DYNAMIC ROUTING AND SCHEDULING OF A CALL - TAXI SYSTEM

A PROJECT REPORT

by

KOMMA ABISHEK

CE01B050

Submitted in partial fulfillment of the
requirements for the award of the degree

BACHELOR OF TECHNOLOGY
in
CIVIL ENGINEERING

and

MASTER OF TECHNOLOGY
in
INFRASTRUCTURE

(DUAL DEGREE PROGRAMME)

Under the guidance of
Prof. KARTHIK K SRINIVASAN

TRANSPORTATION DIVISION

DEPARTMENT OF CIVIL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

CHENNAI- 600036

MAY 2006

CERTIFICATE

This is to certify that the thesis titled "DYNAMIC ROUTING AND SCHEDULING OF A CALL-TAXI SYSTEM", submitted by **Komma Abishek (CE01B050)** is in partial fulfillment of the requirements for the award of **BACHELOR OF TECHNOLOGY in CIVIL ENGINEERING and MASTER OF TECHNOLOGY in INFRASTRUCTURE (DUAL DEGREE PROGRAMME)**, is a bonafide record of work carried out by him in the Department of Civil Engineering at the **Indian Institute of Technology, Madras**. The contents of this thesis, in full or parts have not been submitted to any other institute or university for the award of any degree or diploma.

Dr. Karthik K Srinivasan
Assistant Professor and Project Guide
Transportation Division
Department of Civil Engineering
Indian Institute of Technology, Madras

Dr. S. Mohan
Professor and Head
Department of Civil Engineering
Indian Institute of Technology, Madras.

ACKNOWLEDGEMENT

I thank my project guide, Dr. Karthik K Srinivasan, Assistant Professor, Department of Civil Engineering, I.I.T Madras, for his continuous support and encouragement through out the duration of my project. The learning process under him has motivated and enlightened me to take up a career in research in Transportation Engineering.

I am grateful to Mr. Suresh of Zig-Zag Call-taxi services Ltd and Mr. Vedagiri of Bharati Call-taxi services Ltd, for giving me inputs on the current practices used in the field of call-taxi services. They also helped me in getting the real-time demand data for the services required for the project.

I express my deep sense of gratitude to my family for their perennial moral support and encouragement.

I also record my special thanks to Venu for helping me sort out bugs in the Java programming. I am grateful to my other friends: Bolshevik, Aparup, Abhishek, Srikant, Balwant and KG for making my stay at I.I.T a very memorable one.

Komma Abishek

ABSTRACT

With the economy booming at an exponential since the last decade, the demand for reliable, speedy, comfortable and direct modes of transportation like call taxis from commuters living in urban areas is continuously increasing. Hence, they have gained immense popularity and with the projections of the demand for these services showing tremendous growth in the future, it becomes necessary that their operations be thoroughly optimized in order to survive the cut-throat competition from other competing operators.

The first objective of this project is to study the existing practices adopted in the field to run these services. The second objective is to propose and develop optimization models and heuristic strategies to increase the efficiency of the operations over the methods followed in the field. The third objective is to test, analyze and validate the models developed through computer simulations.

The models captures many features of the operational problem of a real-world taxi fleet that dynamically moves vehicles between different sites according to customer requests that arrive continuously over time. A minimum cost network formulation was proposed for the off-line version of the problem. Then two heuristic strategies were considered for the real-time version. The comparison of the policies is done under a general simulation framework. Several performance measures like revenue generated, average dead mileage and percentage of calls rejected have been used to quantify the performance of the system. The application developed is shown to serve as an excellent decision support application which helps in making planning and as well as operational level decisions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1. BACKGROUND

The problem of finding efficient routes arises in various contexts like freight operations, emergency dispatching systems, network evacuation and call-bus systems. Naturally, the objective functions, constraints and decision settings vary across these contexts. This thesis focuses on one such context, namely, routing algorithms and heuristics for efficient call taxi operations under a random and time-varying demand environment. The mathematical model and its computer simulation which will be described, is an endeavor to provide a robust solution (user and system optimal) to this call taxi optimization problem.

In the last few decades, several researchers have applied techniques such as meta-heuristics (tabu-search), linear programming, network analysis, simulation, genetic algorithms et al. and took advantage of the tremendous computational ability made possible on a parallel track to accomplish this task. The call taxi optimization problem is a complex and challenging problem (for reasons discussed in Section 1.3.5) which requires the synthesis of techniques drawn from various disciplines ranging from mathematics, network optimization and computational algorithms.

1.2 MOTIVATION

1.2.1 Growing demand

Chennai is India's fourth largest metropolitan city. With an estimated population of 7.60 million (2006), it is the 36th largest metropolitan area in the world. The city is a very large commercial and industrial centre. The city bus service and the suburban train networks have long held sway as Chennai's most popular modes of transport. But a new player, the Call, is gaining ground as an important mode of transport from the last couple of years. Passengers are demanding transportation services that offer a high level of reliability and comfort.

1.2.2 Significant revenues

This rising demand offers a potential to generate high revenues and hence huge profits to call taxi providers, if these operations are effective and efficient. To exploit this opportunity, numerous operators have mushroomed and started their operations to provide call taxi services. Three main services, Bharati, Chennai and Fast Track call taxis are functioning with more than 150 vehicles each in Chennai city. Apart from these they are more than 20 medium and small scale firms trying that have entered the market and drawing significant profits.

1.2.3 Need for optimization

In amidst of this stiff competition there arises an absolute need for developing mathematical models and algorithms which run the operations efficiently. An operational efficiency of even 5 % will significantly increase the revenues generated by each operator.

1.2.4 Need for decisions support tools

In order to improve the efficiency of the operations, the operator has to carefully make planning level decisions (before the beginning of the operations) and also several operational level decisions (during the operations). Applications and tools which can help assist the operator make right decisions need to be developed.

1.2.5 Limitations of currently available tools

The tools that are used in practice do not incorporate any optimization algorithms to improve the operations. They operate based on mere human judgment. The techniques do not have any scientific validation.

Due to the above factors, one can realize a need to develop models, algorithms and applications to improve the efficiency of the operations. This study is an endeavor in that direction.

1.3 PROBLEM STATEMENT AND CONTEXT

1.3.1 Problem statement

Consider a call taxi firm, operating a fleet of ‘m’ taxis which has to pick up and drop-off ‘n’ passengers within a city. The operator has to efficiently allocate and dispatch a taxi to each request. An algorithm or strategy which guarantees maximum revenues has to be adopted to run these operations.

In general, the problem can be classified into one of two modes based on the demand. In the *static* mode all requests are known in advance, this is also refereed to as the apriori problem. Calling it an apriori in the study’s context is a little misleading. It can be more appropriately termed as a *hind-sight* solution than an apriori problem because this problem is solved for a particular request set at the end of the day (when the entire request set that came on that day is known). This solution further is used to compare with the heuristics solutions which are adopted to solve the real time problem. But, in order to avoid confusion with the already existing terminology in the literature, this is referred to as the apriori problem. The following section will make the concept more vivid.

In the *dynamic* mode requests are gradually realized in real-time, which make an apriori planning impossible. Global optimization techniques can be used to solve the apriori problem, where as re-optimization techniques or heuristics based approaches are adopted to solve the dynamic problem.

1.3.2 Definition of terms

1.3.2.1 Decision variables

Fleet size required

This is number of taxis required at the source node (hub-where all taxi start at the beginning of the day) that can service the calls on any given day. It has been observed that a value of 25 to 15 taxis can be used for a typical request set of size 150-300 calls per day. The objective is to estimate the value of average minimum fleet size required to service all the requests without significant delay.

Call off-set time (CO)

This is time just before the pick-up time of the call that a certain taxi request is actually processed and allocated a taxi. A call off-set time of 45 minutes is typically used in practice.

1.3.2.2 Performance measures

Percentage rejection due to fleet constraint

It is the percentage of the total calls that are not serviced and hence rejected due to the fleet constraint. Each simulation will start with a constant fleet size and at some point in the simulation all the taxis might be already engaged and at this particular state of the system any call might not be able to be serviced. This measure gives an estimate of the lost revenues due to insufficient fleet size.

Percentage calls late serviced

It is the percentage of the total calls that are pick-up after the specified pick-up time of the request. Each simulation has a particular call-offset time. The taxi allocated to a certain request might take time greater than the call-offset time to reach the origin of the request, which will result in the delay of the pick-up process of that particular request. This measure gives an estimate of the level of service that the system offers to its customers.

Car waiting time

If the time taken to reach the origin of the request once the taxi allocated to the request (due to a sufficient call off-set time) is lesser than the pick-up time of the request, then the car has to wait at the origin of that request before the pick-up occurs. This gives a measure of the taxi idle time and hence the efficiency of the allocation process.

Average dead mileage

For instance a request requires a service from a node A to node B, and for this request say a taxi is allocated which is currently at node P, after the drop-off at node B the taxi moves to a node Q. In the above trip AB is the revenue generating path and the distances PA and BQ contribute to the dead mileage of the trip. Average of the dead mileage of all the trips gives a measure of the idle run of the taxi per trip. In order to make profit the revenue generated from the path AB must be greater than the operational cost incurred to travel from P to Q.

