be lesser the service time will be better. Hence to reduce the access time, it is essential to decrease the traveling distance.

5.4 Proposed Approach to Solve the Incremental Variant of Mobile Facility Location Problem

The proposed algorithm is efficient to accommodate the increased demand as well as to reallocate the existing facilities in order to solve the incremental variants of mobile facility location problem. The major components in facility location problem are area of given region, inhabitants of that region (population), distance, response time and desired numbers of facilities. These are the major parameters considered during the design of algorithm. An effective implementation is promised by the algorithm. This algorithm deals with dynamic nature of demand. A brief introduction to all steps of proposed algorithm is given as under.

Algorithm 5.1 *Algorithm for Incremental Variant of Mobile Facility Location Problem*

Step1: Input $D[][]$, dx_iy, d, n_i, p_{ni}, u, r, n;

//Input the adjacent distance values of inward demand node in distance matrix D

Step2: Compare

If ($dx_iy \leq d$) then //{Where $x \in X$, $1 \leq i \leq k$, and $y \in Y$}

{$y = x_i$ and Stop }

Step 3: Compute P_{ni} , K; // {Where P_{ni} is total population after n^{th}

year}

Step 4: Initialize *max*[]; // {Where *n* is the number of years}

Step 5: Sort *max*[]; //{Keep unique values only}

Step 6: Initialize Set ***L***= *max*[0];

Set ***U***= *max*[*m*]**;** **// {**Where *i*=0,1,2……*m*}

Step 7: Set *d*= *L*, *i*=0;

Step8: Run SCP();

If (*p*>*K*) then //{ *p* represents the number of facilities found}

{ **Set** ***d*** =(*max*[*i*+2])

Go to Step 8;

}

Else If (*p*<*K*) then

{ **Set** ***U*= *d*, *L*=** (***d***-1), ***d*= *L***

Go to Step 8;

}

Step 9: Set *temp* **=d, *L*=*d*-**1**, *U*=*d*** and ***d*=*L*.** **//**{means its feasible solution}

Run SCP();

If (*p*>*K*) then //{*temp* is the optimal covering distance.}

{ **Stop }**

Else If (*p*=*K*) then // {it's feasible solution}

{ Set ***temp*** = ***d*;**

Go to Step 9

}

Output: *d*, *K*,

5.4.1 Algorithm Description

First step of algorithm represents input section where all the required parameters are initialized such as distance matrix *D*[][], distance of inward demand node to nearest facility (***dx_iy***), previous optimal distance value (***d***), number of demand nodes previously connected (***n_i***), population at each node (***p_{ni}***), rate of increase in population/demand (***r***). time period (***n***) and number of previously allocated facility centers (***u***).

Second step deals with immediate allocation of facility to the inward demand, if it satisfies the condition that ***$dx_iy \leq d$***. In other words we can say that if there exist any road linkage with any of the existing facility center then their distance will be checked from distance matrix. If, its distance is less than ***d*** (the previously obtained optimal distance) then it will be assigned to the nearest facility without making any reallocation of facilities. Otherwise it will go to step 3.

Third step of algorithm deals with all related issues such as increased demand and number of facilities required to accommodate current total population/demand. Therefore it computes P_{ni} and K. and reallocation of facilities on the basis of parameter initialized in first step.

The fourth step is used to initialize the *max*[]. The array *max*[] contains a set of maximum distance value of each row in distance matrix. The ***d*** is the coverage distance. The variable ***temp*** keeps the value of *d* temporarily. Fifth step of algorithm performs sorting of *max*[] in ascending order and remove the duplicate values (if any).

Sixth step is used to initialize lower bound (L) and upper bounds (U). Lower bound is initialized with the minimum value in *max*[*i*] where *i*=0, and upper bound is initialized with maximum value in *max*[*i*] where *i*=*m*. Thus we got the initial threshold distance to find out the required number of location to allocate facilities.The seventh step of algorithm is used to initialize the initial threshold distance ***d*** with lower bound (L) i.e. ***d***=*L*.

The eighth step is used to solve set covering problem with the help of a subroutine SCP() which returns ***p*** as number of facilities found in network. If $p > K$ then it is not a feasible solution. So we need to select next distance value to check the feasibility of solution. As per the JBS (Jump Based Scheme) the second next distance value to be assigned to *d* e.g. set *d*=max[*i*+2] and repeat step 8 until it becomes feasible.

There may be a second condition where $p < K$ then it shows that now we need to reduce the distance value and check the feasibility, as well as it an indication that the optimal distance value will be less than the present value of ***d***. Therefore the set U=***d***, L=***d***-1, ***d***=L and repeat step 8 until a feasible solution found. and If it provides a feasible solution for any value of *d* then got next step.

Step nine set *temp* = d, U=***d***, L=***d***-1, ***d***=L and then run SCP() function to check the a feasible solution. If it provides $p>K$ then not feasible and Stop it. Now *temp* is the optimal coverage distance value, otherwise repeat step nine.

