11-20-2003

Analytical Strategies in Deciding Bus Route Alignments

Sandeep Seshan Iyer
University of South Florida

Scholar Commons Citation

Iyer, Sandeep Seshan, "Analytical Strategies in Deciding Bus Route Alignments" (2003). *Graduate Theses and Dissertations.*
https://scholarcommons.usf.edu/etd/1400

Analytical Strategies in Deciding Bus Route Alignments

by

Sandeep Seshan Iyer

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Industrial Engineering
Department of Industrial Engineering and Management Systems
College of Engineering
University of South Florida

Major Professor: Grisselle Centeno, Ph.D.
Steven Polzin, Ph.D.
Jose-Zayas Castro, Ph.D.

Date of Approval:
November 20, 2003

Keywords: accessibility, grid network, route design, activity levels, forward searching heuristic

DEDICATION

This work is dedicated to my parents Nurani Arunachalam Seshan and Annapurna Seshan and my brother Iyer Pradeep Seshan.

ACKNOWLEDGEMENTS

I wish to thank a number of people who have contributed to the making of this thesis. First of all, I would like to thank my major professor – Dr. Grisselle Centeno for her support and belief in me. A special thanks to Dr. Steven Polzin for going out of the way to encourage me and keep me ever motivated in my endeavour inspite of his tight schedule. Thanks to Dr. Jose-Zayas Castro for agreeing to be on my committee.

This acknowledgement would be incomplete without mentioning the names of my friends who have supported me all along the way. A special thanks to Sathya, Sanket, Ram Nehra, Mahesh Chettiar, Pinglikar, Panda, Ashok, Gupte, Sanjay, Utpal, Pummy, Shiny, Matti, Kristin, Chris, Pranav and students from the Industrial Engineering lab. A special word of gratitude to Sandhya Srinivasan, Blondeane, Dr Gulitz and the College of Public Health Family, which made me feel at home away from home. Finally I express sincere thanks to my family for their continued support over the years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ANALYTICAL STRATEGIES IN DECIDING BUS ROUTE ALIGNMENTS

Iyer Sandeep Seshan

ABSTRACT

In this research a heuristic algorithm is developed for searching and identifying preferred actions as applied to the bus route design problem. The search routine evaluates each subsequent segment added to the route in the context of the value of that segment and also the value of future decisions and opportunities for subsequent segments. The total overall maximum accessibility of the system is calculated using a minimum path network between each node pair and adding the accessibility of all route segments. This is equivalent to assuming that there was a direct shortest path route between every two destinations in the network. The quality of the designed network is obtained by comparing the share of the total benefits obtained from the heuristic with the share of the costs incurred with respect to a minimum path network. Several test cases and network scenarios are studied to evaluate the analytical tool developed. In addition, different performance measures are used to identify the connecting routes that increase the accessibility of the system.

CHAPTER 1. INTRODUCTION

A transportation system is one of the basic components of an urban area's social, economic and physical structure. One of the major challenges being faced today is ensuring that cities have operational, economical and efficient public transportation, which enhances their environment, reduces congestion, conserves energy and fulfills the daily transportation needs of the general public. Hence, operation of buses and their routing is one of the potential areas of study. The efficiency of the bus transportation system depends on the network of routes and the frequency of the buses (Dhingra et al., 2000). These routes are generally designed taking into account both passenger and operator interests. According to (Shih et al., 1998) many of the problems presently being encountered by the transit providers originate from economic sources, shortage of manpower and equipment. Hence, optimal routing and scheduling will contribute in alleviating these problems. This thesis is motivated by the route network design problem faced by transit planners while identifying new routes for service.

1.1 Bus Planning Process

The bus planning process includes network design, setting frequencies, timetable development, bus scheduling and driver scheduling. It is a systematic decision sequence as illustrated in the Table 1 (Ceder Wilson, 1986). The table gives a clear summary of the independent inputs, planning activities and the outputs. Independent inputs are a set of variables on which the planning activities depend. Demand data, schedule constraints, running time of buses etc are the inputs which help planning activities like network design, timetable development and scheduling in deciding outputs. The outputs here are better routes, frequencies and schedules. The output of each activity positioned higher in the planning sequence becomes an important input for lower level decisions. Thus, decisions made further down in the sequence have some effect on the higher-level decisions. On the whole, the bus planning process is an optimization problem at all the different levels as outlined in Table 1. Generally the five elements considered for improvement are: 1) routing 2) service frequencies 3) trip arrival and departure times 4) bus schedules and 5) driver schedules. A number of external and operational factors are involved in the design of the bus network such as financial, socioeconomic, political, etc. Increased urbanization has led to the requirement of an infrastructure and a huge economic burden to develop a working infrastructure. Thus, the reduction in costs for the service provider is important. The transit service provider has to bring about a balance between reduction of capital cost and the operating cost of buses over the period of operation, as well as minimize travel time, waiting time and transfer times for the customers. Therefore, optimal design of the route for quality of service offered is essential. Research in transit planning has focused on maximizing economy of resources and at the same time maximizing functionality for users (Bielli et al., 1998).

Table 1: Bus Planning Process

Independent Inputs	Planning Activity	Output
Demand Data Supply Data	Level A	New Routes Route Changes
Subsidy Available Patronage	Level B Setting Frequencies	Service Frequencies
Running Times Trip Timings	Level C Timetable Development	Trip Arrival Trip Departure
Schedule Cost Constraints	Level D Bus Scheduling	Bus Schedules
Driver Work Rules Run Cost	Level E	Driver Schedules

1.2 Problem Description

Every transit agency has to determine which areas in its jurisdiction to serve and how to design routes to operate its service within a city. The city can be characterized as a set of different zones. The transit agency has to decide which zones to connect with service so as to maximize the potential of ridership, subject to a fixed budget. Ridership means the number of people using the service. If the ridership is increased, then the value of the route is also increased. Each zone has a set of characteristics or activities associated with it. These activities determine the levels of travel that might use the zone. These characteristics are related to variables such as: employment, population, and annual income, which determine the activity levels and demand for transportation. Travelers are expected to use a route connecting two zones in some proportion to the activity levels in each of the zones and the cost or impedance of traveling between them.

The goal of the thesis is to find out the areas to be served by the transit service in such a way that the value of the route is maximized. The critical element will be to find a way to identify combinations of sequential links that comprise logical bus routes. The gist of the entire process is to determine the overall maximum possible benefit of serving all demands by calculating the minimum path network between each zone pair and summing the benefits. A forward searching heuristic is proposed to decide the subsequent route segments. The quality of the designed network is obtained by comparing the share of total benefits realized against the share of costs incurred to that of an ideal network wherein each zone is connected to all other zones.

1.3 Proposed Approach

1. In this research a forward searching heuristic is put forth which helps deciding subsequent segments in the route. The formation of routes is based on the attractiveness to travel between zones and the distance between them. This is in turn based on socio economic characteristics like employment, population and annual income and route distance. The search procedure evaluates each subsequent segment in the context of not only the value of the addition of that segment to the route but also the value of each decision in terms of future opportunities of subsequent segments.
2. A shortest path algorithm is used to model this network problem, wherein nodes represent different zones and the links denote the attractiveness to travel between them. This representation helps to find bus routes in an ideal case scenario

wherein each zone is connected to every other zone. The system total accessibility is maximized in this case and the network so formed is optimal.

3. The quality of the route obtained by the forward searching heuristic is then measured with respect to the ideal network in terms of the accessibility added to the system.

1.4 Research Contributions

The contributions of this research are:

1. The current methodology of route formation is decided by historical data and travel trends. The model developed as part of this research will be an analytical tool which, if operationalized will be useful to transit providers in deciding priority areas to service within a city.
2. The formulation of the transportation routing problem is based on socio-economic characteristics like employment, population and annual income using network flow modeling as a framework.
3. The methodology so developed adds each subsequent segment to a route by taking into consideration not only the value of the added segment but also the value of each decision in terms of future opportunities of connecting subsequent segments. In simple words, a segment is added to a route by exploring all the subsequent opportunities arising from its addition.

1.5 Importance of the Research

This research is focused on a level preceding Network design (Level A) as outlined in Table 1. An analytical tool is put forth, which will aid transit providers in deciding routes for service based on the activity levels at each zone and the tendency to travel between zones based on the attractiveness (also referred to as benefit) to travel between them. The benefits of connecting any two zones are proportional to the activity levels at the nodes the population and employment and disproportional to the travel distance/cost between them. The activity levels considered here are the population and employment at the respective zones. Other socio-economic characteristics such as vehicle ownership, or attraction centers, can be considered as part of further research. Currently the methodology used to decide the routes is based on the trends of travel in each zone and approximately estimate the routes without considering the overall value of the route. The value of the route is the ratio of the benefits to the costs of the route. This research finds the value of all the output routes based on which service decisions are taken. This tool can be used by community bus service operators to decide on prospective routes of service based on analytical reasoning rather than deciding routes based on historical travel data.

1.6 Thesis Overview

This thesis is organized in six chapters. The second chapter gives a detailed review of literature. The third chapter provides a formal introduction to the problem at hand. The basics of network modeling have been explained, and a review on the concept of accessibility is provided. The solution methodology developed for solving this problem is described and explained in the fourth chapter. The experimental setup and results from the computational experiments are presented in the fifth chapter. Conclusions drawn from these results are listed. Finally a summary of the thesis work and directions for future research are provided in the sixth chapter.

CHAPTER 2. LITERATURE REVIEW

2.1 Introduction

This chapter presents an extensive literature review of the Route Network Design problem. Route Network Design (RND) is the single most important planning step in urban transit planning process as per (Baaj et. al, 1990). This is because the route network design will invariably affect both the frequency setting and the bus and crew scheduling. The important components of RND design are identified as estimating demand, identification of objective function, constraints, passenger behavior, solution techniques and computation time. Demand may be treated as fixed and independent of service quality. The transit network design problem seeks to configure a bus transit network consisting of a good set of routes and their frequencies. Mathematical formulations of the problem seek to minimize the generalized cost measure, usually a combination of user costs and operator costs. The user costs consist of access cost, waiting cost, and in-vehicle travel cost, whereas the total vehicle operating miles are used to estimate the operating cost. According to (Ceder and Wilson, 1986) the point of view of the user and operator with respect to service is listed as follows:

User's point of view –The public perceives a good bus network to be one that does not have too many routes, long or circuitous routes, or require many transfers.

Operator's point of view – The bus operator envisions a route as one that is perceived favorably by the public and at the same time does not require excessive resources that might be associated with complicated schedules or operational complexities.

In the last four decades several different kinds of models have been developed for the design of public transport network and it's scheduling. These models have used tools and techniques, from simple heuristics to complex simulations and evolutionary algorithms.

2.2 Basic Approaches

A few of the basic approaches are cited below. (Lampkins and Salmans, 1967) developed a heuristic algorithm to design transit network optimizing passenger–kilometer criteria. (Dhingra, 1980) proposed a heuristic approach for generating the transit route network wherein shortest routes are generated using a minimum path algorithm. The criterion for evaluating route alternatives included maximization of passenger – kilometers operated, average link density and the route utilization coefficient maximized over the various route alternatives. (Mandl, 1984) gave a heuristic algorithm to find the optimal routes such that a set of routes remain feasible, and there is a possible reduction in average cost using the set of routes. The new set of routes is compared to the older ones on basis of its performance and if found better, is accepted and the search procedure starts all over again until new improvements are found.

(Bansal, 1981) proposed a mathematical formulation, which minimizes total cost, both operating costs and user costs, for a fixed spatial and temporal network. (Marwah, et al.1995) presented a two level methodical approach for the design of a of bus network. The first level considers the passenger's viewpoint and the second level considers both the passengers and operators viewpoint. The first level approach has been be handled by an optimization program while the second level by a heuristic technique. (Baaj, et al.1991) developed a route generation algorithm that generates different sets of routes corresponding to different trade-offs between user and operator costs. Recent developments in network design have seen the evolution of techniques like Genetic Algorithms (GA), Artificial Neural Networks (ANN), and Artificial Intelligence based approaches. These approaches are discussed below in detail and sets of papers are reviewed to explain how the problem of Route Network Design has evolved over the years.

2.2.1 Two Level Approach

(Ceder and Wilson, 1986) presented an approach for design of routes from the operators and users point of view as mentioned earlier. The paper is divided into two levels: 1) level I formulation (objective function and constraints) considers only the passengers point of view.2) level II formulation considers both passenger's and operator's point of view. The objective functions in the case of level I is minimization of passenger hours, between route travel time and shortest possible route travel time while reducing total passenger transfer time subject to constraints. Level II formulation involves the minimization of number of vehicles used to operate the system, in addition to minimization of passenger hours as in level I. This paper discusses the route design problem taking into consideration both the user as well as operators point of view.

2.2.2 Genetic Algorithm Models in Transit Network Design

A Genetic Algorithm (**GA**) (Dhingra, 1980) is a local search algorithm, which works starting from an initial collection of strings (a population) that represent different solutions to the problem. Each string of the population is called a chromosome, and has an associated value called the fitness function (ff) that contributes in the generation of the new population by means of genetic operators (denoted crossover, reproduction and mutation, respectively). The genetic algorithm model uses the following steps to solve the RND design problem. In the first step a population P is randomly generated whose individuals represent a feasible solution. In the next step the individual members are evaluated to find the objective function value. In the third step the objective function is mapped to the fitness function that computes fitness for each member of the population. Individuals with higher fitness value will have a higher probability of being selected as candidates for further examination. Application of these operators like mutation, reproduction and crossover on the current population creates new operators. The basic step in finding out the best routes is given by the candidate route set algorithm as under:

1. Generate routes for every terminal node pair
2. Generate route by finding the shortest path between origin and destination nodes
3. Check for minimum and maximum route length constraints. If route satisfies constraints then route is adopted as the candidate route.
4. Generate alternate routes by clamping every link on the shortest links generated in step 2 successively and by finding the shortest path between origin and destination and then releasing the link.

5. Check for each alternate route, whether it satisfies constraints or not such as a) existence of routes b) duplication of routes c) significant overlap with shortest route d) maximum route length e) maximum route detour. If the route is satisfactory then the alternate route is accepted as a candidate route.
6. Rank all the routes

Thus a GA manipulates the coded representation of the problem. The individual routes are considered as variables. The value of the variable can be the performance index of the individual route, like cumulative demand satisfied or the passenger kilometer of the individual route.