Revenue

This is the money generated in servicing the calls. As explained in the above example the revenue generated is the difference of money generated in the path AB and cost incurred in traversing from P to Q via A and B. The tariff policy of 12 Rs/km for the revenue generating path and Rs4/km for the operational cost has been used in the simulation.

Closing revenue refers to the difference of the total revenue generated and the operational cost of vehicles including the cost of termination (empty return of the taxis to the hub). Non closing revenue is the revenue generated without the cost of termination.

1.3.3 Objectives

1. To develop models (optimization) and techniques (heuristics) to optimize the call taxi problem. In other words, develop models which maximize revenues and minimize fleet size required, average dead mileage, car-waiting time, % rejection, % late serviced.
2. To test and validate the reliability of the models proposed with real time data.
3. To develop a decision support application based on the above models to provide a robust solution which guarantees an increase in the efficiency of the system.

1.3.4 Scope

1. This is limited to operations inside the Chennai city region. For this purpose, Chennai transit network has been divided into 60 zones and the travel times on these links is assumed to be constant.
2. It is also assumed that the drivers do not use their judgment or knowledge in making routing and scheduling decisions. They receive directions from the system operator and they strictly adhere to the instructions.
3. A single source node (Hub) where are the taxis start and return at the beginning and end of the simulation time respectively has been assumed.
4. The inherent assumption for taxi services, taxis serve only one job at any given point of time exists. In other words, the option of car pooling is not considered.
5. It is assumed that all the taxis move with constant speed.
6. The study is limited to analysis and interpretation of data obtained from a single cal taxi firm.

1.3.5 Difficulties in addressing the problem

1. All optimization problems have limited resources. Here, fleet size and time are the crucial resources. All allocation and dispatching decision should aim at using these resources very efficiently.
2. Another difficulty is the demand. The demand is dynamic and an unknown function of time. Future demands that will arise are not known. So a global optimization is not possible unless all the demands are known apriori. The

operator has to resort to heuristics or re-optimization techniques to tackle the case of dynamic demand.

1.3.6Applications

1. ***Planning level decision support:*** The simulation of the models will provide insights and help the operator make crucial planning level decisions like the heuristic to be used and size of the operating fleet to be maintained.
2. ***Operation level decision support:*** Once a model is selected and the system is setup the model chosen will help in making operation level decisions like optimal taxi to be allocated to any particular request, monitoring the course of the taxis after the drop-off to enhance the efficiency of the operations.
3. ***Analysis Insights:*** The various simulations of the models under different scenarios will help in estimating the revenues and other performance measures. A good performance will indicate increase in revenues, % of calls serviced, decrease in operating fleet required, taxi-idle times, and average dead mileage.

1.4 STRUCTURE OF THE THESIS

In this chapter, the importance of developing models for optimizing the operations of the call taxi system was discussed. The objectives have been clearly laid down for this purpose. The importance of this study and its envisaged benefits are also presented.

Chapter 2 presents a literature review of previous solution approaches to model and solve taxi service problems. Based on the technique and methodology adopted to model the problem, they have been classified into seven categories. Each category or approach is then described briefly. Later it is followed by the summary of the literature review to give a broader picture.

Chapter 3 will take one through the data collection process. It also explains in great detail the lessons learnt in the process along with the routine developed to refine and customize the raw data to suite the models developed.

Chapter 4 introduces the terminology used in the static instance i.e. the ‘Apriori problem’. The assumptions, objective and constraints of this model are described. The formulation of problem as a minimum cost network formulation is described in the next section. The solution methodology and method of implementation in JAVA are also described. A numerical example is given to illustrate the model. Finally, the conclusion section summarizes the advantages and applications of this model.

Chapter 5 starts with describing in detail the practice adopted in the field. Then it proposes several heuristics explaining the logic behind adopting them in the first place and then the algorithm used in each of the proposed heuristic. Then the methodology with definition of the terms used in the dynamic instance and the implementation of the simulation of the heuristic is laid out. This is followed by a small illustration and the conclusion section summarizes the chapter.

In Chapter 6 the results, analysis and inferences of the various models along with the sensitivity analysis are discussed.

In Chapter 7, conclusions of the study are drawn, followed by scope for further research in this area.

1.5 SUMMARY

This chapter introduced the need and importance of this study. Later, it describes in detail the problem features-terminology, objectives, scope, complexity and applications of the problem. The next chapter gives insights into the work that has been attempted by several researches in the past to solve similar fleet management problems.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this chapter, the literature on dynamic routing and scheduling problems is reviewed. To formulate and solve such routing and scheduling problems various approaches have been tried, tested and implemented in the past. The approaches used can be classified into various categories based on the type of formulation and the technique used to solve that particular formulation. Largely the problem has been formulated as a mixed-integer programming, linear programming, dynamic linear programming, and minimum cost network flow problem in literature. To solve the above formulation, techniques varying from genetic algorithms, benders algorithm, minimum cost network algorithm (successive shortest path, relaxation algorithm) have been used. Techniques which adopt local and global heuristics have also been investigated. Recently, a technique based on GPS tracking has evolved and is being successfully used in East-Asian countries like Singapore and Japan. Several studies have investigated the use of taxi information systems on the efficiency and quality of the taxi services. Work on developing and testing real time heuristics that can be easily adopted to manage the services have also been addressed by some researchers.

2.2 THE VARIOUS APPRAOCHES

The type of formulation and the technique adopted to find the solution mainly depends on the constraints, scope and the complexity addressed in the problem.

The following seven categories have been identified based on various methodologies adopted and tested in literature:

1) Insertion and genetic algorithm approach
2) Trip chaining strategy for advanced taxi booking
3) Network equilibrium taxi model approach
4) Effects of taxi information systems on efficiency and quality of services
5) Mixed integer programming formulation approach
6) Real time heuristics approach

The following sections will give will give a brief description of these methodologies.

2.2.1 Insertion & genetic algorithm approach

Liping Fu and Stan Teply et al (1999) tried to develop a model for a call-bus transit system which shares the same features of the call taxi problem. This paper presents the general concepts, models, and computational techniques applied in vehicle routing and scheduling system. They used an insertion algorithm to insert the customer requests. Software called 'First-Win' was developed which incorporated the insertion heuristic for this purpose. It inserts the new customer into the schedule of the vehicle which results in a least possible insertion cost. The developed software explicitly considers travel time variability in urban roadway networks. Advanced computational methods applied in the system, such as the artificial neural network technique, allowed heuristic estimation of origin-destination travel times in a dynamic and stochastic fashion, contributing to the processing speed required to respond expeditiously and efficiently to the user requests.

The objective of this model is to improve the responsiveness, reliability, and productivity of the services. In other words, minimize the fleet required, total vehicle time, customer service time deviations, and customer excess ride times to run the services. The constraints were limited fleet, vehicle capacity, pickup-drop off time windows, and maximum ride time for the customer.

A real scheduling problem from the city of Edmonton, Alberta, was used to illustrate the positive computational experience and the capability of the developed software to handle both off-line and on-line operations.

2.2.2 Trip chaining strategy for advanced taxi booking

Lee, Cheu and Wang et al. (2003) in their study classified the bookings for the services into two categories current and advance. Current bookings are those whereby the customer makes a booking less than half an hour before the taxi is required to reach him or her (most current bookings require taxi companies to dispatch taxis immediately or as soon as possible), and advance bookings are requests made at least half an hour in advance. In this paper, the focal point was on the advance bookings. An improved strategy was proposed over the existing model. In the existing dispatch system, once an advance booking demand comes, the dispatching center broadcasts this booking information immediately to all the taxis, both occupied and empty taxis, since the advance bookings are to be served at least half an hour later. The job will be assigned to the first taxi driver who bid for it. This type of a dispatching system is obviously very inefficient and will result in a very bad performance of the system. In the proposed system chained trips were planned and offered to taxi drivers as a package. This means that several bookings with demand time points that are spread out within a reasonable period of time, and with each pick-up point coinciding or within close proximity to the previous drop-off location were chained. The shortest time paths as generated by this proposed dispatch system based on “real-time” traffic conditions for each job were linked up to form properly planned routes to be offered as a multiple-booking package to the taxi drivers. This helped the drivers to minimize their empty cruising times, as the time was spent fulfilling these advanced demands instead of cruising around in search of customers.

The proposed dispatch system has been modeled as a pickup and delivery problem with time windows (PDPTW). A two-phase method (construction and improvisation) has been deployed to solve the PDPTW problem. This successful approach for solving PDPTW is to construct an initial set of feasible routes that serve all the customers (construction phase) and subsequently improve the existing solution (improvement phase). This two-phase method comprises the insertion algorithm and the tabu Search. Several insertion algorithms such as the least cost insertion algorithm, nearest neighbor insertion algorithm as well as the sweep insertion algorithm along with the proposed minimum time window algorithm have been analyzed. The earliest time window insertion algorithm generated an initial solution efficiently in terms of low computational cost and delivering a solution of reasonable quality. The initial feasible solution is then improved in the improvement phase using tabu search to avoid the search from revisiting the same solution in near future. A tabu list that records the n previous moves performed is maintained in memory. A move is considered tabu if it is in the tabu list. Moreover, a move is “aspired” if the resultant cost is lower than the cost of the best solution encountered.