5.5 Analysis and Discussion

The proposed algorithm is one of the simplest and multipurpose algorithms. It works on special property distance. This directly influences associated factors like access time, etc. which are most important in case of mobile facility. In fact it is the extended or special version of facility location problem. It supplies an efficient way of unraveling several issues related to mobile facility location problem (such as identification of facility locations, minimizing the traveling distance, reducing access time) as well as *k*-center problem in vibrant (dynamic) environment. A special consideration has been given to the increased demand (population) in given region and generated demand. Similarly E. Demaine et al. [124 and 123] worked on specific movement problems and provided good results.

The main objective of our algorithms is to reduce the distance among demand nodes and facilities by identifying such optimal locations of mobile facilities, which can accommodate all existing as well as incremented demand in a dynamic environment through a simple method. Finally the distance is one of the major parameter which directly affects all relevant issues such as covering distance, access time, and response time. The present approach sets aside the given region in approximately equal size partitions (clusters) and then allocates the facility in different clusters in such a way that all connected nodes can access the facility with in equal distance i.e. optimal distance d. We may now dwell upon the effect of distance parameter on all other issues as given under:-

Statement 1: The d is said to be an optimal distance value if $d_{xy} \leq d$ and satisfies the condition $\| x_i - y_j \| \leq d$. where i is fixed and j varies in case of each cluster.

Let us suppose A is the area of given region and K are the facilities to be allocated. Therefore K facilities are to be allocated in given region (area A) with assumption that the access distance of each demand node to the facility should be as minimum as possible. Now the region is divided in K equal parts as shown in Figure 5.5. Each partition contains some demand nodes and we have to identify a location among them which is an appropriate candidate for facility location as shown in Figure 5.5. Partition 5 at distance r which is the radius of given circle. We have to find an optimal distance in given partition to serve efficiently to all demand nodes. Let Area of each partition $= A/K$. And d is the covering distance as shown in circle 5 of Figure 5.5. Therefore all demand nodes in each cluster will lie under the distance d, by geometry $r < A/K$ and by our algorithm $d \leq r$.

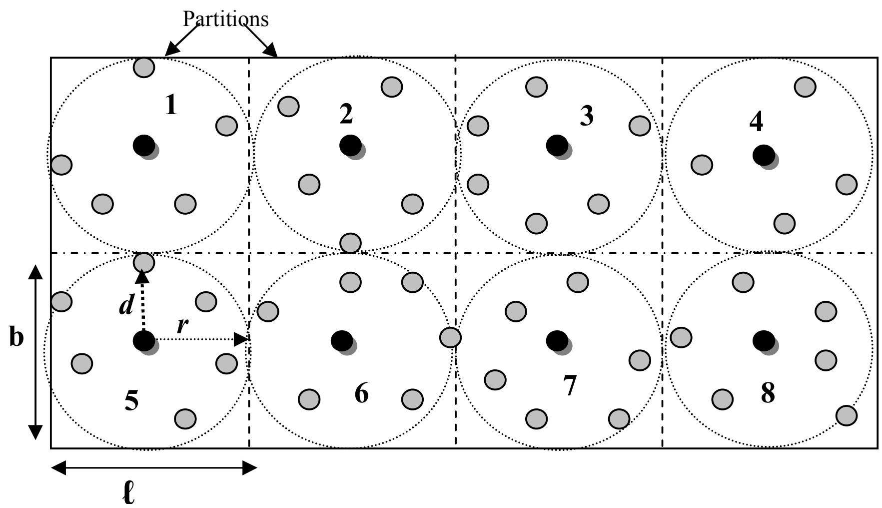


Figure 5.5: Demonstration of Cluster/Partition creation process and showing how a facility at optimal distance can serve all connected nodes equally.

Consequently, it is justified to say that the resultant of this algorithm, d provides such facility locations which are accessible within minimum travelling distance and available in minimum time (i.e. high response time). Since $\| x_i - y_j \| \leq d$. Where x_i represents the candidate facility location and y_j demand nodes, keeping i fixed and value of j varies. In other words, we can say that $d_{xy} \leq d$

Thus d is the optimal distance value for locating a facility which can efficiently serve all the connected demand nodes with minimum traveling distance. Another important factor is the cost of a facility that really does matter. But it is of concern only in case of new facilities and not the existing ones, because here we are discussing mobile facilities and relocation of mobile facilities does not involve any cost issues.

Relocation

The relocation of facilities is one of the important issues in incremental variant of facility location problem. Since the demand arrives one by one, it is significant to find out whether the relocation of existing facilities is obligatory or not on a new arrival. Let us confirm how the relocation issue is dealt by the presented algorithm. The first step of algorithm takes accountability of ensuring relocation. The distance of newly arrived demand node to its nearest facility dx_iy will be checked for relocation purpose.

Statement 2: if $dx_iy \leq d$, then demand will be assigned to the nearest facility x_i, where $x \in X$, $1 \leq i \leq k$, and no relocation will take place. Where y is the newly arrived demand node and x_i is the nearest facility to y, dx_iy is the distance among them. And d is the optimal distance value using which the facilities were allocated. As soon as the new demand arrives immediately our algorithm checks the distance of incoming demand node to its nearest facility and if it is less than the optimal distance d then this demand node will be assigned to the nearest facility and no relocation will take place.