2.2.3 Two Phase Genetic Algorithm

The selection of an optimal public transport route structure for a network is a combinatorial type optimization problem. (Dhingra, et al.2000) presents a good example of the applicability of a two-phase genetic algorithm to solve this problem. The design process is done in two phases. In the first phase major corridors of passenger movement in the network are studied and identified then, optimal routes are developed based on some user-defined constraints. The main objective is to minimize the in vehicle transit time and transfer time for the whole network. In the second phase optimal schedules are found for the routes developed from the above model the main objective considered is the minimization of an overall cost which is a combination of in-vehicle travel time, waiting time, transfer time and the operator costs.

Corridor identification is an important process for route design. Most of the studies start by considering a skeleton of nodes for a route, then more nodes are added to the skeleton based on some predetermined objective and sequentially more routes are generated until a significant demand is satisfied. The main starting point for model would be the identification of major corridors in the network for passenger flow movement. The main considerations include identification of major trip generators based on user specified guidelines. Then it starts developing routes between those nodes, which fall within the route length restrictions, shortest length considerations and also have sufficient route flow values. These nodes are the basis for the development of new routes. Finally, for each selected node pair K shortest paths are developed. Genetic Algorithms are used to select one of the K^{th} routes on a random basis for each of the node pairs, for the network, but within the maximum allowed variation of the K value.

2.2.4 Artificial Intelligence for Transit Route Planning and Design

(Baaj and Mahamassani, 1991) determined a configuration consisting of a set of routes and associated frequencies using an AI based approach. The objective function is a minimization of the total cost measure, a combination of user costs and operator costs. The former is often captured by total travel time incurred by users in the network, while a proxy for operator costs is the total number of buses required for the configuration.

Feasibility constraints include: 1) minimum operating frequencies on all routes 2) maximum load factor on all bus routes and 3) a maximum allowable bus fleet size. By changing the weights to reflect the relative importance of the two cost components one can achieve a trade-off between the two different sets of routes.

The major components of an AI based approach include a route generation algorithm that generates a set of routes corresponding to the set of trade-offs; an analysis procedure that computes a whole array of network level, route level and node level descriptors as well as frequency of buses necessary on all routes to maintain their load factors under a prespecified maximum value; and a route improvement algorithm that considers each set of routes and generates an improved set of routes based on an analysis package.

Route generation starts by sorting a demand matrix in decreasing order of the number of trips and selects M node pairs of the sorted demand matrix. The idea is to connect these high demand node pairs along either by the shortest path or the next shortest path. Thus M highest demand node pairs lead to M skeleton routes. Overlapping routes are avoided.

The next test finds out if these skeleton routes satisfy demand directly without transfers and with transfers. After route generation the next step is to analyze the routes and find out a suitable path choice on the basis of an assignment procedure. Route improvement follows the route analysis procedure.

2.2.5 Bus Transit Service for Maximum Profit and Social Welfare

(Patnaik et al, 1998) presents a framework for finding optimal transit service coverage in an urban corridor. The service variables considered include route length, route spacing and headway (or its inverse frequency). The criterion for optimality is either maximizing profit or maximizing social welfare. However, most transit services do not recover operating costs from the fare box and need to be subsidized from additional external revenue sources. The optimal design variables that maximize operator profit and social welfare are derived from a rectangular corridor with elastic demand, uniformly distributed passenger density and many to one-travel patterns.

2.2.6 Planning and Design Model for Route Networks

A heuristic model is presented for the design of bus transit networks with coordinated operations by (Baaj et al.1990). This model uses a transit center concept and incorporates a trip assignment model developed for timed transfer systems. In addition this model determines the approximate vehicle size for each bus route, and incorporates demand – responsive capabilities to meet demand that cannot be effectively serviced by the route. The model is composed of four major procedures: (1) A route generation procedure (RGP) which constructs transit network around transit center concept; (2) a network analysis procedure which incorporates a trip assignment model and a frequency setting and vehicle sizing procedure; (3) a transit center selection procedure, which identifies suitable transit centers for route coordination; and (4) a network improvement procedure, which focuses on the set of routes generated by the RGP. The work tries to unify the planning activities at the network design level and setting frequencies level. Starting with predefined network and frequencies this proposed scheme obtains new bus networks with better performance and more suitable line frequencies. The bus network is described by a genetic representation. The algorithm loads the initial population. At each generation the algorithm defines a fitness function value (ff value) for each network initially assigned. Each member of the population is evaluated by computing a number of performance indicators obtained by analysis of assignment of O/D demand associated with the considered networks. Thus, ff values are computed by means of a multicriteria analysis executed on performance indicators as found. A heuristic, which allows achieving the best possible network that satisfies, both demand and offer for transport is designed.

2.3 Types of Road Networks

The most important factor in the quality and adequacy of service provided by a fixed-route bus system is the design of the network of routes (Gray and Hoel, 1992). This section describes the major types of bus networks; in actual practice, most urban bus systems employ some attributes of several network types.

2.3.1 Radial Network

In a radial network buses are fanned out in a radial pattern from the central business district (CBD) into the suburbs. This is shown in Figure 1.

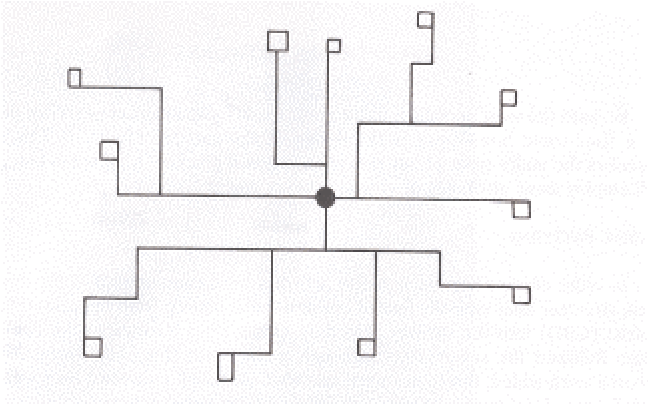

Figure 1: Radial Bus Network

As new suburbs were added, the routes were extended. Although cross-town lines were often added, some local transit systems still follow a basic radial pattern. Radial patterns continue to serve work trips to downtown effectively as long as there is a reasonable concentration of employment there. But if downtown commercial activities, such as shopping, are relocated to the suburbs, this type of transit network may not have convenient access to the new locations. Instead of being able, for example, to go shopping downtown from every neighborhood, access to a new shopping center by transit is possible only if you happen to live in the same transit corridor. Many urban activities have become decentralized, including employment, medical facilities, college campuses, and entertainment. These profound changes in land usage in the typical cities have made it difficult to incorporate a radial bus network to provide adequate service for most urban trips.

2.3.2 Grid Network

Figure 2 shows a grid network, which feature relatively straight, parallel routes spaced at regular intervals and crossed by a second group of routes with similar characteristics. They generally require a minimum of geographic or topographic barriers and an evenly spaced network of arterial streets suitable for bus operations. A major advantage of a grid-type system for an area that has widely scattered activity centers is that riders can get from almost anyplace to almost anyplace else with one transfer, without having to travel back through a central point such as the CBD. Another advantage is the relative simplicity of the system.

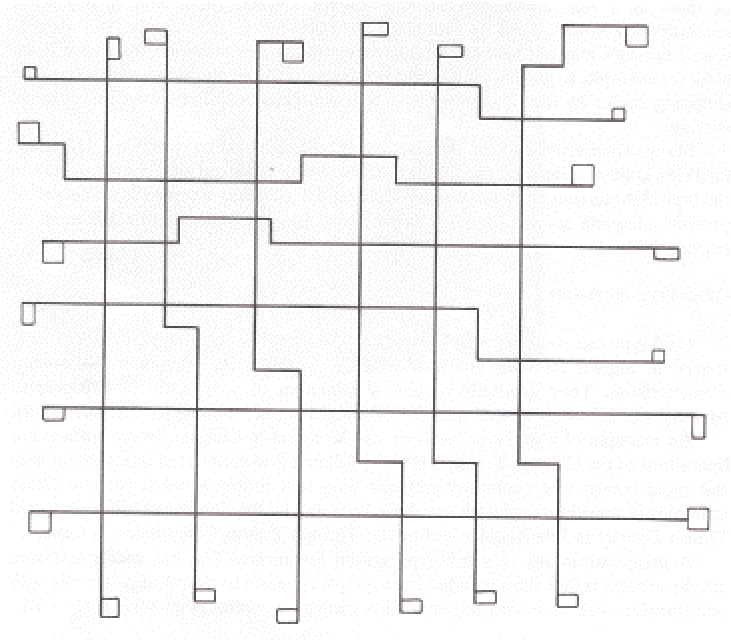

Figure 2: Grid-Type Bus Route Networks

2.4 Route Design Standards

The criteria for route design standards are used in determining or establishing the pathway for the bus route design. Of the criteria related to network design consideration the following five are generally considered the basics of route design standards: population density, employment density, spacing between other routes, limits on the number of branches and geographical coverage through the local tax base. According to the level of importance the criterion can be classified as primary or secondary.

2.4.1 Primary Criteria

The primary criteria are listed as under:

1. Population Density. It represents the number of people residing per mile and is the representation of the potential in terms of daily trips, at the point of the origin.

2. Employment Density. It represents the number of jobs per square mile. Typically, work trips account for well over one- half of a transit ridership.
3. Route Coverage. It refers to spacing distance between adjoining routings. The route coverage criterion guides spacing between bus services, geographically distributing them within the service area. This is done to maximize patron accessibility to transit services within the resources available to the transit agency.
4. Limitation on the number of branches. It provides for regularity in the pattern of main bus routing, whatever the directness of the main routing be. Branching involves selected trips leaving the mainline of the route; the deviation is viewed with regard to routing of the main bus route, not the streets over which the main bus operates.
5. Equal coverage throughout the Local Tax Base Area. Bus routes operate in jurisdictions or other political subdivisions based on local tax based contributions.

2.4.2 Secondary Criteria

Some of the secondary criteria are:

1. Reduction of Duplication. This criterion refers to a situation wherein two or more distinct routings that serve same passenger markets appear within close proximity to each other. Reduction is designed to control the duplication of bus routings to ensure that transit services are distributed geographically within a service area.
2. Network Connectivity. This criterion refers to the physical relationship of a new routing to the existing route system. When a new route is being introduced, its relationship to the entire system is considered. For example, this may mean

designing a route that connects two others, thereby creating a through route, and thus providing one seat route for customers on what would otherwise be three routes.

3. Service Equity. It is the distribution of the service on the basis of the population-based criteria.
4. Route Directness. A mathematical assessment is used to measure the route's deviation from its linear path based on the additional travel time required, which lowers its productivity.
5. Service Proximity to Residences. The service is easily accessible to localities where people live.
6. Service to as many Non–Residential Trip Generators. Service that caters to shopping trips, official trips.

CHAPTER 3. BASIC CONCEPTS OF NETWORKS AND GRAPHS

3.1 Introduction

This section gives a review on networks and graphs. It also explains the definition of the accessibility factor (in other words attractiveness value or benefits of a segment). In addition, the shortest path algorithm and its application to model an ideal network are discussed. Most of the material presented in this chapter is adapted from (Evans et al, 1992).

3.2 Graphs

A graph consists of two parts: the nodes and the lines joining these nodes. The nodes of a graph are its vertices and the lines joining these nodes are its edges. A graph G represented by a set X, whose elements are called vertices, and a set E, whose elements are called edges. Figure 3 contains 4 nodes and 5 edges.

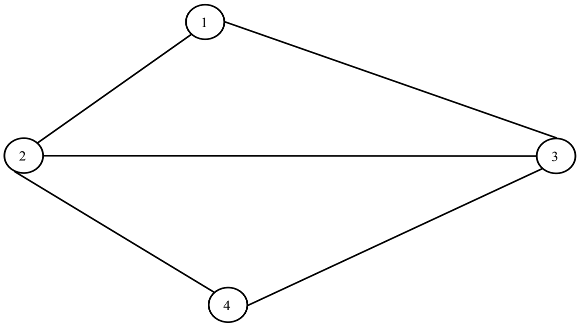

Figure 3: A Graph

The set of vertices X in the figure 3 consists of vertices $\{1,2,3,4\}$, while the set of edges consists of $\{(2,4)(4,3)(2,3)(2,1)(3,1)\}$.

Whenever set E consists of unordered pairs of vertices, we have an undirected graph. In an undirected graph, an edge (x, y) and an edge (y, x) are indistinguishable. In many practical situations, such as one-way streets, drawing arrows on the lines between the vertices specify the direction of the edge. Directed edges are called arcs, and the graph is called a directed graph. An example is shown in the Figure 4.

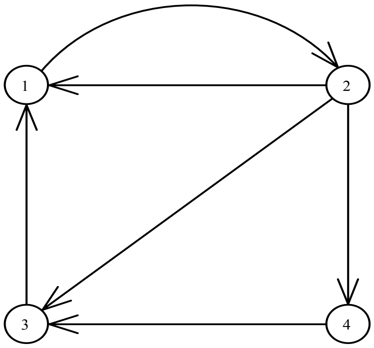

Figure 4: A Directed Graph

An edge that has both its endpoints as it same vertex is called a loop as shown in Figure 5.

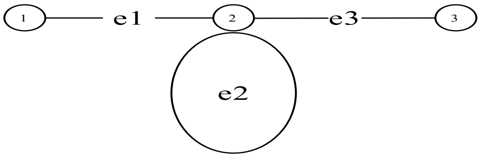

Figure 5: Graph with a Loop

A graph in which every pair of vertices is connected by an edge is called a complete graph. The degree of a vertex is the number of vertices incident on it. A vertex and edge are incident on each other if the vertex is an endpoint of the edge. Consider a sequence $x_1, x_2, ...x_n, x_{n+1}$ of vertices. A path is any sequence of these edges $e_1, e_2, ...e_n, e_{n+1}$ such that the endpoints of the edge e_i are x_i and x_{i+1} for $i = 1, 2, ..., n$. Vertex x_1 is called initial vertex of the path; vertex x_{n+1} is called the terminal vertex of the path. The length of the path equals the number of edges in the path. In Figure 6, a sequence of the edges e1, e2, e3 form a path of length 3 from vertex 1 to 4.