A portion of the Central Business District (CBD) area in Singapore, which is bounded by the Electronic Road Pricing (ERP) gantries, covering an area of approximately 3.0 km by 2.5 km was used for the simulations conducted in this research. Under the proposed system several benefits to the passengers, drivers and the operators were envisaged. The taxi companies would be able handle a higher throughput of bookings with the same resources and reduced empty cruising. The drivers for obvious reasons would be willing to accept a packaged advance booking than a single current booking. The customer would enjoy the high level of service (strict time windows) since strategies adopted increased the efficiency of the operations significantly.

2.2.3 Network equilibrium taxi model approach

Yang and Wong et al. (1998) made an attempt to characterize taxi movements in a road network for a given customer origin-destination (O-D) demand pattern. For this purpose a network equilibrium model was proposed to offer some interesting insights into policy-relevant results for decision making. The model will predict how vacant and occupied taxis will cruise in search for customers and provide services. The effects of the fleet size and uncertainty in the system were also highlighted.

The model described by can be summarized as below:

Consider a road network G (V, A) where V is the set of vertices (nodes) and A is the set of arcs. In any given hour, the number of customers demanding taxi ride from origin zone I to destination zone j is D_{ij} (trips/hour), D_{ij} is assumed to given and constant (no demand elasticity). Let I and J be the set of customer origin and destination zones respectively.

So we have, $O_i = \sum_{j \in J} D_{ij}$ and $D_j = \sum_{i \in I} D_{ij}$. Let h_a be the travel time on the link $a \in A$ and h_{ij} be the travel time through the shortest path from i to j.

Taxi service time constraint:

$$TOT = \sum_{i \in I} \sum_{j \in J} T^{o}_{ij} h_{ij}$$

$$TUT = \sum_{i \in I} \sum_{j \in J} T^{v}_{ji} (h_{ij} + w_j)$$

Where, TOT is the total occupied and TUT the total un-occupied time.

T^{o}_{ij} is the occupied taxi movement from zone i to j.

T^{v}_{ji} is the vacant taxi movement from zone j to i.

w_i is the taxi waiting time at i.

Total service time N = TOT + TUT.

The probability that a vacant taxi originating in zone $j \in J$ meets a customer eventually in zone $i \in I$ is specified by the following logit model:

$$P_{i/j} = \frac{e^{-\theta(hji+si)}}{\sum_{m \in I} e^{-\theta(hjm+sm)}}, \forall i \in I, j \in J$$

Where θ is a non negative parameter that is calibrated from the observed data. In a stationary equilibrium state, the movements of vacant taxis over the network should meet the customer demands at all origin zones.

So, $\sum_{i \in I} D_j \times P_{ij} = O_{i,} i \in I$. This represents the system equilibrium.

While most previous studies as we have seen have been based on an abstract, aggregate demand and supply model or based on a simplified, specific structural model, this model mathematically formulates it as a transportation network problem. *Yang and Wong et al.* first presented the network model for taxi operations as a mathematical optimization problem. The model was designed to explicitly consider the effects of the taxi fleet size and the uncertainty on the system performances as mentioned. *Wong et al.* improved the network model by incorporating variable demand and multi-class vehicle assignment. Their bi-level problem combined both taxis traffic and normal traffic in a network equilibrium model to find the optimal taxi operation pattern. In the same line of their previous work, recently *Yang et al*. further investigated the nature of demand–supply equilibrium in a regulated market for taxi service from a case study for the city of Hong Kong. Their network models contributed to analyzing taxi services by providing a network equilibrium model for the taxi service problem. However, their model framework is based on static equilibrium where time-variant effects are not considered, and the model has limitation in modeling detailed operational characteristics.

2.2.4 Effects of taxi information systems on efficiency and quality of services

Kim, Jun and Jaykrishnan et al. (2005), in their study developed a simulation model for urban taxi services in a dynamic and stochastic network. The model is based on a simple learning model to represent driver's destination choice behavior. In modeling taxi drivers' learning process, they employed a day-to-day evolution approach. Even though the overall structure of the model is based on the day-to-day evolution approach, the taxi model included a with-in day learning process as well to reflect the nature of taxi drivers' repeated travel. In the model, drivers acquire knowledge on network and passenger demand only from their experience, and their knowledge was inductively updated. The taxi drivers' passenger seeking behavior was modeled based their expected travel time and expected waiting time.

The developed simulation model gives good insights for urban taxi service in dynamic situation. Through a simulation experiment, this study identified several interesting points for taxi information system. First, from a drivers' day-to-day learning behavior analysis, it was found that taxi drivers could improve their capability in predicting recurrent traffic condition, but they could not acquire enough knowledge on taxi demand. Second, despite taxi drivers' improved knowledge on network condition from their experience, the operational efficiency and the quality of taxi service may be not improved. Third, the taxi information system helps drivers efficiently seek passengers and reduces unnecessary travel. Lastly, the taxi driver information system can provide benefit equivalent to increasing the number of taxis by 20% in terms of the quality of taxi service.

2.2.5 Mixed integer programming formulation approach

Yang, Jaillet and Mahmassani et al. (2002) have proposed a mixed integer programming formulation for the offline version of the multi-vehicle truck load pick-up and delivery problem. In the problem, they considered a trucking company with a fleet of K trucks. Each truck can carry only one job (request for service) at a time, and cannot serve another job until the current job is delivered to its final destination. At the arrival time of a request, the company is given the pickup location, the delivery location, the earliest pick-up time, and the latest delivery time of the job. The company can either accept or reject a job request within a small prescribed amount of time. The revenue generated from a given accepted job is proportional to the length of the job, defined as the distance between its pick-up and delivery locations. Completion beyond the latest delivery time is allowed but penalized, and the penalty is proportional to both the job's length and the amount of delay occurred. In case a job request is not accepted, the cost of rejection is the gross revenue the company would have otherwise obtained had it accepted the job. Over the course of serving the sequence of requests, the company incurs additional operating costs proportional to the empty distance traveled by trucks in order to serve the accepted jobs. An assumption that the trucks all move at the same constant unit speed is also made.

The above constraints are converted to a mathematical model. In other words, the problem was modeled as an assignment problem with timing constraints. The assignment problem, in turn, consists of finding a least-cost set of cycles going through all the nodes of (1, ...,K, K +1, ..., K +N), where node k for k = 1, ...,K corresponds to truck k and node K +i for i = 1, ...,N corresponds to job i.

2.2.6 Real time heuristics approach

2.2.6.1 Simple heuristics

Nagasayan Alla et al. (2004), a student at IIT Madras has developed an elementary heuristic approach to address the call-taxi operation problem for the Chennai city. The objective was to minimize the dead mileage of the fleet and the caller waiting time. The demand distribution was assumed to be spatially uniform with a mean gap of 5.4 minutes every call. A computer simulation of this model was developed to run and test the strategies developed. Two strategies basically differ in what the taxi does after the drop-off occurs. In one case, the taxi stations itself at the drop-off node until it receives a new request, while in the second strategy the taxi moves to the nearest ‘desirable’ node (where the expected waiting time and probability of getting a call is high). In the latter case it was cleverly taken care that accumulation of the taxis does not take place at any particular node by introducing an upper limit to the number of taxis that can be present at a node. Analysis showed that the latter strategy if adopted improved the performance of the system in terms of decreased taxi idle and taxi waiting times.

2.2.6.2 Advanced heuristics

Yang, Jaillet and Mahmassani et al. (2002) work also encompassed the use of five different heuristics to model the problem. Finally these heuristics along with the integer programming formulation were compared in terms of performance of the system. In all the policies considered, a truck remains idle at the destination of its last job when not assigned to a new job.

The first policy called the ‘Bench’ reflects what a company might do without the aid of sophisticated decision support systems. At a job arrival epoch, bench decides whether or not this new job is accepted, and, if accepted, assigns it to the queue of a specific vehicle k. These decisions are permanent and are based on a sequential evaluation. For each truck k, bench calculates the marginal cost of serving this new job if inserted at the end of its queue. In case all marginal costs are higher than the cost of rejection, the job is rejected. Otherwise it is assigned at the end of the queue of the truck k with the lowest marginal cost.

The second and the third policies are called ‘NE’ and ‘SE’. Here, initial acceptance/rejection decisions are not necessarily permanent, and a job being accepted or rejected at one decision epoch could be reconsidered before a permanent decision has to be made. They evaluate the insertion of each pending job, one at a time, into each truck’s queue. Each pending job is either inserted into a particular queue if the corresponding marginal cost is the smallest among all queues and is smaller than the cost of rejection, or is tentatively rejected otherwise. The two local policies differ in how they consider insertion of a pending job in a truck's queue. NS does not modify the relative ordering of the jobs already in the queue and only considers all possible insertions in between these jobs. It also doesn't consider rejecting a tentatively accepted job in a queue while trying to insert the pending job. On the contrary, SE evaluates all possible orderings of the original waiting jobs together with the current pending job, and does so by solving a one-truck instance of the off-line problem. The optimal solution determines whether the current pending job and previously tentatively accepted jobs of the queue are accepted and, if accepted, in what order in the queue they should be served. If the pending job or a previously tentatively accepted job becomes tentatively rejected, it is added to a temporary list.