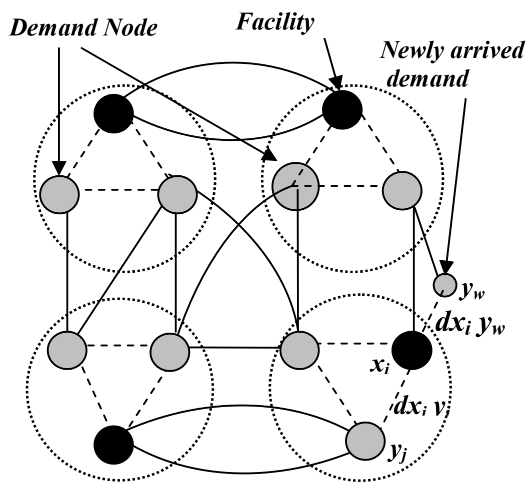


Figure 5.6: Illustrating how incremental demand is entertained by the proposed algorithm

We may point out that without any exhaustive exercise; the demand is accommodated through appropriate facility as illustrated in Figure 5.6. According to the statement 1 unambiguously, the facilities allocated using optimal distance d are located at their best possible places. This distance d is undoubtedly competent to minimize the traveling distance between facility and demand nodes. Similarly, statement 2 exhibits the meticulous behavior of algorithm which is helpful in avoiding the unnecessary relocation of facilities.

The above analysis reveals that the proposed framework endows novel approach for incremental mobile facility location problem. The major issue of minimization of traveling distance can be resolved efficiently by this algorithm in quite a professional way. Moreover, all allied issues like relocation of facilities and minimization of response time stand to get resolved automatically. The beauty of this algorithm lies in its simplicity and ease of implementation. Fault tolerance is one of the major correlated issues in case of mobile facilities, and is an open problem for future researchers.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we have presented our attempt towards designing new approaches for facility location problem. The presented study has focused to design an algorithm for the vertex *k*-centre problem that is a sub problem of facility location problem. Objective of vertex *k*-center problem is to minimize the maximum distance between a facility and demand node. The presented framework provides three approaches to solve the different variants of facility location problems.

Generally, facilities are located at prime locations but when we have a constraint that limited numbers of facilities are to be located in a wide region. In such realistic situations, it becomes a challenge for the decision makers to accommodate the specified demand within approved number of facilities. There are various parameters (such as distance, access time, availability of site, cost of building a facility, and population etc) that should be investigated to find out the appropriate location in given region or network of nodes. The presented approaches have made it possible to solve the abovementioned problems in a simplified and fast way.

In the first approach, we have proposed a framework consisting of Anticipatory Bound Selection Procedure (***ABSP***) accompanied by Jump Based Scheme (***JBS***) to solve the vertex *k*-centre problem. Even though Daskin has given an approach to solve the vertex *k*-center problem, but the main drawback of this approach is that it uses binary method to obtain the initial threshold distance. Our approach provides an efficient way to calculate the initial threshold distance through anticipatory bound selection procedure (***ABSP***). The major achievement of ***ABSP*** is that it produces a set of eligible distance values from the distance matrix which is directly helpful in eradicating the gap between lower and upper bound.

Additionally the jump based scheme (***JBS***) provides an effective way to speed up the processing of algorithm. It directly reduces the number of iterations required to find out the optimal distance value and determine the required number of appropriate facility locations on a given network of nodes. Consequently both the methods implemented gracefully to achieve the pre-defined objectives. A simulator is developed in C++ programming language. More than 30 problems with various numbers of nodes and facilities are tested on this simulator simultaneously for proposed approach and Daskin's approach.

Further, the results obtained and iteration frequency analysis reveals that a combination of ***ABSP*** and **JBS** made the algorithm very efficient. Since the proposed method has taken minimum two iterations to solve an assignment and most of the cases are executed within 2-5 iterations. On the other hand, ***Dakin's*** algorithm takes at least **5** iterations to execute any of the case. The cumulative frequency percentage as shown in Table 4.2 and Table 4.3 reveals that the Daskin's algorithm completes only **8.82%** of jobs within the range of 5 iterations; while the proposed algorithm completes **52.95%** of task.

It is noticeable in both the tables that there is no iteration less than **5** in the first column of Daskin's table while in ***ABSP*** there is the minimum iteration number that is **2** with frequency **5**. Moreover the cumulative frequency of iteration **4** is **16** in ***ABSP*** table it's evidence that more than **47%** of work is done by the proposed algorithm by exhausting at most of **4** iterations.

Eventually, the presented framework proves that the anticipatory bound selection procedure (***ABSP***) and ***JBS*** has shown a considerable improvement over the existing methodology to solve the aforementioned problem.

In the second approach, it emphasizes on to solve the Obnoxious Facility Location Problem (***OFLP***) for *1*-center only. Basically obnoxious facilities are those which are never desired to be located closer to the populated areas due to their adverse effects on human life. So such facilities are always required to locate away from the populated areas. It provides a simplified approach to locate the obnoxious facilities.