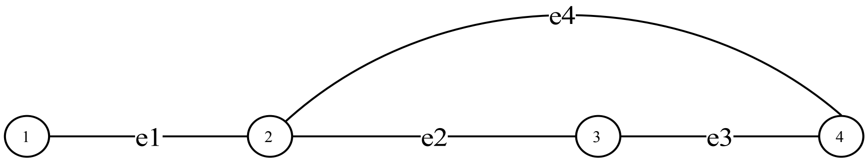

Figure 6: A Path in a Graph

The concept of path in a directed graph is the same as in an undirected graph. In a directed path all arcs are pointed in the same direction from the initial vertex to the terminal vertex as depicted by Figure 7.

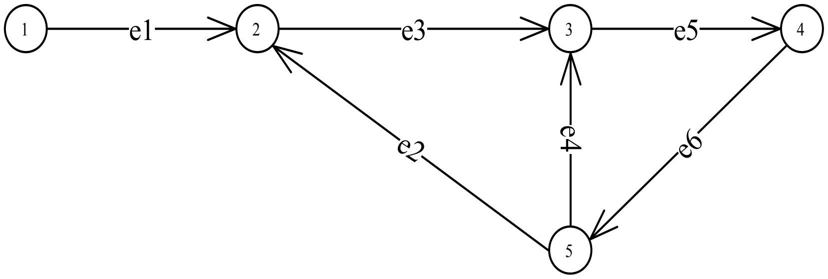

Figure 7: Directed Graph

A cycle is a path whose initial vertex and final vertex are identical. A directed graph is considered acyclic if it has no directed cycles. A path or cycle is called simple if no vertex is incident to more than two of its edges.

3.3 Data Structures for Networks and Graphs

If G=(X, E) is an undirected graph with m vertices and n edges then there exists several ways that represent G for computer processing. These include vertex node adjacency matrix and the node arc incidence matrix.

3.3.1 Vertex Node Adjacency Matrix

Matrix representation provides a convenient way to describe a graph without listing the vertices or edges or drawing pictures. The vertex adjacency matrix can be defined as follows: Let A be a $M*M$ matrix in which $a_{ij}=1$ if vertices i and j are adjacent, that is connected by an edge, and 0 otherwise. The matrix so formed is symmetric for an undirected graph and the number of ones in each row gives the number of edges incident to that vertex. For a directed graph we define $a_{ij}=1$ if there exists an arc (i,j) from node i to node j. the node adjacency matrix shows the connection between nodes. Table 2 represents a node-node adjacency matrix for Figure 8. It depicts if there exists a connection between nodes. The connection between node 2 and node 1 has value of 1 in the adjacency matrix. This means node 1 and node 2 are connected by a segment a.

Table 2: Node Adjacency Matrix

$$\begin{array}{c|cccc} i \backslash j & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 & 1 \\ 3 & 1 & 1 & 0 & 1 \\ 4 & 0 & 1 & 1 & 0 \end{array}$$

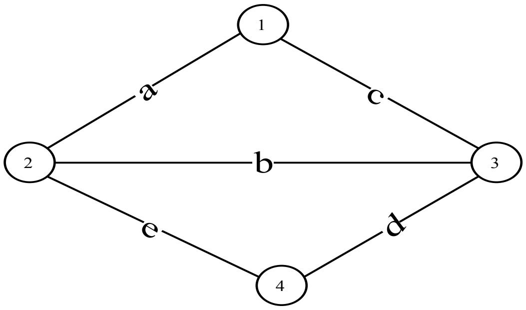

Figure 8: Undirected Graph

3.4 Shortest Path Problems

In a graph G where each arc (x, y) has associated with it a number $a(x, y)$ that represents the length of the arc. The length of a path is defined as the sum of the lengths of the individual paths comprising the path. For any two vertices, s and t in a graph there exist several paths from s to t. The shortest path problem involves finding a path from s to t that has the smallest possible length. Shortest path problems are commonly encountered on transportation applications.

3.4.1 Djikstra's Algorithm

Djikstra's algorithm provides the basis for solving shortest path problems. The main idea underlying the shortest path algorithm is quite simple. If we know k vertices are closest in length to vertex s in the graph and also the shortest path of s to each of these vertices then label the vertex s and their k vertices with their shortest distance from s. The vertex t is the sink node. Then $(k+1)^{st}$ closest vertex to x is found as follows. For each labeled vertex y, construct k distinct paths from s to y by joining shortest path from s to x with arc (x, y) for all labeled vertices x. The shortest path is found by incorporating the following steps: An example of a network is shown in Figure 9 for a better understanding of the algorithm. A typical network problem has been solved using the algorithm and is attached in the Appendix I.

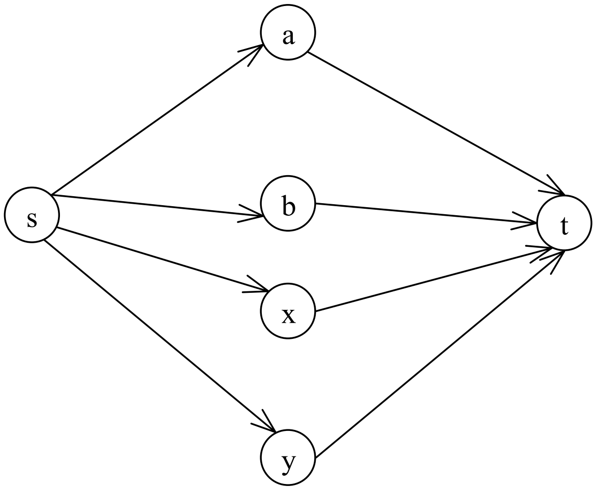

Figure 9: Illustration of Djikstra's Network

Steps in the Algorithm

Step 1**.** Initially all arcs and vertices are unlabelled. Assign a number $d(x)$ to each vertex x to denote the tentative length of the shortest path from s to x that uses only labeled vertices as intermediate vertices. Initially, set $d(s)=0$ and $d(x)=\infty$ for all $x \neq s$. Let y denote the last vertex that was labeled. Label vertex s and let $y=s$

Step 2**.** For each unlabelled vertex x, redefine $d(x)$ as follows:

$$d(x)=\min\{d(x), d(y)+a(y,x)\} \qquad (1)$$

The forward nodes from y are scanned, as these are the only nodes that are affected. If $d(x)=\infty$ for all unlabelled vertices x, then stop the process, as no path exists from s to any unlabelled vertex. Otherwise, label the unlabelled vertex x with the smallest value of $d(x)$. Also label the arc directed into the vertex x from a labeled vertex that determined the value of $d(x)$ in the above minimization. Let $y=x$.

Step 3**.** If vertex t, the end node is labeled the stop, since a shortest path from s to t has been discovered. This path consists of the unique path of labeled arcs from s to t. If vertex t has not been labeled, repeat step 3.

The algorithm labels a vertex (except vertex x) and also labels an arc directed to the vertex. Each vertex has at most one labeled arc directed into it and the labeled arcs cannot contain a cycle since no arc is labeled if both its endpoints have a labeled arc incident on it. It can be thus concluded that the labeled arcs form an arborescence rooted at s. This arborescence is called shortest path arborescence. The unique path contained in from s to any other vertex is the shortest path from s to x. If the shortest path from s to x in shortest path arborescence passes through vertex y, it follows that the portion of this path from y to x is the shortest path from y to x. The labeled arcs at all times form arborescence. The algorithm can be regarded as the growing of the algorithm rooted at vertex s. Once vertex t is reached the growing process can be terminated.

To determine the shortest path from vertex s to every other vertex in the graph, the growing process could be continued till all the vertices are included in the shortest path arborescence in which case the arborescence would become spanning tree arborescence.

In that case step 3 changes to the step 4 that follows.

Step 4**:** If all vertices have been labeled, stop because the unique path of the labeled arcs from s to x for all vertices of x. Otherwise, return to step 2. This algorithm evaluates the shortest paths between nodes. The shortest path algorithm is used in this thesis to determine the overall maximum possible benefit of serving all demands. This maximum possible benefit is obtained by a minimum path network between each node pair and summing the benefits over all the paths obtained. This is equivalent to assuming that there was a direct shortest path route between every two destinations in the network.

3.5 Accessibility

According to the Oxford English Dictionary access is defined as “the habit of getting near or into contact with”. Accessibility is a measure of the ease of access. Access is between entities and in our case it is between zones represented by nodes in a grid shaped network (Harris, 2001). The links of the network denote the impedance to access. Impedance is the hindrance to travel in the form of distance, travel time, waiting time etc. The cost of the link is proportional to the distance between zones. Separation of the zones in space is the opposite to the ease of access. The most straightforward description of accessibility is the state of connectivity. A location is assumed to be accessible if it is connected to other locations via a link to a road, railroad, sea or air network. The extent of accessibility is calculated as the number of different modes and links to which a specific location has access. Accessibility indicators are employed to describe and summarize the characteristics of the physical structure (e.g., accessibility to certain links, the network or specific modes or the transportation system as a whole). These indicators reveal the level of service of the network from the provider's perspective. The major theoretical approaches for the measurement of accessibility indicators are the travel-cost based approach and gravity approach.

Travel cost approach is the first class of accessibility measure and embodies those measuring the ease with which any land-use activity can be reached from a location using a particular transportation system. This indicator has been utilized to indicate performance of the transportation infrastructure. The common aspect for this class of accessibility indicator is determined by their configuration, where the indicator is a proxy of the transport cost (network or Euclidean distance, travel time, or travel cost). A functional form for this class of measure is presented by the equation

$A_i = \sum_{j \in L} \frac{1}{f(c_{ij})}$ Where A_i is the measure of accessibility at location i,

L is the set of all locations,

$f(c_{ij})$ is the deterrence function and

c_{ij} is a variable that represents travel cost between nodes.

The other class of accessibility measure is based on the gravity or opportunities approach. The indicators in this case are based on spatial opportunities available to travelers and also consider the behavioral aspects of travel .A simple model is hereby described below.

$A_i = \sum_{j\in L} \frac{W_j}{f(c_{ij},\beta)}$ where W_j represents the mass of opportunities available to consumers, regardless if they are chosen or not

$f(c_{ij},\beta)$ is the deterrence function,

c_{ij} is a variable that represents travel cost between nodes i and j

β is the travel-cost coefficient usually estimated from a destination choice model.

The deterrence function can be linear or exponential in travel time. The travel cost model and the gravity model are generally examined based on travel time (in other words distance). Accessibility models based on deterrence functions are given below.

Travel cost model based on linear travel time

$$a_1 = \frac{1}{t_{ii}} + \sum_{j\in L} \frac{1}{t_{ij}}, \quad i \neq j$$

Travel cost model based on exponential time

$$a_1 = \frac{1}{e^{bTii}} + \sum_{j\in L} \frac{1}{e^{bTij}}, \quad i \neq j$$

where t_{ii} is the internal travel time at i, a_1 is the accessibility

t_{ij} is the travel time between locations

Accessibility measure based on the gravity approach is shown below

$$b_1 = \frac{P_i}{t_{ii}} + \sum_{j\in L} \frac{P_j}{t_{ij}}, \quad i \neq j$$

$$b_1 = \frac{P_i}{e^{bTii}} + \sum_{j \in L} \frac{P_j}{e^{bTij}}, \qquad i \neq j$$

where b_1 is the accessibility measure and P is a measure of population.

Consider a sixteen-node problem as represented by a grid network of nodes in Figure 10. These zones are connected to each other by links, which are road segments. Every node has a certain set of activities like population, employment, annual income etc. Depending on these activities one can find out the attractiveness to move from one node to other. This is based on a function written on the lines of the gravity approach and can have distance/or cost as a linear or exponential function. It is assumed that travel time in this case follows a linear pattern. One can find out attractiveness to go to all nodes. Thus the accessibility to move from one node to another is a function of the employment, population, annual income at the nodes and the distance between nodes. The accessibility of the route is its value and is obtained by dividing the attractiveness value of the segment by its impedance.

	1	2	3	4
A	●	●	●	●
B	●	●	●	●
C	●	●	●	●
D	●	●	●	●

Figure 10: Grid Network

The attractiveness to move from node $A1$ to node $B1$ is a function of the population at the two nodes, employment and annual income at the two nodes. The function is given as under;

P_1=*Population at node A1*

P_2= *Population at node A2*

Pi=*Population at node i*

$P_0 = \sum_i Pi = \text{Sum of the Populations at all the nodes } i, i = 1,2,....n$

E_1=*Employment at node A1*

E_2= *Employment at node A2*

Ei=*Employment at node i*

$E_0 = \sum_i Ei = \text{Sum of the Employment at all the nodes } i, i = 1,2,....n$

Then attractiveness value on the link between node A_1 and node A_2 is given by the formula Attractiveness $A_{12} = \{(\alpha\,(P_1 + P_2)/(P_0) + \beta\,(E_1 + E_2)/(E_0))\}$

Tij is actually the length of the link and is expressed as a generalized cost. As the length of the link increases the accessibility value reduces. Accessibility of a segment is the attractiveness value of the segment divided by the cost Tij of traversing the segment. This takes into consideration the basic tendency of humans to avoid long routes. An attractiveness matrix is shown in Table 3.

Table 3: Attractiveness Matrix for a Sample Network

	A_1	A_2	A_3	B_1	B_2	B_3
A_1	-	15	18	7	3	8
A_2	15	-	14	6	4	7
A_3	18	14	-	1	3	2
B_1	7	6	1	-	9	4
B_2	3	4	3	9	-	14
B_3	8	7	2	4	14	-

The attractiveness of A1 to A2 is same as A2 to A1. This applies to all segments of all the attractiveness on the links of the route. For example if there is a route A1-A2-A3. The accessibility of the route will be the accessibility to go from A1 to A2 added to the accessibility to go from A1 to A3 added to the accessibility to go from A2 to A3. The accessibility value is thus a sum of (15/(cost of A1A2)+18/(cost of A1A3)+14/(cost of A2A3)) = 15/2 + 18/5 + 14/3 =7.5 + 3.6+ 4.67 = 15.77. Thus Accessibility of path A1-A2-A3 is 15.77. The objective function is to maximize the value of a route by adding segments to the route in such a way that the resulting accessibility value is increased as per above addition method. The accessibility value of the route is calculated on the fly as the route is being formed. The neighbors of a node are found out from the incidence matrix for nodes. The incidence matrix for nodes is given in Table 4.