They further proposed two more re-optimization policies which consider, in one optimization run, all trucks, all acceptance/rejection and allocation decisions of pending and tentatively accepted waiting jobs, and all reallocation decisions of permanently accepted waiting jobs. 'Myopt' optimizes the acceptance and (re-)allocation decisions as if no future new job would ever be requested. It corresponds to solving a full instance of the off-line problem. 'Optun' operates in almost the same way as Myopt. The only difference is that Optun introduces opportunity costs of serving jobs, somewhat accounting for future job requests. It assumes some knowledge about the probability law of future job pickup (and delivery) locations.

Ultimately they found that the policies based on fully optimizing the off-line model of the problem perform very competitively with other policies under typical cost structures. The best policy they found is the one that takes some future job distribution into consideration namely ‘Optun’.

2.3 CURRENT STATE OF PRACTICE IN CHENNAI

There are more than 50 operators according to a news article (Hindu online edition, 2005) running services in Chennai today. Three operators Zigzag, Bharati and Fast track services dominate a good 90 % of the market share. Each major operator has typically a fleet of 100-200 vehicles operating. A wireless voice transmitter is installed in every vehicle to communicate with the central sever located at the head-quarters which guides and monitors the location and the operation continuously.

The methodology adopted to operate and run services is very elementary and involves human judgment in making decisions at every stage. The fleet of vehicles is divided into 3-4 zones, each zone having typically 40 vehicles and is controlled by a dispatcher. Each vehicle is given and identified by a vehicle ID. Each operation broadly goes through the following three phases - booking, allocation, dispatching.

In the booking phase, the customer places his request (if feasible) dropping the necessary details like pick up and drop off location along with pick up time. The pickup time is off-setted by 15 minutes in order to avoid any unexpected delays. This request is sent for allocation. The nearest available taxi to the pick up location of the customer in all the zones will be allocated for this request and sent for dispatching.

The dispatcher of the vehicle who controls that particular zone dispatches this taxi. He is responsible to keep track of this operation and ensure that the vehicle reaches the customer in time and after the drop off, the trip length, revenue generated is updated from the driver to the central server. This taxi is then marked idle and is ready for being allocated to the next request.

Custom made software is being used by each operator to run and keep track of its day to day operations. The application console is split into three screens – The request (customer calls to be served), free taxi (with location and idle time) and in service taxi (location) windows. The request is picked and allocated a taxi from the free taxi window based on its location (nearest) and if two taxis are available close enough to the pick up location the taxi with a higher idle time is picked up to ensure a degree of equity amongst the drivers.

2.4 GAPS IN LITERATURE

1) Several studied have either modeled the static (apriori) or the dynamic versions of the problem independently. This study proposes techniques for offline and the online versions of the problem and tries to quantify the improvements in the performance of the models if information were known apriori.
2) Theoretical models have been but tools and applications which incorporate the model have not been built. In this project, a software application in Java is coded which incorporated the various models.
3) The models already developed have not given particular importance to the sensitivity of the decision variables to the performance of the system. In this study, a detailed sensitivity analysis is carried to assess the importance of the various parameters like fleet size and call-offset time in the model.

2.5 SUMMARY OF THE LITERATURE REVIEW

Insights into several categories of approaches have been discussed. One can very well appreciate the complexity of the problem at hand through the above discussion. In many studies several initial assumptions have been made to simplify the problem and have been later relaxed by other advanced researchers in the field to widen the scope of the problem. It can be summarized that the main components that need to be modeled to get a comprehensive catch of the problem are 1) The demand distribution (function of time and space), 2) transit network, 3) driver behavior, 4) the operations (strategies adopted) and 5) tariff policy. Several performance measures like taxi and passenger waiting times, taxi idle times, revenue generated, fleet size used, dead mileage incurred have been modeled to quantify the performance of the system. Techniques varying from genetic algorithms to integer programming have been used to formulate the offline and the online versions of the problem. The recent work of Yang and Wong gave a new dimension to the modeling aspect of the problem. New technologies like GPS and taxi information systems and increased computational abilities have been utilized and its effects on the performance of the system are also been explored.

The next chapter explains the data collection and extraction process. Crucial elements that need to be modeled are also discussed with flowcharts. The lessons learnt in the process are also listed in detail.

CHAPTER 3

DATA COLLECTION & EXTRACTION

3.1 INTRODUCTION

This chapter will take you through the data collection process. The important lessons learnt through this process are highlighted in the next section. The flowcharts describing the operations will try to reinforce the methodologies used in a typical call-taxi service. Then, the crucial elements that need to be modeled in a call-taxi operation system are identified and explained. Additional flowcharts which describe the data extraction and refining processes are also laid out. This refined data is used in the simulation of the models that will be explained in chapters 4 and 5.

3.2 DATA COLLECTION

Efforts were put in collecting real time taxi service data once the models were fully developed and tested on a sample data. Several call taxi firms were contacted for this purpose. The need for the data along with the relevant problem features, improvements and benefits was fully presented and explained in detail. The methods adopted as seen were very crude and based on the rule of thumb and their experiences in the field with no proper scientific basis. Hence, there was big window of opportunity and scope to build models and relevant applications to suitably increase the efficiency of their operations. Valuable lessons were also learnt in the process, which will be described in the next section. Finally, a call taxi firm offered their data for the full month of January, 2006.

This quantum of data was very helpful in producing reliable and consistent results which will be explained in chapter 6.

3.3 LESSONS LEARNT IN THE PROCESS

All the operators contacted were very helpful and provided with a lot of inputs on the strategies they adopt and constraints they face to run the services.

3.3.1 Constraints in practice

1. Installing and maintaining a GPS system in every vehicle and purchasing software to coordinate with the GPS data required high initial capital investment. But, the lack of a GPS systems attached to vehicles posed a host of other problems like: controlling the drivers location (leverage to use it to his advantage since, the exact location of the vehicle cannot be tracked by the operator) and speed (complaints of over-speeding)
2. Tamil Nadu state regulates the operations of the taxis, which do not allow the operators to make pick-ups at crucial locations like airport and railway stations (for the benefit of local, individual drivers) where the maximum demand for the services among all the zones in the city exists.
3. With manual allocation of trips to the vehicles, the operator faced a serious problem of not ensuring equity among drivers (the commission the drivers get is based on the number of trips he is allocated to).

4. Driver’s knowledge of the city routes often chokes the process of allocating a particular request (to out-skirts of the city) which has a high return if accepted to the nearest available free taxi.

3.3.2 Flow chart - Crucial elements to be modeled (step-by-step)

Chapter 2.3 described in detail the practice followed by call-taxi firms in Chennai to handle the operations. The flowchart in the figure 3.1 below will give further reinforce the idea and it will also identify the crucial elements and processes that needs to be modeled to run a call-taxi service efficiently.

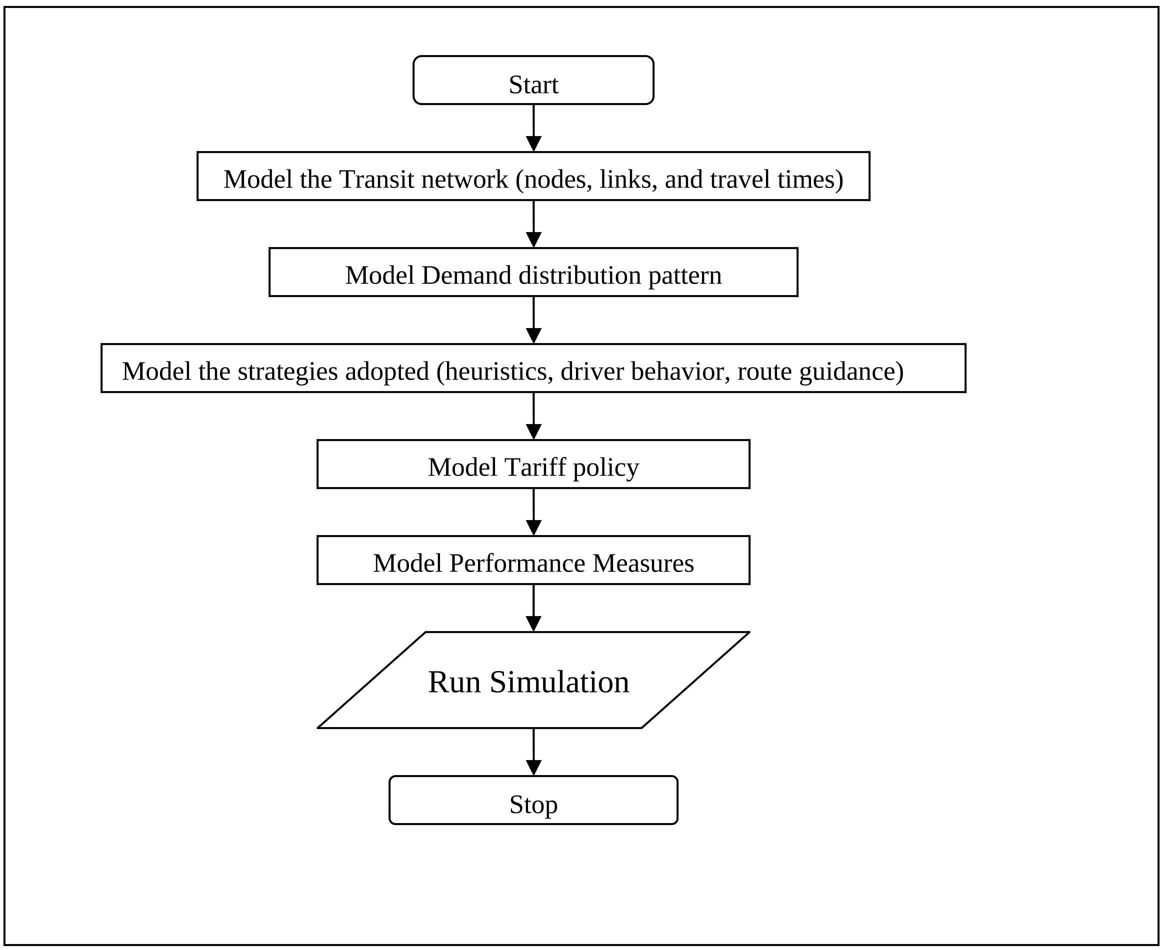

Figure 3.1 Crucial elements to be modeled

The above flowchart highlights the elements that need to be modeled to run a call-taxi service efficiently.