The main objective of ***OFLP*** is to maximize the minimum distance between facility and all connected nodes. The presented approach introduces a simplified method to achieve the objective mentioned above. It selects the maximum distance value out of a set of minimum distance values of each row of distance matrix. This selected distance value helps us to identify that particular node which is the most eligible candidate site for the obnoxious facility. More than 50 problems have been tested for a range of nodes from 8 to 98. Each case has been repeated three times to check the accuracy of time taken by the algorithm for allocation. A gradual increase in time has been noticed with the increase of nodes as shown in Table 4.5. Computation times are expressed in CPU seconds Program executed on dual core 1.6 GHz Microprocessor with 1GB RAM and code is written in Turbo C++.

Our third approach concentrates on Mobile Facility Location Problem (***MFLP***) and provides an improved approach for mobile facility location problem in a dynamic environment. Since it is again a challenge for decision makers to allocate a facility in such environment where various parameters (e.g. population or demand) are frequently used to change with the time. Presented framework provides a methodology to allocate the mobile facilities (such as Fire Brigade, PCR vans, and Ambulance) keeping in view all the important aspects of these facilities.

Increased population over a period of time and access time of such facilities are the most important parameter that has been considered during this study. Then ***ABSP*** and ***JBS*** are used to speed up the processing of algorithm.

6.2 Limitations and Future Directions

The presented work has a few limitations; these limitations may lead to many research directions through which the presented work can be extended.

In the first approach both methods Anticipatory Bound Selection Procedure (***ABSP***) and Jump Based Scheme (***JBS***) are used to speed up the process. Still the time complexity may be improved. Therefore this is an open problem for future work related to our first approach.

As far as, the second approach is concerned; which provides a simplified approach for obnoxious facility location problem. The presented approach provides a solution for single center; it can be solved for k centers (more than one center, where k is a non negative integer). Additionally, routing and transportation of obnoxious materials is also an open problem for future work.

Finally, the third approach provides an improved approach for incremental variant of mobile facility location problem. It addresses a few parameters such as demand (population), distance and access time. The cost of facility may be an important parameter for allocation of facilities. So cost may be considered as major parameter for future research work.

References

[1] M. S. Daskin, *Network and Discrete Locations: Models, algorithms, and applications*, John Wiley & Sons, New York, pp 154-191, 1995.

[2] J. Stollsteimer, *A working model for plant numbers and locations*, Journal of Farm Economics, vol. 45, pp. 631-645, 1963.

[3] A. A. Kuehn and M. J. Hamburger, *A heuristic program for locating warehouses*, Management Science, vol. 9, pp. 643-666, 1963.

[4] J. M. Mulvey and H. L. Crowder, *Cluster Analysis: An application of Lagrangian Relaxation*, Management Science, vol. 25, pp. 329-340, 1979.

[5] M. S. Daskin, *What You Should Know About Location Modeling,* Wiley Periodicals, Inc., pp: 283-294, 2008.

[6] J. J. Sylvester, *A question in the geometry of situation*, Quarterly Journal of Mathematics, pp. 1-79, 1857.

[7] A. Weber, *Theory of the Location of Industries*, University of Chicago Press, Chicago, 1909.

[8] B. Ben-Moshe, M. J. Katz, M. Segal, *Obnoxious Facility Location: Complete Service with Minimal Harm*, International Journal of Computational Geometry and Applications, vol. 10, pp. 581-592, 2000.

[9] S. Bespametnikh, K. Kedem, M. Segal, A. Tamir, *Optimal Facility Location Under Various Distance Functions*, International Journal of Computational Geometry And Applications, vol. 10, pp. 523-534, 2000.

[10] Emerson, Selwyn Piramuthu, Denise, *Agent-based framework for dynamic supply chain configuration*, System Sciences, Proceedings of the 37th Annual Hawaii International Conference on IEEE, 2004.

[11] M. J. Katz, K. Kedem, M. Segal, *Improved Algorithm for Placing Undesirable Facilities*, Computers and Operations Research, vol. 29, pp. 1859-1872, 2002.

[12] L. Cooper, *A Random Locational Equilibrium Problem*, Journal of Regional Science, vol. 14(1), pp. 47-54, 1974.

[13] S. L. Hakimi, *Optimum Locations of Switching Centers and the Absolute Centers and Medians of Graph*, Operations Research, vol. 12(3), pp. 450-459, 1964.

[14] S. L. Hakimi, *Optimum Distribution of Switching Centers in a Communication Network and some related graph theoretical Problems*, Operations Research, vol. 13(3), pp. 462-475, 1965.

[15] P. B. Mirchandani, A R Odoni, *Locations of Medians on Stochastic Networks*, Transaction Science, vol. 13(2), pp. 85-97, 1979.

[16] P. B. Mirchandani, *Locational Decisions on Stochastic Networks*, Geographical Analysis, vol. 12(2), pp. 172-183, 1980.