Table 4: Incidence Matrix for a Sample Network

i /j	A1	A2	A3	B1	B2	B3	C1	C2	C3
A1	0	1	0	1	0	0	0	0	0
A2	1	0	1	0	1	0	0	0	0
A3	0	1	0	0	0	1	0	0	0
B1	1	0	0	0	1	0	1	0	0
B2	0	1	0	1	0	1	0	1	0
B3	0	0	1	0	1	0	0	0	1
C1	0	0	0	1	0	0	0	1	0
C2	0	0	0	0	1	0	1	0	1
C3	0	0	0	0	0	1	0	1	0

Nodes A1 and A2 are connected by a link as the incidence matrix shows the values of A1A2 and A2A1 to be 1. Nodes B3 and A1 are not connected so the incidence matrix has a value of zero for the connection.

CHAPTER 4. PROBLEM MODELING

4.1 Introduction

This section provides the practical significance of the bus route network design problem followed by the modeling assumptions and the heuristic method developed.

4.2 Problem Significance

Bus route design is one of the most important elements of public transit system planning. In times of reduced funding from governments it becomes mandatory for transit agencies to operate routes of value. Greater the ridership means higher the value of the route. This also means improved benefit to cost ratio for the route. The main thrust area of route design is to make the route more accessible to the transit population. The critical element is to find a way to identify combinations of sequential links that comprise logical bus routes.

The total route length is constrained to a percentage of the longest path across the network. The Djikstra's algorithm is used to calculate all possible shortest paths between the existing node pairs. The paths are then ranked in a descending order based on length. The longest path is the one, which is ranked the highest.

In this research we have constrained the length to a percentage of the longest path formed by the applying Djikstra's algorithm to a test network. This is also called the threshold distance of the route. This has been done to restrict the route length within practical limits in order to make it a realistic route. The test network is shown in the Figure 11.

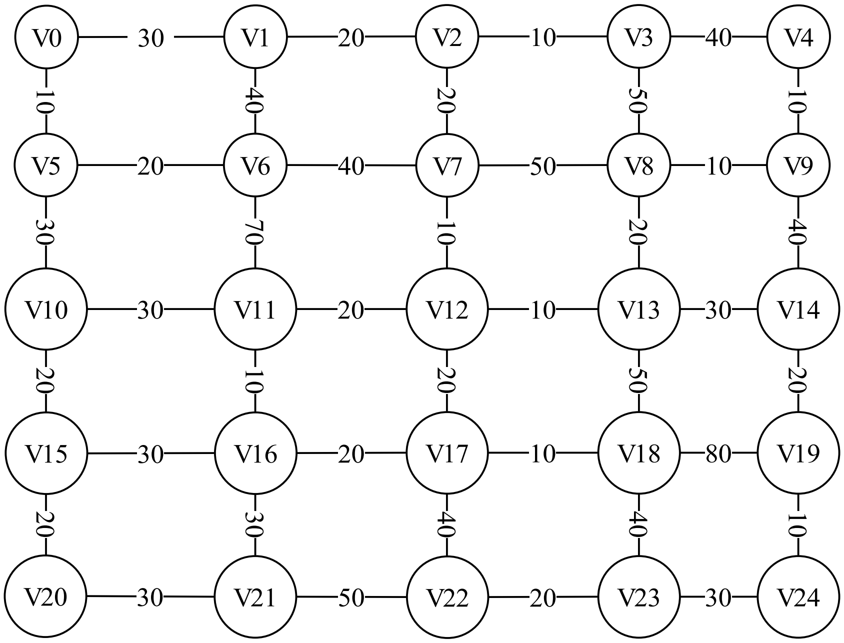

Figure 11: Test Network

The accessibility of connecting any two nodes is proportional to the activity levels at the nodes expressed as a combination of the population, employment of trip ends and inversely proportional to the travel distance/cost between them as defined in chapter 3. The term accessibility is used to identify the factor that we will be striving to optimize, i.e., routes connecting segments that produce the most benefits will determine the attractive routes and higher accessibility.

The distance to travel, cost of travel, waiting times etc., are the parameters that hinder travel. These parameters are known as impedance. The impedance in this case is the distance to travel. It is represented as a cost and this cost has a linear relationship to the route length. The cost matrix gives the cost on all the possible links of the network.

As a logical starting step, the attractiveness of connecting each pair of adjacent point is determined. Adjacent points are defined as those points where the shortest path has no intervening points on the network. This can be used to find a starting link for the first bus route.

This segment may not be an overall optimal as the best first segment and its best subsequent segment may not be optimal over the best two-segment section, nonetheless it is a logical starting point.

The challenge lies on finding the subsequent segments that are most attractive based on the accessibility value of the combinations that arise by searching from the extremities of the segment. The combinations cover nodes that are adjacent to the extremities of the selected segment. Accessibility of a segment is it attractiveness divided by the cost of using the segment in the route formation. In general, each segment can be expanded on either end to any adjacent node. That set of possibilities will define the set of possible second segments. Evaluating the subsequent segments is thus the critical challenge that has been addressed with the help of a logical strategy. The strategy is discussed using a case study for a nine-node example later in this chapter. The strategy uses a one level and a two level search method. The strategies are applied to a small nine-node grid network for a comprehensive understanding of the route formation.

The basic behavior of the one level search and the two level search techniques is explained by applying both the strategies on a simple network shown in Figure 12. The search routine evaluates each subsequent segment in the context of not only the value of the segment but also the value of each decision in terms of future opportunities of subsequent segments. For example, using Figure 12 assume segment N_5-N_6 is the starting segment, the options for subsequent segments in the route could be segments from either ends of the initial segment. That is: N_5-N_1, N_5-N_9, N_6-N_2, N_6-N_7, N_6-N_{10}.

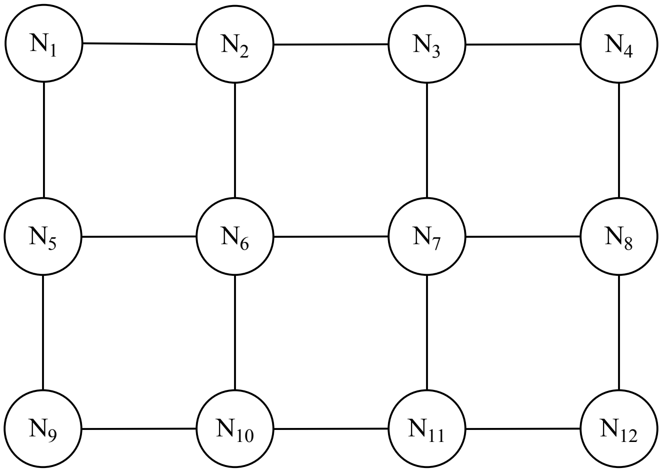

Figure 12: Example Network

N_5-N_1, N_5-N_9, N_6-N_2, N_6-N_7, N_6-N_{10} are all in the choice set. This is termed as One Level Search.

Each of those segments opens the possibility of subsequent segments. Thus N_5-N_1 opens up the possibility of N_1-N_2 as the next immediate segment. Similarly, N_5-N_9 opens up N_9-N_{10}, N_6-N_2 opens up N_2-N_1 and N_2-N_3, N_6-N_7 opens up N_7-N_3 and N_7-N_{11}, N_6-N_9 opens up N_{10}-N_9 and N_{10}-N_{11} as the next segments. Thus, N_9-N_{10}, N_2-N_1, N_2-N_3, N_7-N_3, N_7-N_{11}, N_{10}-N_9 and N_{10}-N_{11} form the second choice set. This is termed as Two Level Search.

Thus, determining the subsequent segment would involve a search of the all the above-mentioned options occurring in the first level set and the second level set to determine the path with the highest accessibility or with the highest contribution to benefits or has the highest probability of a positive contribution to benefits. The available paths for Figure 12 are shown in Table 5.

Table 5: Search Choice Set

One Level Search Choice Set	Two Level Search Choice Set
N_5-N_6-N_2, N_5-N_6-N_7, N_5-N_6-N_{10}, N_6-N_5-N_9, N_6-N_5-N_1	N_5-N_6-N_2-N_3, N_5-N_6-N_2-N_1, N_5-N_6-N_7-N_3, N_5-N_6-N_7-N_8, N_5-N_6-N_7-N_{11}, N_5-N_6-N_{10}-N_{11}, N_5-N_6-N_{10}-N_9, N_6-N_5-N_1-N_2, N_6-N_5-N_9-N_{10}

The benefits of adding segment n_6-n_2 to a starting link n_5-n_6 can be expressed as the benefits of N_5-N_6 + N_6-N_2 plus the benefits of N_5-N_6-N_2 plus the max benefits of (N_5-N_6-N_2-N_3) or (N_5-N_6-N_2-N_1). This is based on the definition of total accessibility of a route as defined in chapter 3. The accessibility value of N_5-N_6-N_2 will be the sum of accessibility values of each of the combinations i.e., N_5-N_6 plus N_6-N_2 plus N_5- N_6- N_2. The value of each of the segments is computed by dividing the attractiveness of the segments by the respective cost incurred to traverse the segment. This logic could be extended to subsequent segments by examining the benefits to additional sequential segments.

The ultimate goal is to find a system with the highest total accessibility value. Each subsequent segment is contrasted with the next highest individual segment accessibility value. For example, if the benefit of extending a route from N_5-N_6 to N_2 is lower than starting a new route with the initial segment as N_{11}-N_{12}, then a new route will be started and two routes will be formed at the same time by adding subsequent links to the route of N_5-N_6-N_2 and route N_{11}-N_{12}. If one level search is used for route formation and the cumulative accessibility value of route N_5-N_6-N_2 is better than those available in the one level search choice set, then N_6-N_2 is the next added segment to the route N_5-N_6. If two level search were to be applied and N_5-N_6-N_2-N_3 were to be chosen as the best route then in that case the actual route is taken as N_5-N_6-N_2 by truncating route N_5-N_6-N_2-N_3 and the Two level Search procedure is applied recursively to this route on either end i.e., on either N_2 or N_5. A new route is started from N_{11}-N_{12} if this segment is more attractive than the incremental value of adding n_6-n_2. That allows, two routes to be formed at the same time. They are 1) N_{11}-N_{12} 2) N_5-N_6-N_2. Each of these routes is constrained by a percentage of the path, which has ranked first by applying Djikstra's algorithm to the test network.

4.3 Problem Assumptions

The following assumptions were taken for modeling purpose.

1. The grid network used for testing purposes is an undirected graph without any cycles.
2. The starting node for the route and the ending node are decided from the attractiveness matrix formed or can be decided based on attraction centers or trip production centers, which will always lie on the route. The node-node adjacency matrix is known.
3. The grid network has no diagonal links. This means that if there are 4 nodes max at the ends of a square one cannot traverse between nodes situated across the diagonals.

4. Two sets of constraints have been included. The route length is constrained to a length as discussed under the problem significance section. No cyclic routes are allowed. A route is said to be cyclic if it reaches the node it originated more than once in the route formation. For example route N_1-N_3-N_5-N_6-N_1 is cyclic, as it started from node n_1 and ended at the same node after covering nodes N_3, N_5, N_6 along the way. A flowchart to represent the heuristic is shown by Figure 13.
5. A route connecting two zones is used by travelers in some proportion to the activity levels in each of the zones and the cost or impedance of traveling between them.

4.4 Inputs to the Problem

The population, employment and annual income at all the nodes are known. The attractiveness value between nodes is calculated using the following formula as under:

$$\text{Attractiveness } A_{12} = \{(\alpha\ (P_1+P_2)/(P_0)+\beta\ (E_1+E_2)/(E_0)\}$$

and an attractiveness matrix is derived. Employment values are generally one fourth of the population numbers. The Adjacency matrix provides information on the nodes that are connected to each other. The cost matrix gives the cost of traveling between the nodes. The cost is directly proportional to the distance as indicated earlier. A search matrix is initialized at the start of the procedure. It is set to a null matrix. Each time a segment is added to a route, the search matrix takes the value of 1 for that segment. This helps to keep a track of segments already added to the route and removes the duplication of segments in two different routes. The entire search process is stopped when the search matrix is an exact replicate of the adjacency matrix. Figure 13 shows the steps of the heuristic algorithm and the details of each step are provided next.

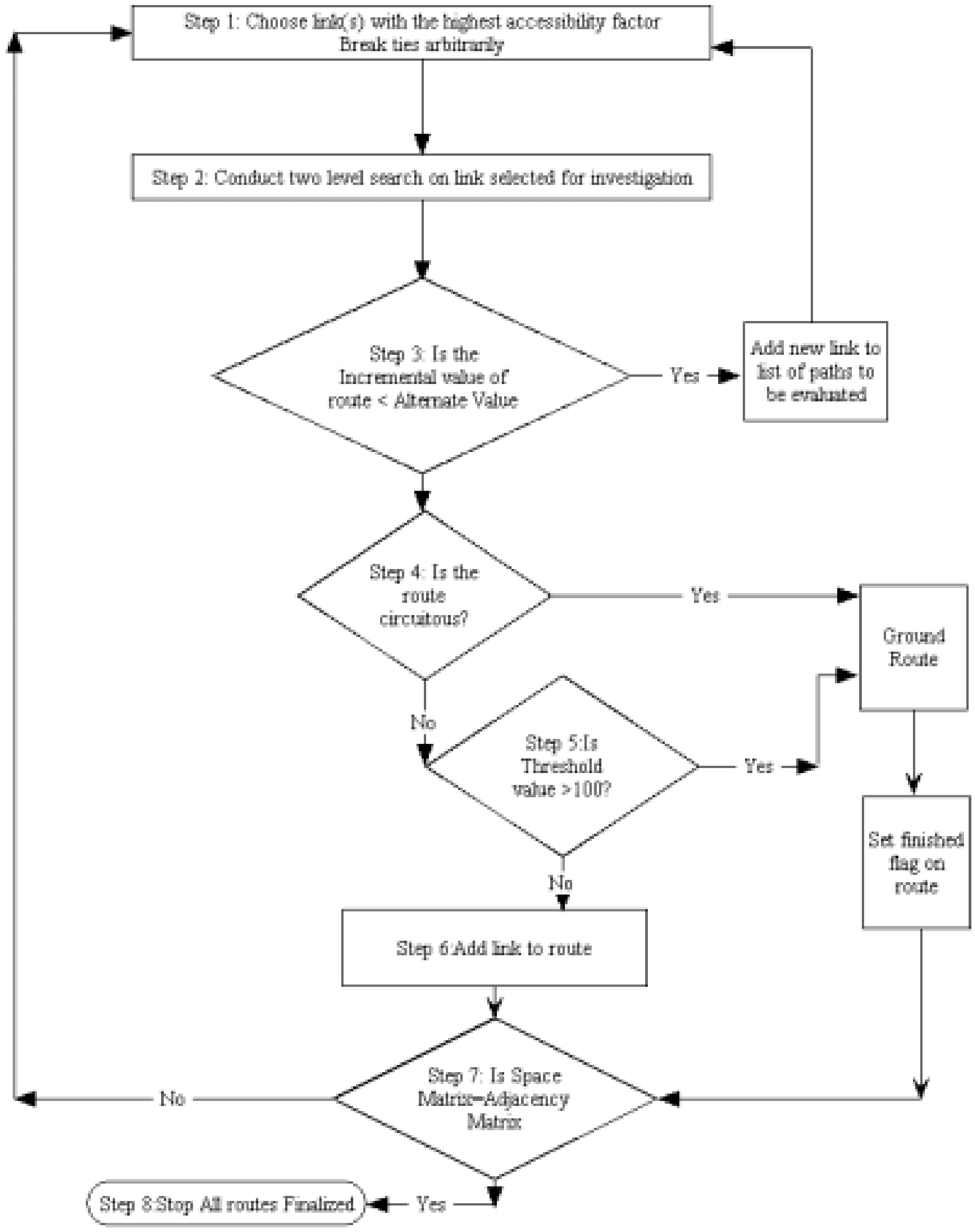

Figure 13: Flowchart of the Forward Searching Heuristic

The heuristic comprises the following steps:

Step 1. Choose link with highest benefits from the accessibility matrix. The adjacency value for this candidate path has to be 1 in the node- node adjacency matrix. This link becomes the starting link and acts as a seed to the route generation procedure. Depending on the number of links having the same attractiveness value in the matrix, a set of candidate paths is formed. The selected candidate paths are now ready for investigation. Each of these candidate paths is evaluated.