3.3.2.1 Transit network

One has to model the transit network of the city in which the operations are to be carried out. It should encompass the map, overlapped with nodes (potential places where the deans might originate and terminate), links (road network, travel time, distance) and shortest path between the nodes For the purpose of our simulation, we divided the Chennai map is divided into 60 zones.

The origins and destinations are revised in terms of these zone ids for easy computation. The centroid of each zone is considered as a node and the distance between any two particular such nodes gives the arc length connecting these nodes in the transit network inter-zonal. The nodes and arcs put together constitute the transit network.

The transit network is represented by an adjacency matrix. The list of the adjacency matrix tables are given in the appendix.

A more complicated version of the transit network should also take into account the congestion which result s in variable travel times. But, in our study it is neglected for the sake of simplicity.

Figure 3.2 Chennai zonal map

The table below gives the list of zones identified for the operations in Chennai city:

Table 3.1 Zone IDs, Zonal areas

ZONE ID	ZONAL AREAS		
1	CHEPAUK		
2	TRIPLICANE		
3	MYLAPORE	MANDAVELI	
4	ROYAPETAH	SATYAM THEATRE	
5	ALWARPET	TTK ROAD	
6	RK SALAI	GOPALPURAM	
7	PATTINAPAKKAM		
8	R.A.PURAM	SAVERA	
9	ADYAR	I.I.T	
10	BESANT NAGAR		
11	THIRUVANMAYUR		
12	PALAVAKKAM		
13	THARAMANI		
14	THURAIPAKAM		
15	MINT		
16	PARRYS		
17	CENTRAL		
18	EGMORE		
19	PURASAIVAKAM	PATTALAM	
20	DAS PRAKASH	SOWCARPET	MANNADI
21	KILPAUK		
22	GREAMS ROAD		
23	NUNGAMBAKKAM		
24	KODAMBAKKAM		
25	CHOOLAIMEDU	POONMALAE	BREEZE HOTEL
26	ANNA NAGAR		
27	T NAGAR		
28	MOUNT ROAD	CIT NAGAR	GEMINI
29	WEST MAMBALAM		
30	SAIDAPET	TAJ COROMANDEL	GREAMS APPOLLO
31	KOTTURPURAM		
32	NANDANAM		
33	GUINDY		
34	BUTT ROAD		
35	ASHOK NAGAR		
36	EKKATUTHANGAL		
37	VADAPALANI		
38	KK NAGAR		
39	JAFFERKHANPET		
40	ADAMBAKKAM	VALASARAVAKKAM	
41	AIRPORT		
42	SALIGRAMAM		
43	VIRUGAMBAKKAM		
44	KOYAMBEDU		
45	ICE HOUSE		
46	VEPERI		
47	KELLYS		
48	PERIYAR NAGAR		
49	AMINJIKARAI		
50	THIRUMANGALAM		
51	ARUMBAKKAM		
52	VILLIVAKKAM		
53	IYANAVARAM		
54	ICF		
55	MOGAPAIR		
56	PADI		
57	MADHAVARAM		
58	WASHERMENPET		
59	ROYAPURAM		
60	THIRUVETRIYUR	SHENOY NAGAR	

3.3.2.2 Demand distribution

The historical data can be used to generate a trip table matrix between the zones. In our study the data for the month of January 2006 has been used to generate the trip tables for the month. This trip table matrix will clearly indicate the most used origin – destination pairs (O – D pairs) and in general the most demanded traffic patterns The data from these matrices is used to calculate the probabilities of a customer call from various zones i.e., the probability of these zones being the pick-up and also the probability of a zone being the drop-off point.

The table 3.2 below shows one sample such distribution from historical data (for the month of January, 2006). It shows the number of calls as function of call time and pick-up time between the zones, Airport and Adyar calculated for the full month of January, 2006.

The trip table matrices between all O – D pairs for the full month of January 2006, has been included in the appendix section.

Desire lines

Desire lines for the trips have been plotted on the map between the zones. This gives a very vivid indication of the travel patterns of the call taxis in the city. These desire liens give valuable information regarding the patterns of the traffic demand and the desire patterns of the callers. It can be observed from the se lies that most of the trips are concentrated around the airport and the central railway station regions. These desire lines give a strong visual indication of the travel desires of the calls inside the Chennai city. The figure below shows the desire lines for the call taxis in Chennai. The thickness of the lien is directly proportional to the number of trips between the zones.

This type of modeling is very important to get a clear picture of the travel desire patterns. This will help in and devising strategies for the operations accordingly.

Table 3.2 Trip table between Adyar and Airport zones for Jan, 2006

Call time	Pick-up time ➔					
	0 am to 4 am	4 am to 8 am	8 am to 12 pm	12 pm to 4 pm	4 pm to 8 pm	8 pm to 12 am
0 am to 4 am	7	1	0	0	0	0
4 am to 8 am	0	1	2	1	0	0
8 am to 12 pm	0	0	6	5	0	0
12 pm to 4 pm	0	0	0	5	7	0
4 pm to 8 pm	0	0	0	0	6	5
8 pm to 12 am	0	0	0	0	0	34