[17] E.S. Sheppard, *A Conceptual Framework for Dynamic Location-Allocation Analysis*, Environment and Planning A, vol. 6, pp. 547-564, 1974.

[18] P. B. Mirchandani, A. Oudji, *Localizing 2-Median on Probabilistic and Deterministic Tree Networks,* Networks, vol. 10(4), pp. 329-350, 1980.

[19] V. Kumar, M. Hsu. *A Superior Two-Phase Locking Algorithm and its Performance*, Information Sciences An International Journal, Vol. 54, No 1,2. pp: 147-168, 1991.

[20] J. Hershberger, *Minimizing the sum of diameters efficiently*, Computational Geometry: Theory and Applications, vol. 2, pp. 111-118, 1992.

[21] J. Hershberger, *A Faster Algorithm for the Two-Center Decision Problem*, Information Processing Letters, vol. 47, pp. 23-29, 1993.

[22] A. Marchetti-Spaccamela, *The p-Center Problem in the Plane is NP-Complete*, Proc. Of the 19th Allerton Conference on Communication Control and Computers, pp. 31-40, 1981

[23] D. S. Hochbaum, D. B. Shmoys, "*A Best Heuristic for the k-center Problem*". Math. Operations Research, vol. 10, pp 180-184, 1985.

[24] J. Plesnik, "*A Heuristic for The p-center Problem in Graphs*". Discrete Applied Mathematics, North Holland, vol. 17, pp 263-268, 1987.

[25] J. Plesnik, "*Two Heuristics for the Absolute p-Center Problem in Graphs*". Math. Slovaca, vol. 38, No. 3, 227-233, 1988.

[26] A. Glozman, K. Kedem and G. Shpitalnik, On Some Geometric Selection and Optimization Problems via Sorted Matrices, Proc. Of the 4th Workshop on Algorithms and Data Structure, LNCS 995, pp. 26-37, 1995

[27] P.K. Agarwal, M. Sharir, *Efficient Algorithms for Genetic Optimization*, ACM Computing Surveys, vol. 30, pp. 412-458, 1998.

[28] B. Annappa, L Thomas. *Application of Parallel K-Means Clustering Algoithm for Prediction of Optimal Path in Self Aware Networks with Link Stability*, Advances in Computing & Communication, Springer Berlin Heidelberg, pp396-405, 2011.

[29] R. Z. Hwang, R. C. Chang and R. C. T. Lee, *The Searching Over Separators Strategy to Some NP-hard Problems in Subexponential Time*, Algorithmica, vol. 9, pp. 398-423, 1993.

[30] R. Z. Hwang, R. C. Chang and R. C. T. Lee, *The Slab Dividing Approach to Solve the Euclidean P-Center Problem,* Algorithmica, vol. 9, pp. 1-22, 1993.

[31] S. Bespametnikh, D. Kirkpatrick, *Rectilinear 2-center problem*, Proc. of the 11th Canadian Conference on Computational Geometry, pp. 68-71, 1999.

[32] T. M. Chan, *More Planar two-center Algorithms*, Computational Geometry: Theory and Applications, vol. 13, pp. 189-198, 1999.

[33] D. Eppstein, *Faster Construction of Planar two-centers*, Proc. of the 8th ACM-SIAM Symposium on Discrete Algorithms, pp. 131-138, 1997.

[34] J. W. Jaromczyk, M. Kowaluk, *An Efficient Algorithm for the Euclidean two-center Problem*, Proc. of the 10th Annual ACM Symposium on Computational Geometry, pp. 303-311, 1994.

[35] M. J. Katz, K. Kedem, M. Segal, *Discrete Rectilinear 2-center Problems*, Computational Geometry: Theory and Applications, vol. 15, pp. 203-214, 2000.

[36] M. Sharir, *A Near-Linear Algorithm for the Planar 2-center Problem*, Discrete Computational Geometry, vol. 18, pp. 125-134, 1997.

[37] J. Elzinga and D. W. Hearn, *Geometrical Solutions to Some Minimax Location Problems*, Transportation Science, vol. 6, pp. 379-394, 1972.

[38] M. I. Shamos and D. Hoey, *Closet-point Problem*, Proc. of the 16th Annual IEEE Symposium Foundation Computer Science, pp. 151-162, 1975.

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

[40] M. I. Shamos, *Computational Geometry*, Ph.D. Thesis, Department of Computer Science, Yale University, New Haven, CT, 1978.

[41] S K Gupta, Ashish Gupta, NG Rao, *A Network Based Model for Armed Forces Medical Stores Depot*, Conf. On Medical Informatics, CSI, Srinagar, 1982.

[42] N. Megiddo, *Linear-time Algorithms for Linear Programming In R3 and Related Problems*, SIAM Journal on Computing, vol. 12, pp. 759-776, 1983.

[43] F. Hurtado, V. Sacristan and G. Toussaint, *Facility Location Problems with Constraints*, Studies in Locational Analysis, vol. 15, pp. 17-35, 2000.