Step 2. This step incorporates a search procedure to find the subsequent links to the segments already existing in the list of candidate paths in step 1. The segments can be extended on either side to any adjacent node. The starting link expands to the side, which has a more attractive segment. The accessibility values of the route formed by adding the next segment is calculated based on the explanation in chapter 3.

Step 3. The incremental value of adding the next best segment to the route is compared with the idea of starting a new route from any other link in search space. This link is not yet part of any route. If the value of the adding a segment to the route is greater than starting a new route, from any other link which is not yet part of a route, then it that case the route is checked for threshold length and circuitous routes in step 4 and step 5 respectively. If the starting of a new route is more attractive than just adding a segment to the route, then in that case the link is added to the set of candidate paths to be evaluated.

Step 4. If the route is circuitous then the routes are ended and a finished flag is set for the route. Route N_1-N_6-N_5-N_0-N_1 is a circuitous route as the route formation encounters node 1 twice. The adjacency matrix is checked with the search space matrix. The search space matrix is a null matrix whose elements are 0 initially but changes to 1 if a link becomes part of a route. If the condition of equality of the search space matrix and adjacency matrix is satisfied, display all paths else go to step 1.

Step 5. If the route length crosses a set threshold value then the route is stopped and a finished flag is set on the route. The adjacency matrix is checked with the search space matrix. The search space matrix is a null matrix whose elements are 0 initially but changes to 1 if a link becomes part of a route. If the condition is satisfied display all paths else go to step 1.

Step 6**.** In case both the threshold and circuitous conditions are not met, the route is extended further using the two level search method as in step 2 till it encounters a condition where the search matrix is equal to adjacency matrix.

Step 7**.** Once all the candidate paths are evaluated and the search space matrix equals the adjacency matrix, then in that case the route formation is stopped as all the routes are finalized.

Step 8**.** Route formation is stopped.

The overall maximum possible benefit of serving all demands is calculated by evaluating the minimum path network between each node pair and summing the Accessibility for the path obtained This minimum path network has been calculated using the Djikstra's network algorithm. The actual route network is found by using the forward searching heuristic. Thus, any actual route network is sub optimal to this optimal network but this optimal network is unrealistic in that it would have unrealistic levels of service (costs). This research measures the quality of the designed network to the ideal network by making a comparison of the benefits to costs ratio realized by both networks.

4.5 A Step-By-Step Example of the Heuristic Procedure

To better illustrate the hueristic a nine-node network is shown in Figure 14. The Attractiveness Matrix is given in Table 6, the Adjacency Matrix in Table 7; the Cost Matrix is in Table 8 and Search Space Matrix in Table 9 respectively. The Attractiveness matrix has been generated using the attractiveness equation discussed in chapter 3. The nine values of population have been randomly generated and employment values are 0.25 times the population at a particular node.

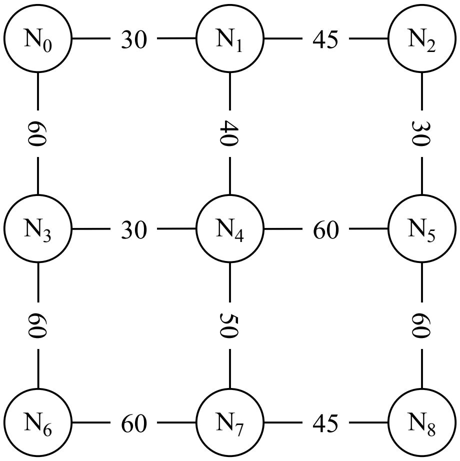

Figure 14: Nine-Node Network

Table 6: Attractiveness Matrix for a Nine-Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
N_0	0	18293	22056	20952	23567	26009	19464	24853	14194
N_1	18293	0	20523	19419	22035	24477	17931	23321	12662
N_2	22056	20523	0	23183	25798	28240	21695	27084	16425
N_3	20952	19419	23183	0	24694	27136	20591	25980	15321
N_4	23567	22035	25798	24694	0	29751	23206	28595	17936
N_5	26009	24477	28240	27136	29751	0	25648	31037	20378
N_6	19464	17931	21695	20591	23206	25648	0	24492	13833
N_7	24853	23321	27084	25980	28595	31037	24492	0	19222
N_8	14194	12662	16425	15321	17936	20378	13833	19222	0

Table 7: Adjacency Matrix for a Nine-Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
N_0	0	1	0	1	0	0	0	0	0
N_1	1	0	1	0	1	0	0	0	0
N_2	0	1	0	0	0	1	0	0	0
N_3	1	0	0	0	1	0	1	0	0
N_4	0	1	0	1	0	1	0	1	0
N_5	0	0	1	0	1	0	0	0	1
N_6	0	0	0	1	0	0	0	1	0
N_7	0	0	0	0	1	0	1	0	1
N_8	0	0	0	0	0	1	0	1	0

Table 8: Cost Matrix for a Nine-Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
N_0	0	30	0	60	0	0	0	0	0
N_1	30	0	45	0	40	0	0	0	0
N_2	0	45	0	0	0	30	0	0	0
N_3	60	0	0	0	30	0	60	0	0
N_4	0	40	0	30	0	60	0	50	0
N_5	0	0	30	0	60	0	0	0	60
N_6	0	0	0	60	0	0	0	60	0
N_7	0	0	0	0	50	0	60	0	45
N_8	0	0	0	0	0	60	0	45	0

Table 9: Search Space Matrix for a Nine-Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
N_0	0	0	0	0	0	0	0	0	0
N_1	0	0	0	0	0	0	0	0	0
N_2	0	0	0	0	0	0	0	0	0
N_3	0	0	0	0	0	0	0	0	0
N_4	0	0	0	0	0	0	0	0	0
N_5	0	0	0	0	0	0	0	0	0
N_6	0	0	0	0	0	0	0	0	0
N_7	0	0	0	0	0	0	0	0	0
N_8	0	0	0	0	0	0	0	0	0

Step 1. Choose the link with highest attractiveness value (or benefits) from the accessibility matrix. The adjacency value for this link has to be 1 in the node-node adjacency matrix. This link becomes the starting link. For the example considered here, the starting link from the Accessibility matrix is link N_4-N_5 as it has maximum attractiveness of 29751. Also the adjacency matrix for this link gives the value 1. The subsequent segment for this link is decided by the one level search method.

Step 2. This step incorporates a search procedure to find the subsequent links to the segment already selected. This segment can be extended on either side to any adjacent node. For link N_4-N_5 the adjacency from search space is worked out using the adjacency matrix and the network grid. Adjacency from search space for node N_4 is N_1, N_3 and N_7. Similarly adjacency from search space for node N_5 is N_2 and N_8. This is the first level of adjacency. Thus there are subsequent segments possible from nodes N_4 and N_5. The matrix in Table 10 gives the adjacency for nodes and the various paths formed with their accessibility values.

Table 10: Matrix for One Level Search

Adjacency of Node	Adjacent Nodes	Paths Formed	Accessibility Value
4	1	1,4,5	1289
	3	3,4,5	1619
	7	7,4,5	1348
5	2	4,5,2	1722
	8	4,5,8	983

All the paths so obtained are ranked according to their value of accessibility and the path with the highest value helps to decide on the next subsequent link. In the above example N_4-N_5-N_2 has the highest attractiveness value of 1728. This accessibility value for the path N_4-N_5-N_2 is obtained by the summation of the accessibility value of the individual segments N_4-N_5 and N_2-N_5 and N_4-N_2, which have a cost of 60,30,90 respectively. The Accessibility values of these segments are obtained by dividing the attractiveness by the cost. The accessibility values for each of these segments are 495,941 and 286 respectively. The path N_4-N_5 is thus continued with the addition of node 2 in the path. The path now is N_4-N_5-N_2.

Step 3. The next step is to find out the incremental value of adding link N_1-N_2 to the path. This value is the difference between the accessibility of path N_4-N_5 and path N_4-N_5-N_2. The incremental value is 1722-495=1227. This value is compared to all the unused segments in space to see if they are better. But by comparing the values it was found that no such segment existed in the search space. Once a node is found to be part of a route its value in the search space matrix is made 1 so that it does not become part of other routes. In the search space the values of N_4-N_5 and N_5-N_2 are changed to 1 to indicate that these segments have been used up in the route formation. The route N_4-N_5-N_2 is then checked for circuituity and threshold distance in steps 4 and 5 respectively. This is shown by Figure 11.

Table 11: Search Space Matrix

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
N_0	0	0	0	0	0	0	0	0	0
N_1	0	0	0	0	0	0	0	0	0
N_2	0	0	0	0	0	1	0	0	0
N_3	0	0	0	0	0	0	0	0	0
N_4	0	0	0	0	0	1	0	0	0
N_5	0	0	1	0	1	0	0	0	0
N_6	0	0	0	0	0	0	0	0	0
N_7	0	0	0	0	0	0	0	0	0
N_8	0	0	0	0	0	0	0	0	0

The incremental value of adding segment N_5-N_2 is greater than starting a new route from any other segment in space. Hence the same route is extended in the next iteration by applying the one level search recursively. The length of the route N_4-N_5-N_2 is 90.

Step 4. Check the circuitous nature of the route. The route is not circuitous and all paths have not yet been formed hence the heuristic proceeds to step 1.

Step 5. The threshold distance is set to a certain level for every experiment In this case it is set as 250. This means a route is formed till the length of the route overcomes 250 for the first time. Route N_4-N_5-N_2 hasn't yet overcome the threshold distance and is neither circuitous. Hence the heuristic moves to step 6.

Step 6. If both conditions of threshold and circuitous routes are not met then the route is extended further by step 2 by using a one level search.

Step 7. Once all candidate paths are evaluated and the search space matrix equals the adjacency matrix the route formation is stopped

Step 8. Route formation is stopped as all the routes are obtained as shown in Table 12.

Table 12: List of Best Paths by One Level Search

Routes	Route Links	Accessibility Value	Cost
1	6,3,4,5,2,1,0	6545	255
2	1,4,7,6	2117	150
3	7,8,5	1061	105
4	3,0	349	60

After the first path is completed the next most attractive segment in space is singled out for the route formation. In the case of the considered test case, that segment is N_4-N_7. The segment N_4-N_7 is extended as N_1-N_4-N_7-N_6. The incremental value of adding N_7-N_6 to route N_1-N_4-N_7 is less than starting a new route from segment N_8-N_5. Hence while route N_1-N_4-N_7-N_6 is continued a new route is also started in N_8-N_5. The route N_1-N_4-N_7-N_6 becomes locked as all its adjacent segments are taken by some other route and the one level search cannot be applied to it. Hence it is also stopped. The only segment that remains is segment N_3-N_0 and it becomes the last route. The route formation process is continued till all the links of the test network are part of the routes.

The list of paths formed using the heuristic are given by Table 12. A nine-node network has 12 segments in all. All the 12 segments are featured in Table 12.

4.5.1 Two Level Search Applied to a Nine Node Network

The Two level search procedure is used to form routes for the same network as used by the one level search. All the matrices that include the attractiveness matrix, adjacency matrix and cost matrix remain the same. The two level search is explained with the help of the first iteration in the search process. The search looks two segments ahead before it decides to extend the current segment. As compared to the one level search the search process is the only aspect, which is different. For the network under consideration a two level search process will extend the most attractive segment N_4-N_5 after searching through the two level choice set as shown in Table 13.

Table 13: List of Best Paths by Two Level Search

Adjacency of Node	Adjacent Nodes	Paths Formed	Accessibility Value
4	1,0	0,1,4,5	2434
	1,2	2,1,4,5	2400
	0,3	0,3,4,5	2402
	6,3	6,3,4,5	2389
	8,7	8,7,4,5	2116
	6,7	6,7,4,5	2094
5	2,1	4,5,2,1	2667
	8,7	4,5,8,7	1878

From the choice set the path with the best accessibility value is N_4-N_5-N_2-N_1.This path is then truncated and the route becomes N_4-N_5-N_2. Thus the segment N_5-N_2 is included in the route after a segment ahead of it N_2-N_1 is also considered in the evaluation process. This search is an intensive with more combinations being evaluated, and it guides the process by looking beyond the first level. A list of paths formed by applying the two level search process is listed in Table 14.

Table 14: List of Best Paths by One Level Search

Routes	Route Links	Accessibility Value	Cost
1	7,6,3,4,5,2,1	6224	285
2	0,1,4,7,8	3326	165
3	8,5	339	60
4	3,0	349	60

The shortest paths from a node to all other nodes are obtained by the Djikstra's algorithm. The accessibility of this ideal system is then measured by dividing the sum of attractiveness of all the paths by the costs of the paths. The paths obtained by forward searching heuristic, both the one level and two level searches are then compared and contrasted with the ideal network in terms of accessibility to cost ratio.