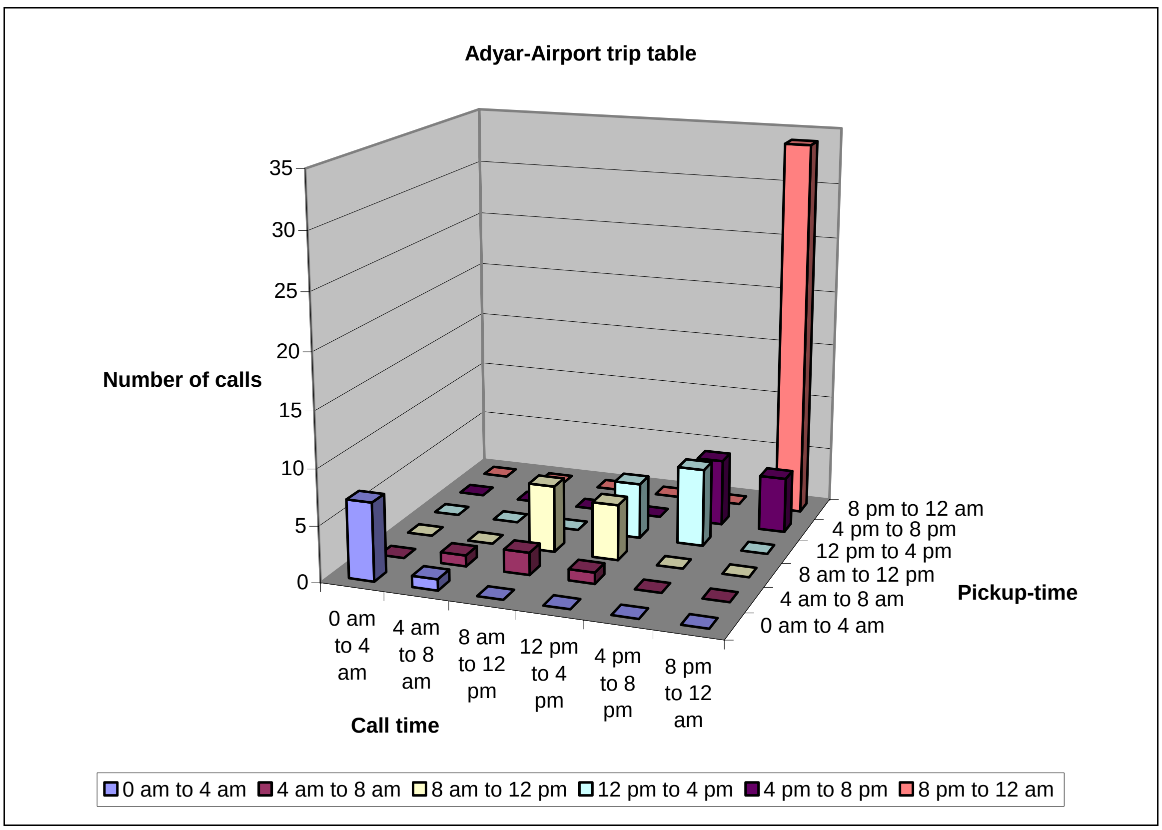

Figure 3.3 Histogram of the trip table between Adyar and Airport

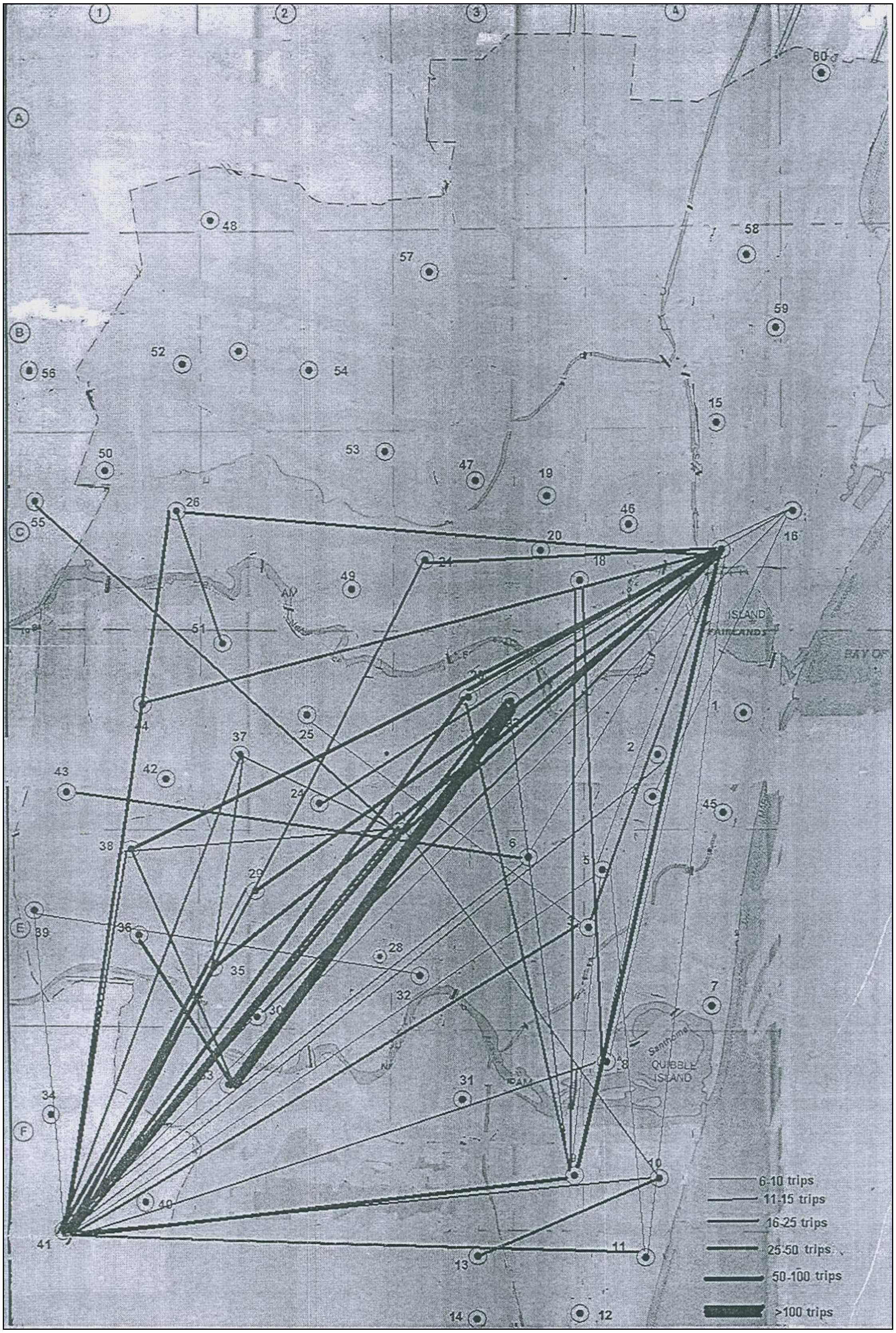

Figure 3.4 Desire lines between different zones in the map

3.3.2.3 Strategies adopted

This is the crucial element which is responsible for the efficiency of the system. Several approaches and models as explained in the literature review can be incorporated in this module. Re-optimization techniques and heuristics for allocation and dispatching, route guidance after the drop-off models and driver behavior models form the core of this module. These have to be carefully designed and adopted for efficient operations of the call-taxis.

3.3.2.4 Tariff policy

The following policy is used in practice by all the operators in Chennai. It is subjected to changes as a function of the fuel cost.

The customer pays

- For the first 3 km a fixed charge, Rs.50
- For every additional km, Rs.10

The driver gets

- Driver gets Rs.10 for every Rs.100 revenue generated (10 % commission)

Operational costs incurred by the operator

- Gas (Fuel-LPG cylinders) cost – Rs.25/ liter which amounts to Rs.1.75/km

An equivalent policy is used in the simulation models:

The customer pays

- Rs. 12 for every km he travels

Operator incurs

- Rs. 4 for every km the taxi runs (Fuel + Driver cost)

This tariff structure varies locally and according to the level of service provided also (A/c and Non-A/c, type of car) in more complicated scenarios. But, the above tariff structure assumed and used in the models is a very simplified version of the actual case. This has been chosen and applied to all the models uniformly.

3.3.2.5 Performance measures

The performance measures chosen and its modeling has been clearly explained in the chapter 1.3.2.2.

3.4 DATA EXTRACTION PROCESS

The data provided by the call taxi firm for each request contained the following fields: Request ID, booking date, request date, call time, pick-up time, origin, destination, allotment time, dispatch time, taxi idle time, vehicle ID (allotted to that request), start meter, end meter, revenue generated.

3.4.1 Description of raw data

On an average each day had around 120 requests. Each request was given a request ID, the request origins and the destinations were the nearest land marks. As mentioned earlier, Chennai has been divided into 60 zones (each zone is given a zone index) for the purpose of simulation. So, a significant amount of time was spent to categorize these landmarks into their respective zones.

The first of the seven fields mentioned above are given by the caller making a request and the remaining fields are filled in as the process of allocation, dispatching and fulfilling of service occurs.

Details of a sample request are shown in the table below:

Table 3.3 Raw data structure

Request ID	AU11012310002
Booking date	1/1/2006
Request date	1/1/2006
Call time	14:23
Pick-up time	15:00
Origin	TNagar
Destination	Anna Nagar
Allotment time	14:32
Dispatch time	14:51
Taxi idle time	13:14
Vehicle ID	104
Start meter	17017
End meter	17037
Revenue generated	220

3.4.2 Format required by the models

The fields call time, pick-up time, origin and destination are only required for the purpose of simulation. An algorithm in Java was coded to cull the remaining fields, allocate appropriate zone IDs to the origin and destination fields, and convert the call and pick-up time into a proper format required by the simulation. This code neatly generates text files for each day containing these fields which can be readily used as inputs to the codes which run the simulation.

3.4.3 Data refining process

The data refining process can be better understood using the flowchart shown in the figure below:

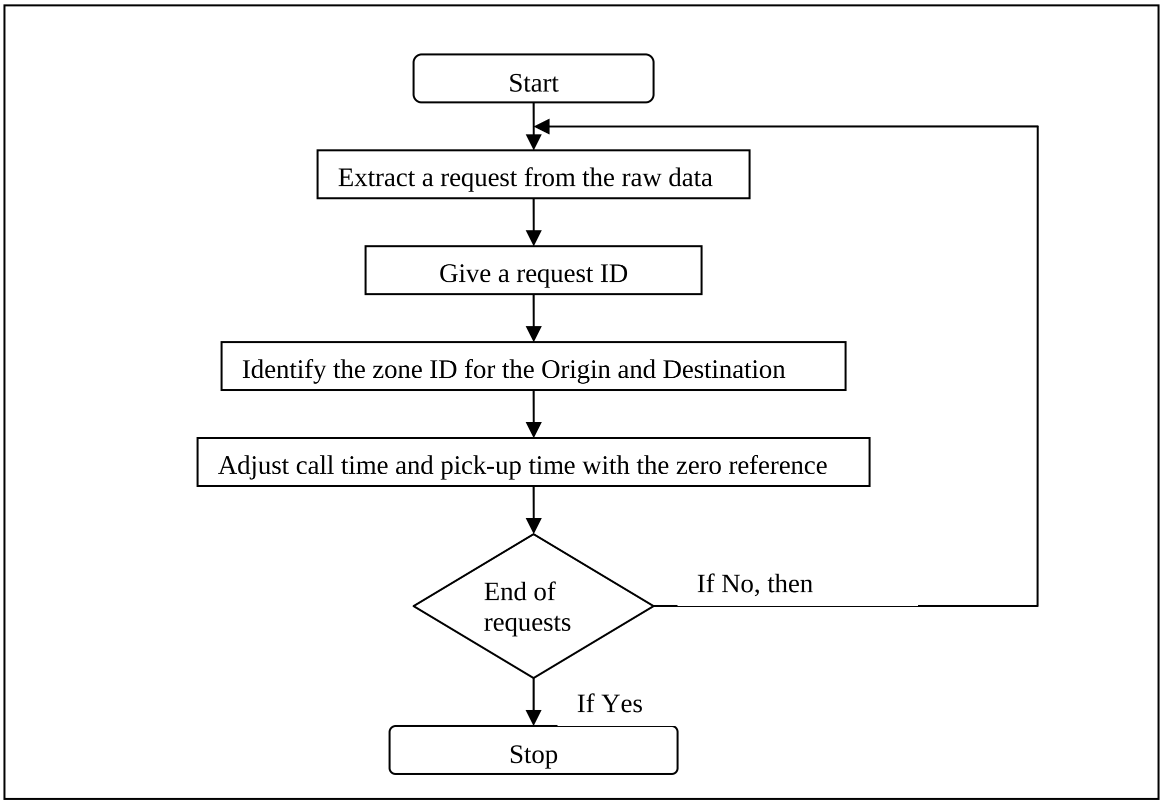

Figure 3.5 Flowchart – Data refining process

This is how the refined data request would look like:

Table 3.4 Refined data structure

Request ID	1
Call time	863
Pick-up time	900
Origin ID	26
Destination ID	25

This is given as an input to the simulation codes:

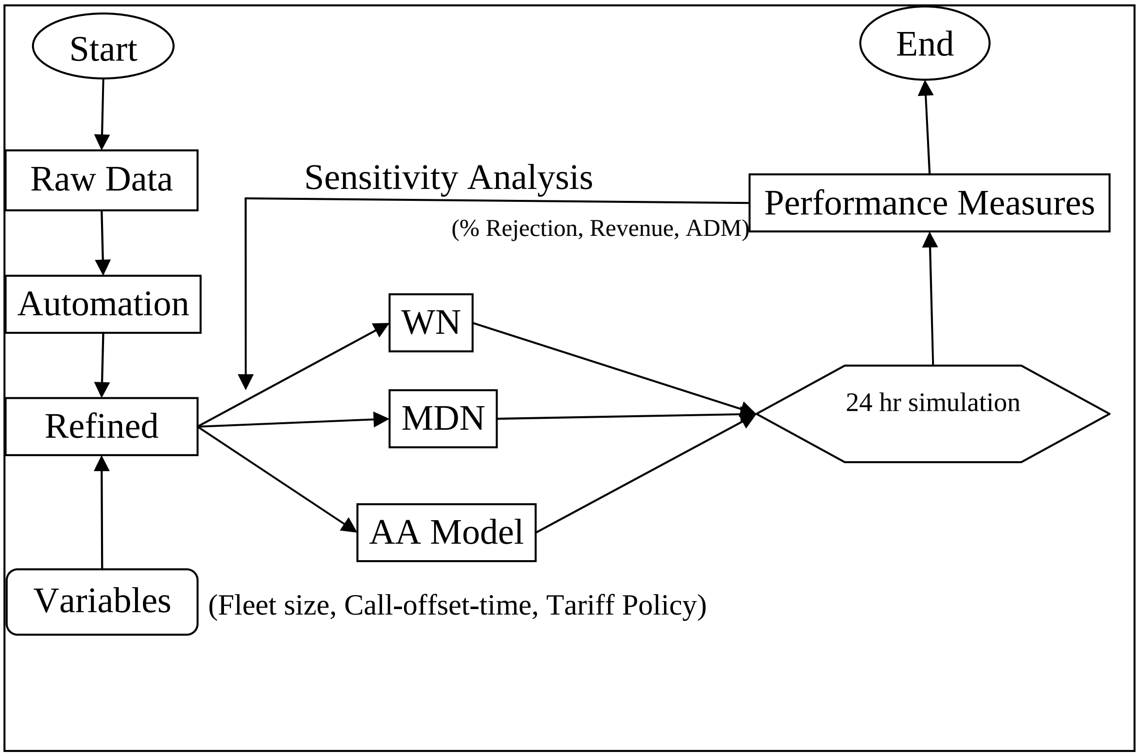

Figure 3.6 Flowchart- Simulation structure

WN (wait-at-node), MDN (move to desirable node) and AA (apriori approach) are the different models which will are developed and tested. They will be described in detail in the following chapters.

This raw data was also used to generate trip tables (number of trips requested for all the O-D pairs in the network) to identify important nodes where the maximum activity occurs. This information was used to calculate the desirability of nodes which will be explained in detail in chapter 5. The trip table for the all the O-D pairs for the month of January 2006 is listed in the appendix as mentioned.

3.5 SUMMARY

Thus, this process laid the foundation to develop and improve the methods and strategies used in practice. This exercise helped understand the concepts, constraints and mainly get a feel of the problem. Finally, data after the refining process was used in the models and appropriate results were generated which will be summarized in the following chapters.

CHAPTER 4

APRIORI MODEL

4.1 INTRODUCTION

In order to find the best possible global optimal solution to the routing and the scheduling problem at hand, one has to have the complete information of the request set apriori (in advance). The usage of the word apriori in this context has been explained in the problem context (Section 1.3.1). In reality this is normally not the case. The caller places a request for the service only few hours before his intended pick-up time. So a global optimization technique cannot be adopted. At the end of the day the system operator will have the knowledge of the request set for the day. This set is used to calculate the global optimal solution and is compared with the solution used in practice to measure the gap between them. So, in essence we are solving a hind-sight problem than an apriori problem. The apriori (hind-sight) problem can be formulated as a minimum cost network problem.

4.1.1 DEFINITION OF TERMS

Two types of networks are typically used in this formulation. One is the ‘Physical Network’ and the other, the ‘Decision Network’. A typical transit network is the union of nodes and links. In the context of the physical network, a node is a place where demand might originate or terminate, in other words nodes are the passenger’s origin and destination points. One of the nodes in the network is a source node or hub node. This is the node where the fleet starts at the beginning and return at the end of each day. A new taxi is dispatched from the hub as and when there is need to satisfy the demand from a passenger arises. A link is physical connection (road) between two nodes; the length of the link is proportional to the distance or the travel time between the nodes connected by the link. Computer representation of the transit network is done using a node-node adjacency matrix. This matrix contains the link travel distances if a link exists between a given pair of nodes and -1 otherwise, as illustrated in Table 4.1. In the network shown below, node 1 is the hub node.

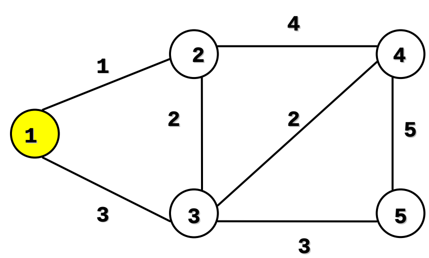

Figure 4.1 Example network

Table 4.1 Adjacency Matrix

	1	2	3	4	5
1	0	4.5	9	-1	-1
2	4.5	0	7.5	12	-1
3	9	7.5	0	6	10.5
4	-1	12	6	0	15
5	-1	-1	10.5	15	0

The network in Figure 4.1 can be represented using the node-node incidence matrix shown above. Link travel times serve as an input to the model using the above adjacency matrix. Link travel times, in minutes, are input as travel-time matrix, which is shown below:

Table 4.2 Travel-time matrix

	1	2	3	4	5
1	0	13.5	27	-1	-1
2	13.5	0	22.5	36	-1
3	27	22.5	0	18	31.5
4	-1	36	18	0	45
5	-1	-1	31.5	45	0

The definition of the node holds well in the case of the Decision network too. In addition to the hub node, a sink node is also present in the Decision Network. All the taxis after servicing all its requests are directed to this node at the end of each day. In the decision network the links can be classified into four distinct varieties. They are the request links, inter-request connector links, the terminal-request connector links and the sink-source connector.

Each request call between a specific origin and a destination is modeled as a request link. The inter- request connector links are added to the decision network based on the feasibility of the calls i.e. if a request can be satisfied and the same taxi can satisfy another subsequent request at a later time and at a different node, connectivity is established between origins of such request links using a inter-request connector link. The terminal-request connectors connect the source (hub) and the sink nodes to the origins and the destinations of the appropriate request links respectively. Finally the sink is connected to the source by a link called the sink-source connector link. So based on a set of the requests given apriori, the decision network on which apriori formulation is to be performed is constructed.

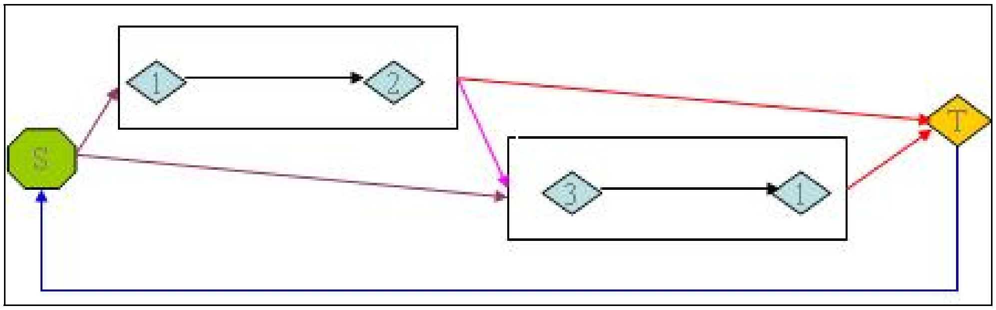

Figure 4.2 Example Decision network

In the above example decision network ‘S’ is the source and ‘T’ is the sink, the links in the boxes are the request links, the links in the pink color are the inter-request connector links, the one in the brown and red color are the terminal-request links and the one in the blue is the sink-source connector link.

4.2 ASSUMPTIONS

The following assumptions are made in the apriori formulation:

1. The requests which arise on a particular day are known apriori, so that an optimal solution to cater to that request set can be sought for through this apriori model.
2. The travel times on the links i.e. between any two nodes is assumed to be constant and determinate. The value of the travel time between any two nodes is obtained dividing the corresponding distance with an assumed constant average speed of 20 kmph for the taxi.
3. The passenger pickup and the drop-off occur only at the nodes and the pick-up and the drop-off happens instantaneously with out any delay at the corresponding pick-up and the drop-off times.