[44] P.K. Agarwal, M. Sharir and E. Welzl, *The Discrete 2-center Problem*, Proc. of the 13th Annual ACM Symposium on Computing Geometry, pp. 147-155, 1997.

[45] S. K. Kim, 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, pp. 299-309, 2000.

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

[47] O. Kariv, S. Hakimi, *An Algorithmic Approach to Network Location Problems Part-I: The P-Centers*, SIAM Journal of Applied Mathematics, vol. 37, pp. 513-538, 1979.

[48] E. Minieka, "*The m-center problem*", SIAM Review 12, pp. 138-139, 1970.

[49] C. Toregas, R. Swain, C. ReVelle, and L. Bergman, *The Location of Emergency Service Facilities*, Operations Research, vol. 19, pp. 1363-1373, 1971.

[50] R.L. Church, C. ReVelle, *The Maximal Covering Location Problem*, Papers of the Regional Science Association, vol. 32, pp. 101-118, 1974.

[51] N. Megiddo, E. Zemel, and S. L. Hakimi, *The Maximum Coverage Location Problem*, SIAM Journal of Algebraic and Discrete Methods, vol. 4, pp. 253-261, 1983.

[52] E. Minieka, *The Centers and Medians of a Graph*, Operations Research, vol. 25, pp. 641-650, 1977.

[53] R. S. Garfinkel, A. W. Neebe, and M. R. Rao, *The m-center Problem: Minimax Facility Location*, Management Science, vol. 23, pp. 1133-1142, 1977.

[54] Z. Drezner, *The p-Center Problem-Heuristic and Optimal Algorithms*, Journal of Operational Research Society, vol. 35(8), pp. 741-748, 1984.

[55] M. E. Dyer, A. M. Frieze, A Simple H*euristic for the p-Center Problem*, operations Research Letters, vol. 3(6), pp. 285-288, 1985.

[56] J. Flynn, S. Ratick, *A Multi-objective Hierarchical Covering Model for the Essential Air Services Program*, Transportation Science, vol. 22, pp. 139-147, 1988.

[57] P. Kolesar, W. E. Walker, *An Algorithm for the Dynamic Relocation of Fire Companies*, Operations Research, vol. 23, pp. 249-274, 1974.

[58] W. E. Walker, *Using the Set Covering Problem to Assign Fire Companies to Fire Houses*, Operations Research, vol. 22, pp. 275-277, 1974.

[59] D. R. Plane, T. E. Hendrick, *Mathematical Programming and the Location of Fire Companies for the Denver Fire Department*, Operations Research, vol. 25, pp. 563-578, 1977.

[60] S. Belardo, J. Harrald, W. A. Wallace, and J. Ward, *A Partial Covering Approach to Siting Response Resources for Major Maritime Oil Spills*, Management Science, vol. 30, pp. 1184-1196, 1984.

[61] D. Eaton, M. S. Daskin, D. Simmons, B. Bulloch, and G. Jansma, *Determining Emergency Medical Service Vehicle Deployment in Austin*, Texas, Interfaces, vol. 15(1), pp. 96-108, 1985.

[62] R. Batta, N. Mannur, *Covering Location Models for Emergency Situations That Require Multiple Response Units*, Management Science, vol. 36, pp. 16-23, 1990.

[63] D. A. Schilling, V. Jayaraman, and R. Barkhi, *A Review of Covering Problems in Facility Location*, Location Science, vol. 1, pp. 25-55, 1993.

[64] C. ReVelle, K. Hogan, *The maximum reliability location problem and a-reliable P-center problem: Derivatives of the probabilistic location set covering problem,* Annals of Operations Research, vol. 18, pp. 155-174, 1989.

[65] G. Y. Handler, P. B. Mirchandani, *Location on Networks*, M.I.T. Press, Cambridge, MA, 1979.

[66] G. Y. Handler, *p-center Problems, in Discrete Location Theory*, John Wiley Inc., New York, Chapter 7, pp. 305-347

[67] M. S. Daskin, *A New Approach to Solve the Vertex P-Center Problem to Optimality: Algorithm and Computational Results,* Communications of the Operations Research Society of Japan, vol. 45(9), pp. 428-436, 2000.

[68] S. Elloumi, M. Labbe, and Y. Pochet, *New Formulation and Resolution Method for the P-center Problem*, 2001 http:/ www. Optimization-online.org/DB_HTML/2001/10/394.html.

[69] T. Ilhan, and M. C. Pinar, *An Efficient Exact Algorithm for the Vertex p-center Problem*, 2001.

[70] A. Al-khedhairi, S. Salhi, *Enhancement to Two Exact Algorithms for Solving the Vertex P-center Problem*, JMMA, vol. 2(2), pp 129-147, 2005.

[71] A. Al-khedhari, *An Enhancement of Daskin's Algorithm for Solving p-center Problem*, Journal of Approximation Theory and Applications, vol. 2(2), pp 121-131, 2007.

[72] Q. Z. Wang, K. H. Cheng, *Parallel Time Complexity of a Heuristic Algorithm for the k-Center Problem with Usages Weights,* Proc. of the 2nd IEEE Symposium on Parallel & Distributed Processing, pp. 254-257, 1990.