CHAPTER 5. COMPUTATIONAL EXPERIMENTS AND RESULTS

In the previous chapter the proposed heuristic solution to solve the route network design problem were presented and explained in detail. In this chapter the computational experiments are performed to test the effectiveness of the heuristic algorithm and are compared to an ideal network. The complete experimental setup and analysis of the results obtained are also discussed.

5.1 Software Implementation

The forward searching heuristic and the Djikstra's shortest path algorithm are coded in java, an object oriented programming language for better efficiency and run time. The code is run on Windows platform using JDK1.3.1_01 version. The attractiveness matrix is obtained and written to a file by using a Matlab program. A batch file is created where the user provides the number of vertices in network, the data folder to be read, specifies one level search or two level searches, the threshold distance, the budget and the output file. This makes the program user-friendly. The data folder contains following matrixes

1. Attractiveness
2. Cost
3. Adjacency

The forward search heuristic has been tested on a nine node network, sixteen node network, twenty five node network, thirty six node network and a forty nine node network. The Djikstra's algorithmn computes the shortest path from one node to all other nodes with a run time of a few minutes for all the different kind of networks mentioned.

5.2 Experimental Setup

The Heuristic algorithm is tested for five different sizes of grid networks. The higher the size of the network, the greater the run times for the shortest path algorithm and the forward searching heuristic. The attractiveness matrix is generated using randomly generated population values. Employment is assumed to be one-fourth the population value. The adjacency matrix and the cost matrix are generated randomly such that the output results are easy to test, analyze and interpret. Different test scenarios and the test statistics computed are explained on a case-by-case basis.

5.2.1 Comparison of Network Quality of Heuristic v/s Ideal Network

In this experiment we apply the heuristic to a twenty five-node problem. For the same network we also apply the Djikstra's algorithmn and compute the accessibility value of the system. The attractiveness matrix is obtained by changing the population randomly in the attractiveness equation. The demographics are changed in the eight replications carried out, while the cost matrix and adjacency matrix for the network remain the same. The equation for Attractiveness between node 1 and node 2 is

$A_{12} = \{(\alpha\ (P_1+P_2)/(P_0)+\beta\ (E_1+E_2)/(E_0))\}$ as mentioned in chapter 3.

The α coefficient of population is 0.5 and the β coefficient of the population is 0.5. This means both the coefficients have a weightage of 50% in the accessibility value. The shortest path algorithm is run to find out the ideal network wherein each node is connected to every other node. The accessibility values of the matrix divided by the cost of the shortest path between nodes gives the ideal routes. A sum of all the accessibilities after dividing by the each individual shortest path and then it addition gives the total accessibility of the ideal system. The twenty five-node network and the corresponding adjacency matrix have been generated to easily compare the results with a manual calculation of the costs of the route. The replications were analyzed to test how the accessibility of a one level search and a two level search compare with an ideal network.

The replications were carried out in such a way that the population was varied in random integer distribution so as to have a high variance distribution and have sufficient variability in the data. The population ranges vary from a low 10000 to a high 200000. A sample cost matrix and adjacency matrix are included in Appendix II in Table 22 and Table 23 respectively. This data was chosen in such a way, that a city might be mapped in terms of population and employment in the real sense. This is done by having areas, which have a low, medium, and high population density. The results gathered from the experiment have been shown in Table15. The Accessibility in the ideal case network is higher than the accessibility of the network obtained from a one level and two level searches. This is because there are 600 paths in the entire network, where one can travel from one node to all other nodes. This additional accessibility is obtained at a higher cost and includes routes that are unnecessary to travel in practical cases.

Table 15: Comparison of Ideal Case with One Level and Two Level Searches

Data Set	Ideal Accessibility	Ideal Cost	Level One Accessibility	Level One Cost	Level Two Accessibility	Level Two Cost
1	87570	2080	49942	1040	48037	1040
2	91862	2080	23978	1040	22215	1040
3	90396	2080	27087	1040	25668	1040
4	90486	2080	24931	1040	26323	1040
5	91786	2080	23665	1040	26438	1040
6	95652	2080	29051	1040	29668	1040
7	91385	2080	25761	1040	26994	1040
8	95156	2080	27179	1040	28345	1040

The paths obtained by the search techniques were compared to the corresponding shortest paths between the same origin destination pairs of each route to understand the utility in the real sense i.e., accessibility value added to the route per unit cost and is illustrated by example below for data set 8. The paths obtained by the two level search method and shortest path algorithm is listed in Table 16.

Table 16: Comparison of Paths Obtained by Shortest Path and Search Technique

Paths Obtained By Two level Search	Two Level Search Accessibility	Shortest Path Accessibility
5,10,11,16,17,12,7,8,9	110	46
13,8,3,2,1,0,5	55	35
12,13,18,19,14,9,4,3	182	126
21,22,17,18,23,24	26	20

On comparing paths in the eight data sets it was clearly observed that the paths obtained by the search techniques had more accessibility added to the system. This is evident from Table 15. The Shortest Path between origin destination pair of N_5, N_9 is N_5, N_6, N_7, N_8, and N_9. This covers only nodes N_6, N_7 and N_8. But the path formed by the two level search methods caters to people from nodes N_{10}, N_{11}, N_{16}, N_{17}, N_{12}, N_7, and N_8. This indicates that the search method covers its neighborhood and then makes a decision on the route to be taken rather than opting for the shortest path between the origin destination pairs. This means it reaches out to more people and thereby increases ridership. Accessibility obtained by the search techniques is better than the accessibility between the same origin-destination pairs along shortest path. This is clearly observed from Table 15. A set of eight replications was observed and in all the cases the routes generated by the search techniques performed considerably better than it's corresponding shortest path.

5.2.2 Comparison of Two Level Search vs One Level Search

The accessibility values are calculated for a system obtained by one level search and two level search techniques. Both these search methods are applied to a nine-node network, sixteen-node network and a twenty five-node network. The adjacency matrix and the attractiveness matrix are left unchanged while the cost was varied within a range of 20 to 120 units in case of tests for each network.

The following results were observed in the test run for the experiments to find out which of the two search methods performed better than the other. The results are described in Table 17.

Table 17: Comparison of One Level Search with Two Level Search

Number of Nodes In Network	One Level Search Accessibility	Two Level Search Accessibility
3x3	6	4
3x3	7	7
3x3	6	6
3x3	7	7
3x3	6	7
4x4	55	65
4x4	63	56
4x4	54	60
4x4	69	64
4x4	65	69
5x5	2638	2638
5x5	2297	2795
5x5	2810	2895
5x5	2704	2914
5x5	2759	3124

It is evident from the table that for a small network the difference is not significant enough to differentiate between one level search and two level searches. As the network size increases two level search performs better, resulting in larger accessibility values. Since the two level search technique looks for one extra segment ahead in comparison to the one level search, it is a more sophisticated and advanced technique and should give better results.

But at the same time, as the network size grows, longer initial routes tends to occupy all the space and thus hinder creation of longer routes for other attractive route seeds. This results in a reduced contribution to accessibility increment from later formed routes. The results show that for some cases one level search performs better than the two level search. This is because though the two level searches looks ahead for segments, the dynamics of the route formation do not allow the second segment look ahead as the segments are already part of other routes. Thus in these cases the two level search starts behaving like the one level search as per the route system dynamics.

Two level search and one level search were carried out on a nine node network and the paths formed are listed in Table 15. Similarly two level search was carried out on a sixteen node network and a twentyfive node network. A set of the results of the one level search outperforming the two level search for the nine node network is depicted in Table 19 and the corresponding graph of cumulative accessibility vs.cumulative cost is plotted in Figure 15. results for a twenty five node network is listed in Appendix II in Figure 27 , Table 27 and 28.

Table 18: Comparison of One Level Search with Two Level Search

		One Level Search			Two Level Search	
Path Number	Path	Cost	Accessibility	Path	Cost	Accessibility
1	6,3,4,5,2,1,0	255	6545	7,6,3,4,5,2,1	285	6224
2	1,4,7,6	405	8662	0,1,4,7,8	450	9550
3	7,8,5	510	9723	8,5	510	9889
4	3,0	570	10072	3,0	570	10238

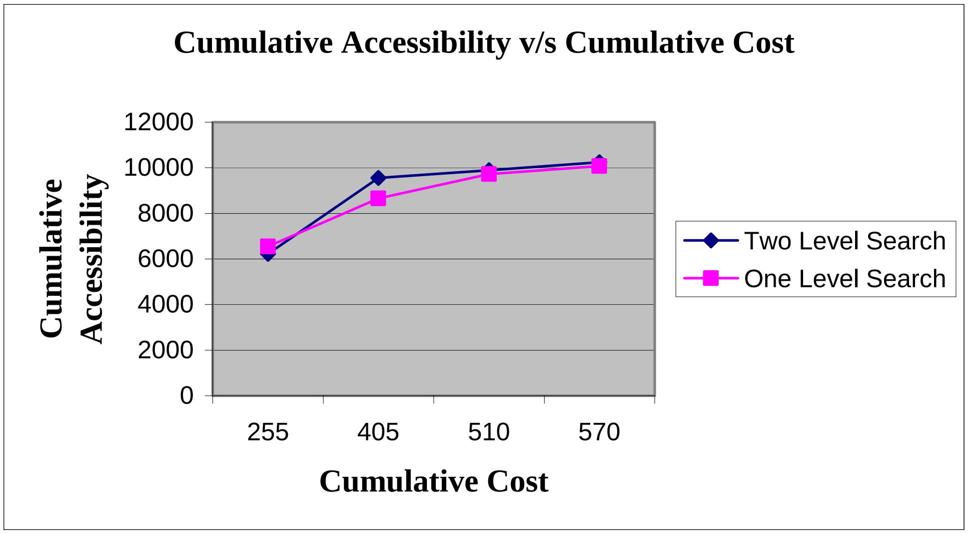

Figure 15: Cumulative Accessibility vs. Cumulative Cost

In the nine-node network for a particular test run it was noticed that the one level search performed better than the two level search. This is because the two level starts behaving as a one level search without being able to obtain two level segments which are already part of the other routes. This result can be seen in the test run for a nine-node network.

Table 19: One Level Search better than Two Level Search

		One Level Search			Two Level Search	
Path Number	Path	Cost	Accessibility	Path	Cost	Accessibility
1	7,4,3,0,1,2	21487	6	0,1,2,5	21630	2
2	3,6,7,8	24770	7	3,6,7,4	20441	3
3	2,5,4,1	19834	7.5	3,4,1	11709	4
4	5,8	5007	7.8	7,8,5	16479	4.5
5				0,3	8646	5
6				4,5	3797	5.5

5.2.3 Effect of Threshold Distance on the Route Formation

In this exercise, paths are formed for various threshold distances and accessibility for the system is measured. Interestingly, the value of accessibility first rises as the threshold distance increases and then drops. This may be explained on the basis that as the threshold distance value increases, the accessibility also increases significantly for first few routes. But at the same time the longer routes so formed block the way in the formation of the remaining routes. As the threshold distance is further increased only fewer number of initial routes contribute for a significant increment in accessibility. This results in overall decrease in accessibility for higher threshold distances. Hence the accessibility increases rapidly first as the threshold distance increases and later it drops. This is shown by Table 20.

Table 20: Comparison of Paths Obtained by Shortest Path and Search Techniques

Threshold Value	Accessibility
10,000	109
20,000	115
30,000	119
40,000	115
50,000	110

The graph of accessibility v/s threshold value is plotted as per the trend observed and is shown by Figure 16.

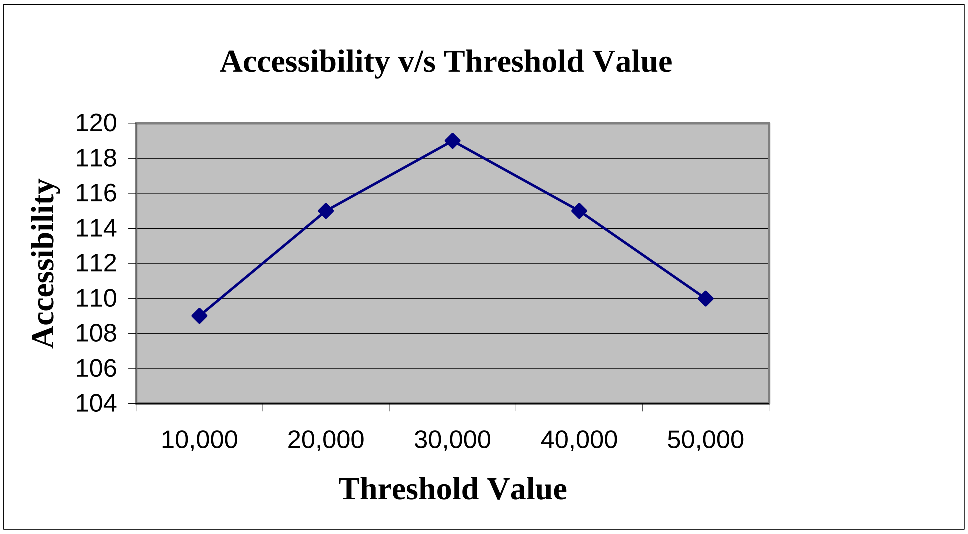

Figure 16: Plot of Accessibility v/s Threshold Value

5.2.4 Effect of Budget on Accessibility

Budget is an important factor in deciding the routes and has a vital role in the total accessibility of the system. As the budget is increased, accessibility also increases to some extent. (It is similar to the fact that an increase in transit facilities will result in increase in transit riders but only up to a particular extent.)

For given route length constraint (30,000), cumulative budget and cumulative accessibility is calculated for routes formed by a two level search and a graph is plotted to depict the trend. It is evident from the Table 21 and Figure 17 that as the cumulative budget increase, cumulative accessibility also increases but up to a particular extent. After a certain limit it saturates and results in a meager increase in cumulative accessibility. Its perfect analogy is to a typical transit network where the increases in number of transit riders saturate after a particular budget.