[73] R. Chen, G. Y. Handler, *The Conditional p-Center Problem in Plane*, Naval Research Logistics, vol. 40, pp. 117-127, 1993.

[74] D. S. Hochbaum, A. Pathria, *Generalized p-Center Problems: Complexity Results and Approximation Algorithms*, European Journal of Operational Research, vol. 100, pp. 594-607, 1997.

[75] S. Chaudhuri, N. Garg, R. Ravi, *P-neighbor k-center Problem*, Information Processing Letters , vol. 65(3), 131-134, 1998.

[76] D. B. Shmoys, E. Tardos, K. Aardal, *Approximation Algorithms for Facility Location Problems*, Proc. of the 29th Annual ACM Symposium on the Theory of Computing, pp. 265-274, 1997.

[77] K. Jain, M. Mahdian, A. Saberi, *A New Greedy Approach for Facility Location Problems*, Proc. STOC'02 Proceedings of the 34th Annual ACM Symposium on Theory of Computing, pp. 731-740, 2002.

[78] A. B. Arabani, and R. F. Zanjirani, *Facility Location Dynamics: An Overview of Classifications, Models, Solutions, and Applications*, Computers & Industrial Engineering, vol. 62(1), pp. 408-420. ISSN (print) 0360-8352, 2012.

[79] D. Fotakis, *Incremental Algorithms for Facility Location and k-Median*, Lecture Notes in Computer Science, vol. 3221/2004. pp. 347-358, 2004.

[80] T. Gonzalez, *Clustering to Minimize the Maximum Inter-cluster Distance*, Theoretical Computer Science, vol. 38, pp. 293-306, 2005.

[81] G. Lin, C. Nagarajan, R. Rajaraman, D. P. Williamson, *A General Approach for Incremental Approximation and Hierarchical Clustering*, SIAM Journal of Computing, vol. 39, pp. 3633-3669, 2010.

[82] R. R. Mettu, C. G. Plaxton, *The Online Median Problem*, SIAM Journal on Computing, vol. 32, pp. 816-832, 2003.

[83] C G. Plaxton, *Approximation Algorithm for Hierarchical Location Problems*, Journal of Computer & System Sciences, vol. 70, pp. 425-443, 2006.

[84] M. Charikar, C. Chekuri, T. Feder, and R. Motwani, *Incremental Clustering and Dynamic Information Retrieval*, SIAM Journal of Computing, pp. 1417-1440, 2004.

[85] S. Dasgupta, P. Long, *Performance Guarantees for Hierarchical Clustering,* Journal of Computer and System Sciences, vol. 70, pp. 555-569, 2005

[86] J. Hartline, Alex Sharp, *An Incremental Model for Combinatorial Minimization*, International Workshop on Experimental Algorithms, Menorca Island, Spain, 2006

[87] Z. Friggstad, R. Salavatipour, *Minimizing Movement in Mobile Facility Location Problems*, ACM Transactions on Algorithms, Vol. 7 (3): 28, 2011.

[88] R. M. McCutchen, S. Khuller, *Streaming Algorithms for k-Center Clustering With Outliers and With Anonymity*, Workshop on Approximation Algorithms (APPROX), 2008

[89] M. Albareda-Sambola, E. Fernandez, Y. Hinojosa, J. Puerto, *The Multi-Period Incremental Service Facility Location Problem*, Journal of Computer and Operation Research, vol. 36(5), 2009.

[90] D. Chen, R. Chen, *New Relaxation-Based Optimal Algorithms for the Solution of the Continuous and Discrete p-Center Problems*, Computers and Operations Research, vol. 36, pp. 1646-1655, 2009.

[91] J. H. Lin, J. S. Vitter, *ε-Approximations with Minimum Packing Constraint Violation*, Proc. of the 24th Annual ACM Symposium on Theory of Computing, pp. 771-782, 1992.

[92] T. S. Hale, C. R. Moberg, *Location science research: a review,* Annals of Operations Research, vol. 123, pp. 21-35, 2003.

[93] R. L. Church, T. L. Bell, *An Analysis of Ancient Egyptian Settlement Patterns Using Location-Allocation Covering Models*, Ann Assoc Am Geog 78, pp. 701–714, 1988.

[94] M. R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Free man and Co., New York, 1979.

[95] M. J. Kuby, *The p-Dispersion and Maximum Dispersion Problems*, Geographical Analysis, vol. 19, pp. 315-329, 1987.

[96] M. E. O'Kelly, *Activity Levels at Hub Facilities in Interacting Networks*, Geographical Analysis, vol. 18, pp. 343-356, 1986.

[97] M. E. O'Kelly, *The location of interacting hub facilities*, Transportation Science, vol. 20, pp. 92- 106, 1986.

[98] J. F. Campbell, *Locating Transportation Terminals to Serve an Expanding Demand*, Transportation Research, 24B, pp.173-192, 1990.

[99] J. F. Campbell, *Integer Programming Formulations of Discrete Hub Location Problems*, European Journal of Operational Research, vol. 72, pp. 387-405, 1994.

[100] M. H. J. Webb, *Cost Functions in the Location of Depot for Multiple-Delivery Journeys*, Operational Research Society, vol. 19, pp. 311-320, 1968.

[101] S. Eilon, C. D. T. Watson-Gandy, N. Christofides, *Distribution Management: Mathematical Modelling and Practical Analysis*, Griffin, London, 1971.

[102] J. Perl, M. S. Daskin, *A warehouse location-routing model*, Transportation Research, 19B, pp. 381-396, 1985

[103] J. R. Current, *The Design of a Hierarchical Transportation Network with Transshipment Facilities*, Transportation Science, vol. 22, pp. 270-277, 1988.

[104] J. R. Current, H. Pirkul, *The Hierarchical Network Design Problem with Transshipment Facilities*, European Journal of Operational Research, vol. 51, pp. 338-347, 1991.

[105] H. B. Fisher, *Spatial Efficiency of Service Locations and the Regional Development Process Papers,* Regional Science Association, vol. 42, pp. 83–97, 1979.

[106] J. L. Cohon, *Multiobjective Programming and Planning,* New York, NY: Academic Press, 1978.

[107] D. A. Schilling, *Dynamic Location Modeling for Public Sector Facilities: A Multi-criteria Approach*, Decision Sciences, vol. 11, pp. 714–724, 1980.

[108] R. L. Church, *A Bi-criterion Maximal Covering Location Problems Which Considers the Satisfaction of Uncovered Demand*, Decision Sciences, vol. 22, pp. 38–52, 1991.

[109] A. S. Manne, *Capacity Expansion and Probabilistic Growth*, Econometrica, vol. 29, pp. 632-649, 1961.

[110] A. S. Manne, *Investments for Capacity Expansion: Size, Location and Time Phasing*, MIT Press, Cambridge, MA, 1967.

[111] R. H. Ballou, *Dynamic Warehouse Location Analysis*, Journal of Marketing Research, vol. 5, pp. 271-276, 1968.

[112] J. Current, S. Ratick, C. ReVelle, *Dynamic Facility Location When the Total Number of Facilities is Uncertain: A Decision Analysis Approach*, European Journal of Operational Research, vol. 110, pp. 597-609 1998.

[113] H. Frank, *Optimum Locations on a Graph with Probabilistic Demands*, Operations Research, vol. 14, pp. 409-421, 1966.

[114] O. Berman, R. C. Larson, S. S. Chiu, *Optimal Server Location on a Network Operating as an M/G/1 Queue*, Operations Research, vol. 33, pp. 746-771, 1985.

[115] M. S. Daskin, *Application of an Expected Covering Model to Emergency Medical Service System Design*, Decision Sciences, vol. 13, pp. 416-439, 1982.

[116] J. C. Bean, J. L. Higle, R. L. Smith, *Capacity Expansion Under Stochastic Demands*, Operations Research, vol. 40, pp. 210-216, 1992.

[117] S. C. Chapman, J. A. White, *Probabilistic formulations of emergency service facilities location problems*, Paper Presented at the ORSA/TIMS Conference, San Juan, Puerto Rico, 1974.

[118] K. Van Der Heijden, *Probabilistic Planning and Scenario Planning*, Subjective Probability, eds. G. Wright and P. Ayton, Wiley, New York, pp. 549-572, 1994.

[119] J. H. Vanston, W. P. Frisbie, S. C. Lopreato, D. L. Poston, *Alternate Scenario Planning*, Technological Forecasting and Social Change, vol. 10, pp. 159-180, 1977.

[120] Z. Drezner, H. W. Hamacher, *Facility Location Applications and Theory*, sSpringer, New York, 2002.

[121] M. B. Teitz, P. Bart, *Heuristic Methods for Estimating Generalized Vertex Median of a Weighted Graph*, Operations Research, vol. 16, pp. 955-961, 1968.

[122] J. Mihelic, B. Robic, *Solving the k-Center Problem Efficiently with a Dominating Set Algorithm,* Journal of Computing and Information Technology-CIT 13, 2005, vol. 3, pp 225-233, 2005.

[123] J. Mihelic, B. Robic, *Approximation Algorithms for the K-Center Problem: An Experimental Evaluation,* Proc. of Operations Research, Klagenfurt, Austria, 2002.

[124] E Demaine, M Hajiaghayi, H Mahin, A Sayid-Roshkar, S Oveisgharan, M Zadimoghaddam. *Minimizing Movement,* In Proceeding of the 18th annual ACM Symposium On Theory of Computing (STOC). (2003).

[125] J Puerto, F Ricca, A Scozzari. *Unreliable Point Facility Location Problems on Networks*, Discrete Applied Mathematics, Vol. 166, pp.188-203, 2013.

[126] X Wang, M K Lim, Y Ouyang. *A Continuum Approximation Approach to the Dynamic Facility Location Problem*, Transport Research Board 93rd Annual Meeting, Washington DC, January 2014.