Table 21: Cumulative Budget v/s Accessibility

Cumulative Budget	Cumulative Accessibility
32903	39
65872	57
103628	74
135319	92
170505	105
182191	109
184355	111
198830	112
203537	115
212313	116
218312	117
227036	118
237716	119
248982	119
259127	119
267326	119
274320	119

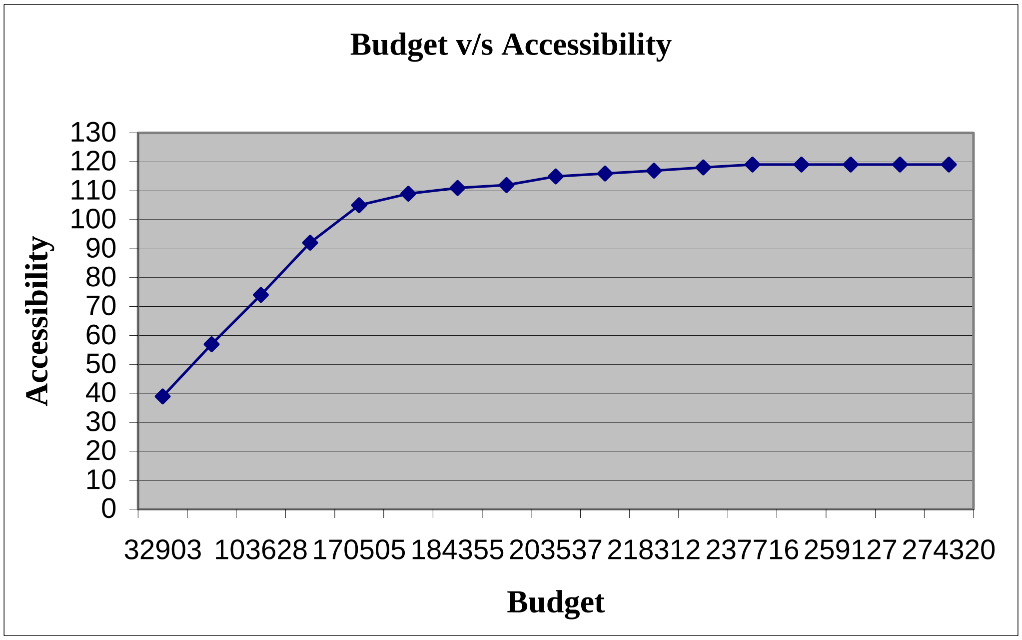

Figure 17: Plot of Accessibility v/s Budget

5.2.5 Effect of an Exponential Cost Function

The accessibility value was also calculated by dividing the attractiveness value by the exponential function of cost, which looks like $\frac{Attractiveness}{e^{\lambda Tij}}$ while forming the path. Here gamma is the value obtained from travel destination model. Typically is a 0.05. The program was modified to accommodate this case but since the cost in the denominator turned very high due to the exponential function, the program gave accessibility values, which were very low as the route length increased. Another contributor to the fact was the values of the cost that were considered.

CHAPTER 6. CONCLUSION AND FUTURE RESEARCH

6.1 Summary

In this thesis, a decision making situation of a transit provider, who has to decide the routes for the transit service based on the activity levels in each area was addresed. The critical aspect of the route formation is to come up with routes, which have a high value of accessibility. The challenge lies in deciding which areas should be served by the transit service such that the value of the route is increased. The difficult part of the entire exercise is to find combinations of sequential segments that comprise logical routes in the context of not only the value of that segment but also the value of each decision in terms of future opportunities for subsequent segments.

A heuristic was developed based on a network-modeling framework and incorporated two search techniques namely the *one level* search and the *two level* searches to output the routes of value. These routes were compared to the ideal system wherein each node is connected to every other node. The search techniques were tested on nine-node, sixteen-node, twenty-five node, thirty-six node and forty-nine node networks. The output obtained mapped the entire service area with fewer routes of better value than an infeasible system of having routes between each and every node. Circuitous routes were avoided. The length of the routes was constrained to practical limits and routes were obtained based on the budget available with the transit provider. The results show that the routes obtained had more accessibility value per unit cost as compared to routes formed between origin destination pairs using the shortest path algorithm. The two level search technique was observed to be better than the one level search in the case of larger networks. A series of tests were carried out on the various kinds of networks as discussed in earlier chapters and the behavior of accessibility value of the system in various test scenarios were observed. Accessibility increases with the increase in budget but then saturates after reaching a particular value. With increase in the threshold distance accessibility increases to a certain limit and then shows a falling trend. The two level search performs better than the one level search as it looks two segments ahead before deciding the immediate segments.

6.2 Conclusion

For the route network design problem based on maximizing accessibility value the following conclusions can be made.

1. The heuristic algorithm produces better routes as compared to routes obtained by the shortest path for the same origin destination pair.
2. The two level search performs better than the one level search in the route formation and more accessibility is added to the system on the whole because of the look ahead technique. The two level search performs better than one level search in larger networks as compared to smaller networks.

6.3 Suggestions for Future Research

The bus routing problem discussed in this thesis is a practical problem encountered by transit providers these days. The program developed could be improved with a user interface feature to make it a commercially viable proposition. Several additional enhancements might be pursued. When routes cross each other the value of both the routes are reduced as the general public has the option of using both and the attractiveness is nearly reduced by half. This exists particularly in perpendicular routes. Thus future routes could attempt to accommodate shared stops. In the current model the moment the route cost goes over the prescribed threshold distance for the first time the route formation is stopped. Actually trimming the path by a node and searching for other viable segments adjacent to the route so that more segments could be served would be a better alternative. The two level search model could have more intelligence if its terminal route segment was selected based on the results of the one level search.

The algorithm currently forms routes until the search space matrix equals the adjacency matrix. In our case a segment is considered for route formation till it becomes part of a route. From that instance the segment is no longer considered for further route formation. The value of that segment is changed to 1 in the search space matrix. These criteria can be changed so as to allow the occurrence of the same segment in different routes.

This effort showed that analytical tools can be developed to support the route planning process while several simplifying assumptions were made to enable computerization the resultant model provided a useful learning tool and perhaps some logic elements that can be used in subsequent initiatives to improve route planning tools.

REFERENCES

[1] Ahuja,R.K., Magnanti,T.L., and Orlin,J.B. (1993). Network Flows: Theory, Algorithms, and Applications. *Parentice Hall,* New York, NY.

[2] Baaj, M.H., and Mahmassani,H.S. (1990). TRUST: A Lisp program for transit route system planning and design. *Transportation Research Record.*no 1283*,* pp 125-135.

[3] Baaj, M.H., and Mahmassani,H.S. (1995). Hybrid route Generation Heuristic algorithm for the design of transit networks. *Transportation Research C3*, pp 31-50.

[4] Baaj, M.H., and Mahmassani,H.S.(1991). An AI-based approach for transit route system planning and design. *Journal of Advanced Transportation*. Vol 25, pp 187-210.

[5] Bansal, A.N.(1981).Optimization of bus route network for fixed spatial distribution. *Scientific Management of Transportation Systems.* North Holland Publishing Company,Amsterdam,The Netherlands,346-355.

[6] Bielli, M., Caramia, P., and Carotenuto,P.(1998). Genetic Algorithm in bus network optimization. *Transportation Research C 10*, pp 19-34.

[7] Ceder, A. and Wilson, N.H.(1986). Bus network Design. *Transportation Research.* 20B, pp 331-344.

[8] Dhingra, S.L.(1980). Simulation of Routing and Scheduling of City Bus Transit Network. *Ph.D.Thesis,* Department of Civil Engineering, I.I.T Kanpur, India.

[9] Dhingra, S.L., Rao, K.V.K.and Muralidhar,S. (2000). Public transportation and routing using genetic algorithmns. Proceedings of the *8th International Conference on Computer-Aided Scheduling of Public Transport* (CASPT-2000), 21 – 23 June, 2000 Berlin, Germany.

[10] Evans, R.J. and Edward, M.(1992). Optimization Algorithms for Networks and Graphs. *Marcel Dekker Inc,* New York, NY.

[11] Goldberg, E. (1996) Genetic Algorithms in Search Optimization and Machine Learning. *Addison-Wesley*, Reading, MA.

[12] Gray, G.E. and Hoel,A.E.(1992) Public Transportation. *Parentice Hall*, NewJersey, NJ.

[13] Harris, B., (2001). Accessibility: Concepts and Applications. *Journal of Transportation and Statistics,* vol.4, no.2/3 pp 15-30.

[14] Harris, B., (2001). Performance of Accessibility measures in Europe. *Journal of Transportation and Statistics,* vol.4, no.2/3 pp 31-48.

[15] Lampkin,W. and Saalmans, P.D. (1967). The design of routes,service frequencies and schedules for a municipal bus undertaking: A case study. *Operartions Research Quarterly.* Vol 18, pp 375-397.

[16] Mandl, C.E.(1984). Evaluation and Optimization of Urban Transit Networks. *European Journal of Operations research,* 6, pp 31-56.

[17] Marwah, B.R., Umrigar, F.S. and Patnaik, S.B. (1995). Hybrid route Generation Heuristic algorithm for the design of transit networks. *Transportation Research Record-99.* Transportation Research Board, Washington D.C., 12-25.

[18] Patnaik, S. B., Mohan, S. and Tom,V.M. (1998). Urban transit bus route network design. *Journal of Transportation Engineering,* vol.124, no.4 pp 368-375.

[19] Shih, M.C., Mahmassani, H.S. and Baaj, M.H. (1998). A planning and design model for transit route networks with coordinated operations. *Transportation Research Record.* no.1623, pp 16-23.

[20] Spasovic, L.N., Boile, M. and Bladikas, A.K.(1994). Bus transit service coverage for maximum profit and social welfare. *Transportation Research Record.* no.1451, pp 12-22.

APPENDICES

Appendix I

An example to show Djikstra's algorithm is shown in Figure 19.

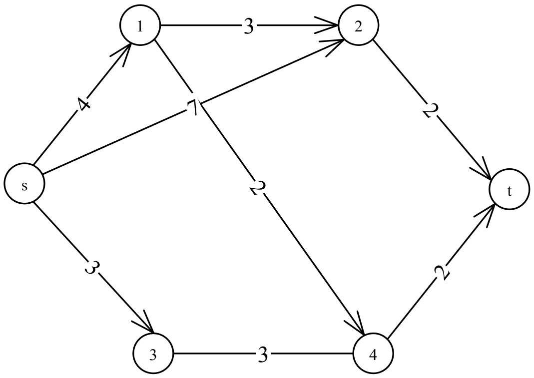

Figure 19: Shortest Path Example Network

Step 1. Initially only node s is permanently labeled, $d(s)=0$. Assign tentative distance $d(x)=\infty$ for all $x \neq s$

Step 2. Compute the tentative distances for the unlabelled nodes in the forward star of y as under:

$$d(1)=\min\{d(1)+a(s,1)\}=\min\{\infty,0+4\}=4$$

$$d(2)=\min\{d(2)+a(s,2)\}=\min\{\infty,0+7\}=7$$

$$d(3)=\min\{d(3)+a(s,3)\}=\min\{\infty,0+3\}=3$$

The minimum distance on any of the unlabelled node is $d(3)=3$, node 3 is labeled and also arc $(s,3)$. The shortest path arborescence consists of arc $(s,3)$ and value of y is 3.

Refer Figure 20 for the first Iteration.

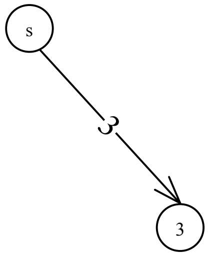

Figure 20: First Iteration of Djikstra's Algorithm

Step 3. Node t has not been labeled so return to step 2

Step 2. $d(4) = \min\{d(4), d(3) + a(3,4)\} = \min\{\infty, 3+3\} = 6$

The minimum tentative distance on the unlabeled nodes is $d(1) = 4$. Label node 1 and arc $(s,1)$, which determined $d(1)$. The value of $y = 1$. The current shortest path arborescence is shown in Figure 21.

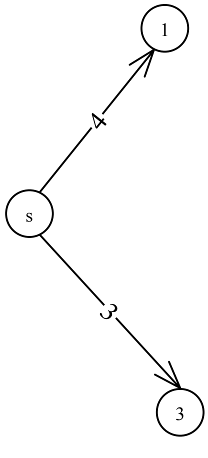

Figure 21: Second Iteration of Djikstra's Algorithm

Appendix I (Continued)

Step 3. Vertex t has not been labeled, so return to step 2.

Step 2. $d(2) = \min\{d(2), d(1) + a(1,2)\} = \min\{7, 4+3\} = 7$

$$d(4) = \min\{d(4), d(1) + a(1,4)\} = \min\{6, 4+2\} = 6$$

The minimum tentative distance on the unlabelled nodes is $d(4) = 6$. Node 4 is labeled and either of the arcs $(1,4)$ or $(3,4)$ are chosen as both determined d(4).Arbitrarily selecting arc $(3,4)$.The shortest path arborescence consists of arcs $(s,3)$, $(s,1)$ and $(3,4)$ is shown by Figure 22.The value of $y = 4$.

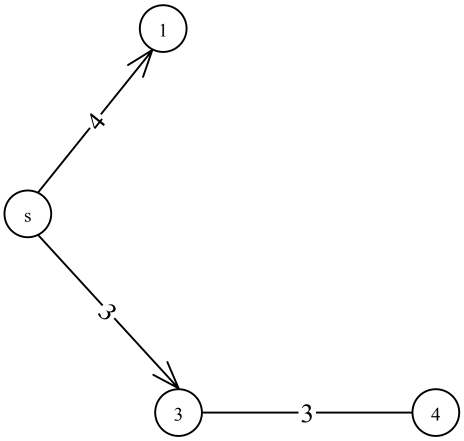

Figure 22: Third Iteration of Djikstra's Algorithm

Step 3. Vertex t has not been labeled, so return to step 2.

Step 2. $d(t) = \min\{d(t), d(2) + a(2,t)\} = \min\{\infty, 6+2\} = 8$

The minimum tentative distance label is $d(2) = 7$, so node 2 is labeled and $(s,2)$, which determined $(d,2)$. The current shortest path arborescence consists of arcs $(s,3)$, $(s,1)$, $(3,4)$, and $(s,2)$. is shown by Figure 23. The value of $y = 2$.

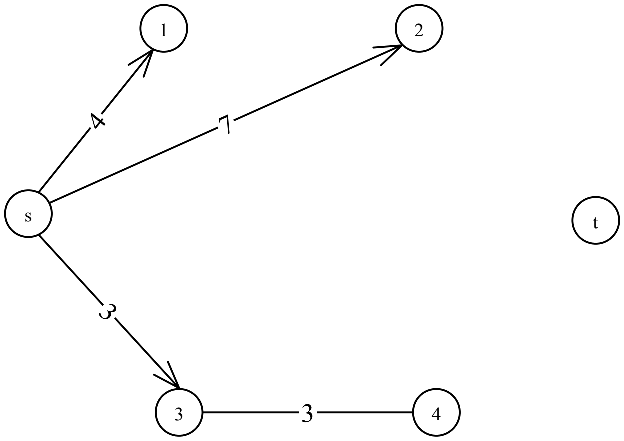

Figure 23: Fourth Iteration of Djikstra's Algorithm

Step 3. Vertex t has not been labeled, so return to step 2.

Step 2. $d(t) = \min\{d(t), d(2) + a(2,t)\} = \min\{8, 7+2\} = 8$

Thus node t is finally labeled. Also arc $(4,t)$, which determined $d(t)$, is thus labeled. The final shortest path aborescence consists of arcs $(s,3)$ $(s,1)$, $(s,2)$. $(3,4)$ and $(4,t)$. The paths arborescence is shown by Figure 24.

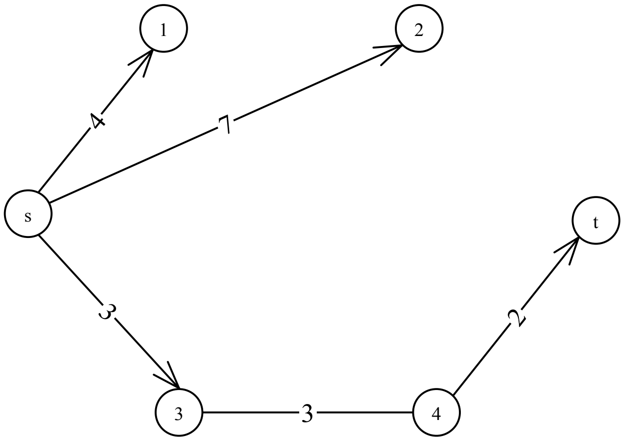

Figure 24: Fifth Iteration of Djikstra's Algorithm

A shortest path from s to t also consists of arcs $(s,3)$ $(s,1)$, $(s,2)$ $(3,4)$ and $(4,t)$ with a length of $3+3+2=8$. This path is not the only shortest path from s to t as there exists a path $(s,1)$, $(1,4)$ $(4,t)$ with the same length 8. A shortest path from s to t is unique if there is no choice with respect to the arcs to be selected for labeling. During labeling if there is a tie as to which node should be labeled, in the case when the $d(x)$ value is the same, then arbitrarily any one of these nodes could be chosen .The other node gets selected in the next iteration of step 2. Djikstras algorithm has been used, as in this case all cost associated on the links of the network are positive. The shortest path algorithm basically consists of only two arithmetic operations, addition and minimization.

Appendix II

Table 22: A Sample Cost Matrix for a Twenty-Five Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}	N_{16}	N_{17}	N_{18}	N_{19}	N_{20}	N_{21}	N_{22}	N_{23}	N_{24}
N_0	0	30	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_1	30	0	20	0	0	0	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_2	0	20	0	30	0	0	0	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_3	0	0	30	0	10	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_4	0	0	0	10	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_5	20	0	0	0	0	0	20	0	0	0	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_6	0	45	0	0	0	20	0	30	0	0	0	65	0	0	0	0	0	0	0	0	0	0	0	0	0
N_7	0	0	45	0	0	0	30	0	10	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0
N_8	0	0	0	10	0	0	0	10	0	30	0	0	0	45	0	0	0	0	0	0	0	0	0	0	0
N_9	0	0	0	0	15	0	0	0	30	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0
N_{10}	0	0	0	0	0	25	0	0	0	0	0	30	0	0	0	15	0	0	0	0	0	0	0	0	0
N_{11}	0	0	0	0	0	0	65	0	0	0	30	0	40	0	0	0	15	0	0	0	0	0	0	0	0
N_{12}	0	0	0	0	0	0	0	20	0	0	0	40	0	35	0	0	0	10	0	0	0	0	0	0	0
N_{13}	0	0	0	0	0	0	0	0	45	0	0	0	35	0	30	0	0	0	20	0	0	0	0	0	0
N_{14}	0	0	0	0	0	0	0	0	0	40	0	0	0	30	0	0	0	0	0	15	0	0	0	0	0
N_{15}	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0	25	0	0	0	10	0	0	0	0
N_{16}	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	25	0	20	0	0	0	15	0	0	0
N_{17}	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	20	0	40	0	0	0	20	0	0
N_{18}	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	40	0	25	0	0	0	30	0
N_{19}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	25	0	0	0	0	0	45
N_{20}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	10	0	0	0
N_{21}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	10	0	30	0	0
N_{22}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	30	0	20	0
N_{23}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30	0	0	0	20	0	30
N_{24}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	0	0	0	30	0

Appendix II (Continued)

Table 23: A Sample Adjacency Matrix for a Twenty-Five Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}	N_{16}	N_{17}	N_{18}	N_{19}	N_{20}	N_{21}	N_{22}	N_{23}	N_{24}
N_0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_2	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_3	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_4	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_5	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N_6	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
N_7	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
N_8	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
N_9	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
N_{10}	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
N_{11}	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0
N_{12}	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0
N_{13}	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
N_{14}	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
N_{15}	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0
N_{16}	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0
N_{17}	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0
N_{18}	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0
N_{19}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1
N_{20}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
N_{21}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0
N_{22}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0
N_{23}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1
N_{24}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0

Appendix II (Continued)

Table 24: A Sample Attractiveness Matrix for a Twenty-Five Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}	N_{16}	N_{17}	N_{18}	N_{19}	N_{20}	N_{21}	N_{22}	N_{23}	N_{24}
N_0	0	56	60	79	46	33	58	58	23	72	79	46	44	79	83	103	58	93	58	83	91	35	56	60	79
N_1	56	0	70	89	56	43	68	68	33	81	89	56	54	89	93	112	68	103	68	93	101	44	66	70	89
N_2	60	70	0	93	60	46	72	72	37	85	93	60	58	93	97	116	72	106	72	97	104	48	70	74	93
N_3	79	89	93	0	79	66	91	91	56	104	112	79	77	112	116	135	91	126	91	116	124	68	89	93	112
N_4	46	56	60	79	0	33	58	58	23	72	79	46	44	79	83	103	58	93	58	83	91	35	56	60	79
N_5	33	43	46	66	33	0	44	44	10	58	66	33	31	66	70	89	44	79	44	70	77	21	43	46	66
N_6	58	68	72	91	58	44	0	70	35	83	91	58	56	91	95	114	70	104	70	95	103	46	68	72	91
N_7	58	68	72	91	58	44	70	0	35	83	91	58	56	91	95	114	70	104	70	95	103	46	68	72	91
N_8	23	33	37	56	23	10	35	35	0	48	56	23	21	56	60	79	35	70	35	60	68	12	33	37	56
N_9	72	81	85	104	72	58	83	83	48	0	104	72	70	104	108	128	83	118	83	108	116	60	81	85	104
N_{10}	79	89	93	112	79	66	91	91	56	104	0	79	77	112	116	135	91	126	91	116	124	68	89	93	112
N_{11}	46	56	60	79	46	33	58	58	23	72	79	0	44	79	83	103	58	93	58	83	91	35	56	60	79
N_{12}	44	54	58	77	44	31	56	56	21	70	77	44	0	77	81	101	56	91	56	81	89	33	54	58	77
N_{13}	79	89	93	112	79	66	91	91	56	104	112	79	77	0	116	135	91	126	91	116	124	68	89	93	112
N_{14}	83	93	97	116	83	70	95	95	60	108	116	83	81	116	0	139	95	130	95	120	128	72	93	97	116
N_{15}	103	112	116	135	103	89	114	114	79	128	135	103	101	135	139	0	114	149	114	139	147	91	112	116	135
N_{16}	58	68	72	91	58	44	70	70	35	83	91	58	56	91	95	114	0	104	70	95	103	46	68	72	91
N_{17}	93	103	106	126	93	79	104	104	70	118	126	93	91	126	130	149	104	0	104	130	137	81	103	106	126
N_{18}	58	68	72	91	58	44	70	70	35	83	91	58	56	91	95	114	70	104	0	95	103	46	68	72	91
N_{19}	83	93	97	116	83	70	95	95	60	108	116	83	81	116	120	139	95	130	95	0	128	72	93	97	116
N_{20}	91	101	104	124	91	77	103	103	68	116	124	91	89	124	128	147	103	137	103	128	0	79	101	104	124
N_{21}	35	44	48	68	35	21	46	46	12	60	68	35	33	68	72	91	46	81	46	72	79	0	44	48	68
N_{22}	56	66	70	89	56	43	68	68	33	81	89	56	54	89	93	112	68	103	68	93	101	44	0	70	89
N_{23}	60	70	74	93	60	46	72	72	37	85	93	60	58	93	97	116	72	106	72	97	104	48	70	0	93
N_{24}	79	89	93	112	79	66	91	91	56	104	112	79	77	112	116	135	91	126	91	116	124	68	89	93	0

Appendix II (Continued)

Table 25: A Sample Cost Matrix for a Sixteen Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}
N_0	0	55	0	0	31	0	0	0	0	0	0	0	0	0	0	0
N_1	55	0	54	0	0	33	0	0	0	0	0	0	0	0	0	0
N_2	0	54	0	88	0	0	99	0	0	0	0	0	0	0	0	0
N_3	0	0	88	0	0	0	0	38	0	0	0	0	0	0	0	0
N_4	31	0	0	0	0	91	0	0	30	0	0	0	0	0	0	0
N_5	0	33	0	0	91	0	24	0	0	69	0	0	0	0	0	0
N_6	0	0	99	0	0	24	0	73	0	0	80	0	0	0	0	0
N_7	0	0	0	38	0	0	73	0	0	0	0	56	0	0	0	0
N_8	0	0	0	0	30	0	0	0	0	33	0	0	36	0	0	0
N_9	0	0	0	0	0	69	0	0	33	0	44	0	0	107	0	0
N_{10}	0	0	0	0	0	0	80	0	0	44	0	84	0	0	69	0
N_{11}	0	0	0	0	0	0	0	56	0	0	84	0	0	0	0	114
N_{12}	0	0	0	0	0	0	0	0	36	0	0	0	0	117	0	0
N_{13}	0	0	0	0	0	0	0	0	0	107	0	0	117	0	87	0
N_{14}	0	0	0	0	0	0	0	0	0	0	69	0	0	87	0	114
N_{15}	0	0	0	0	0	0	0	0	0	0	0	114	0	0	114	0

Appendix II (Continued)

Table 26: A Sample Adjacency Matrix for a Sixteen-Node Network

	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}
N_0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
N_1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
N_2	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
N_3	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
N_4	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
N_5	0	1	0	0	1	0	1	0	0	1	0	0	0	0	0	0
N_6	0	0	1	0	0	1	0	1	0	0	1	0	0	0	0	0
N_7	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0
N_8	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0
N_9	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0
N_{10}	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0
N_{11}	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1
N_{12}	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
N_{13}	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0
N_{14}	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1
N_{15}	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0

Appendix II (Continued)

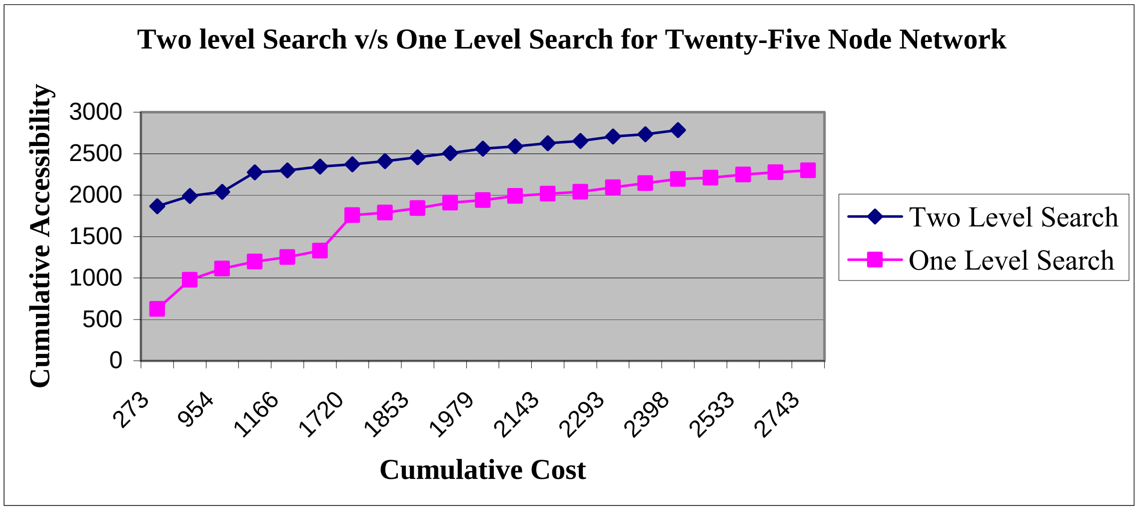

Figure 25: One Level V/s Two Level for Twenty-Five Node Network

Table 27: List of Paths for Two Level Search on Twenty-Five Node Network

		Two Level Search	
Path Number	Path	Cost	Accessibility
1	5,0,1,2,7,12,11,10,15,20,21,16,17,18,19,14,9,8,3,4	1239	1864
2	24,23,22	136	123
3	24,19	47	52
4	5,6,7,8,13,12	375	234
5	23,18	104	24
6	22,17	57	47
7	22,21	92	27
8	18,13	57	40
9	16,11	58	46
10	16,15	62	47
11	11,6	47	56
12	17,12	106	28
13	14,13	61	38
14	10,5	89	28
15	9,4	53	51
16	6,1	101	28
17	3,2	59	52

Appendix II (Continued)

Table 28: List of Paths for One Level Search on Twenty Five-Node Network

		One Level Search	
Path Number	Path	Cost	Accessibility
1	24,19,14,9,8,3,4	273	624
2	24,23,22,21,20,15,16	459	352
3	23,18,13,14	222	135
4	22,17,12	163	85
5	21,16	49	55
6	17,16	37	78
7	16,11,10,5,0,1,2,3	517	427
8	19,18	88	31
9	13,12	45	55
10	12,7	40	66
11	18,17	86	32
12	12,11	55	50
13	15,10	109	25
14	13,8	103	22
15	11,6	47	56
16	9,4	53	51
17	7,6	52	48
18	8,7	66	18
19	7,2	69	36
20	6,1	101	28
21	6,5	109	23