4. The model is executed on a daily-basis, in other words requests for a service on every day are pooled together and an optimal solution is sought for it.
5. A single taxi can service only one request at any point of time. No two requests can be simultaneously satisfied by the taxi (No pooling of requests).
6. Enough fleet is available at the source node to cater to all requests without delay. Finally the fleet size actually used will give the optimal value of the fleet size required to satisfy all requests.
7. Another crucial assumption that a no congestion scenario has been assumed
8. It is also assumed that calls once booked are not cancelled.

4.3 OBJECTIVES & CONSTRAINTS

The objective is to maximize the revenue/profit generated from the services provided. This can be achieved by optimal allocation of the fleet to the requests. Three main constraints are to be considered, they are the time and the capacity constraint.

4.3.1 Time constraint

This constraint ensures that the taxi reaches on or before the pick-up time requested by the caller while placing a request. Time-windows are not allowed, meaning the taxi cannot reach the origin of the caller after the pick-up time requested. This constraint determines whether interconnections are feasible between various requests in the decision network.

4.3.2 Capacity constraint

The capacity constraint of one for each taxi ensures that only one request is been serviced at any particular time.

4.3.3 Conservation of flow constraint

The number of taxi entering a particular node must equal to the number of taxi leaving the node. This constraint ensures the conservation of the taxis i.e. avoids duplication of the taxi and it maintains the fact that the total number of taxis available for operation is limited.

4.4 MODEL FORMULATION

The apriori problem is formulated as a network minimum cost circulatory problem. First the decision network is built based on the inputs i.e. from the call request set (origin, destination, pick-up time of the request) and the physical network based on the feasibility of serving a sequence of requests.

Inputs required:

1. The apriori known request set for which the optimal solution is sought.
2. Time period of operation (period over which the services are offered)
3. Physical network (Distance and travel time matrices)

Input Variables:

T = Time period of operation

$\mathbf{L}_{ij}$ = Lower bound on the arc $\mathbf{A}_{ij}$

$$L_{ij} = \begin{Bmatrix} 1 \ \forall \ requestlink \ \ Aij \\ 0 \ \forall \ non-requestlink \ \ Aij \end{Bmatrix}$$

$\mathbf{U}_{ij}$ = Upper bound on the arc $\mathbf{A}_{ij}$

$$U_{ij} = \begin{Bmatrix} 1 \ \forall \ requestlink \ \ Aij \\ \infty \ \forall \ non-requestlink \ \ Aij \end{Bmatrix}$$

$\mathbf{D}_{ij}$ = Travel distance on the arc $\mathbf{A}_{ij}$

$\mathbf{C}_{ij}$ = Cost incurred on traversing the arc $\mathbf{A}_{ij}$ (Travel time)

$$C_{ij} = \begin{Bmatrix} Dij \times CRij \,/\, km \ \ \forall \ \ requestlink \ \ Aij \\ Dij \times CNRij \,/\, km \ \ \forall \ \ non-requestlink \ \ Aij \end{Bmatrix}$$

$$CR_{ij} = \ O - R$$

$$CNR_{ij} = O$$

$O =$ Operational cost per km of distance travelled

$R =$ Revenue generated per km of distance served

$\mathbf{T}_{ij}$ = Time taken to traverse the arc $\mathbf{A}_{ij}$ (Travel time corresponding to the shortest)

Decision Variables:

$\mathbf{F}_{ij}$ = Flow on the arc $\mathbf{A}_{ij}$

N = Fleet size required to serve all the requests optimally

4.4.1 Objective function:

The objective is to minimize the cost incurred in satisfying the entire request on any particular day. The cost C_{ij} is negative on the request links (where the revenue is generated) and positive on every other links (operational costs)

We, thus, formulate a minimum cost network formulation for the problem:

Network **G = (N, A)** where **N** = {n_0; n_1; : : : ; n_n; n_{n+1} } is the node set and **A** = {(A_{ij}); A_{ij} Є A, i ≠ j }is the arc set. Vertex n_0 and n_1 represents a source and the sink nodes respectively, and the remaining **n** nodes represent origins and destinations for the transportation requests. L_{ij}, U_{ij} and C_{ij} are the lower bound, upper bound and the cost of traveling on the link $\mathbf{A_{ij}}$. $\mathbf{F}_{ij}$ be the flows on the links $\mathbf{A_{ij}}$.

Objective function: Minimize $\sum \mathbf{C}_{ij} * \mathbf{F}_{ij}$ for every $\mathbf{A}_{ij}$ Є A

$$L_{ij} \leq \mathbf{F}_{ij} \leq U_{ij} \text{ \{Lower and upper bound\} } (4.1)$$

$$\sum_j \mathbf{F}_{ij} = \sum_i \mathbf{F}_{ji} \text{ \{Flow conservation constraint\} } (4.2)$$

4.4.2 Constraints of the formulation:

The above constraints ensure that all the requests are satisfied while maintaining the conservation of the flow (taxis). The request links have a lower and upper bound of one, this ensure that the request is satisfied by allowing a flow of one on this particular type of links. All other links have a lower bound of zero and an upper bound of infinity. Since this problem is formulated as a circulatory problem, it ensures that the taxi return to the source node at the end of the day.

4.5 SOLUTION METHODOLOGY

The above global optimization technique was implemented in JAVA using the algorithm given below:

Global Optimization Algorithm

Step 1: Get the inputs namely request set after appropriate refining from the raw data. The refining process includes matching and allocation of zone ids for the request origins and destinations.

Step 2: Build the decision network

For each request, check the feasibility for servicing a subsequent request. If yes make a connection (arc), else move to the next request

Step 3: For each arc, assign appropriate lower bound, upper bound, arc cost, travel time values.

Step 4: The objective function and the constraints are generated in the format as required by LP solver, lp_solve version 5.1 and are given as input to the solver

Step 5: Solve the problem using the solver and get the outputs

The optimal number of taxis required and the sequence of request each taxi should service to get the system optimal solution is the output.

4.6 ILLUSTRATION

This section an illustration will be presented. Consider the physical network given in the figure 4.1, the adjacency matrix corresponding to this network is given in table 4.1. Consider a small request set as given in the table 4.3 below for which an optimal scheduling of taxis is required.

Table 4.3 Sample Request Matrix

S .no	Origin node	Destination node	Call time	Pick-up time
1	1	3	45	100
2	4	5	50	110
3	5	2	150	210

The corresponding decision matrix will be as given in the figure 4.3. ‘S’ is the hub node, ‘T’ is the sink node and the rectangular boxes are the requests (between origin and destination) represented as link. The feasibility of serving subsequent requests leads to several inter connections between such requests as shown. The other link components such as arc cost and travel times of the table 4.4 are imposed on this layer of the network.

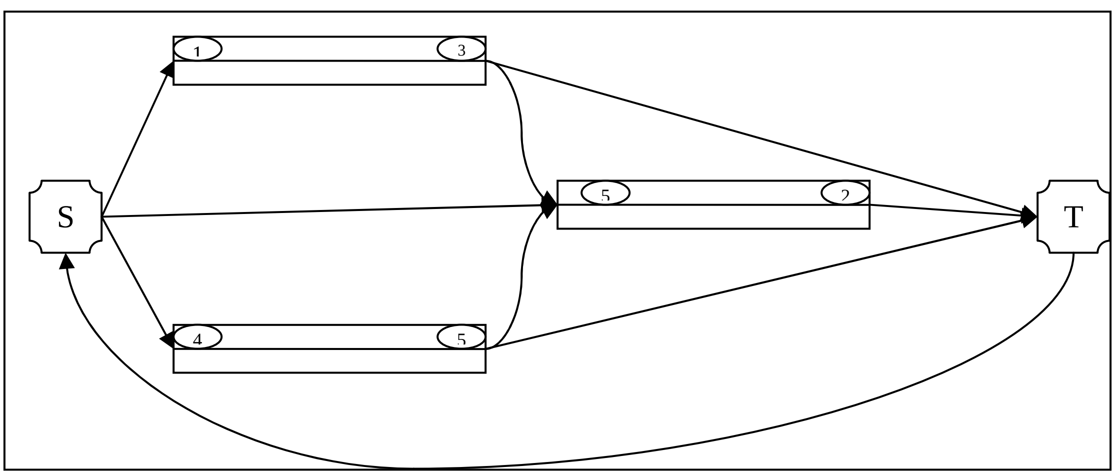

Figure 4.3 Decision network

Input parameters:

Table 4.4 Input parameters

Parameter	Value
Revenue generating path	Rs. 12/km (negative)
Dead mileage path	Rs. 4/km
Travel distance on each link	Corresponding shortest path
Travel times on each link	Travel times corresponding to the shortest path

For this super-imposed decision network, the minimum cost solution is obtained with the help of the simulation code embedded with the LP solver. The formulation investigates all the feasible solutions (three feasible solutions exist in this example). In the first feasible solution taxi 1 serves request 1 & 3 and taxi 2 serves request 2. In the second feasible solution taxi 1 serves request 1 and taxi 2 serves request 2 & 3. In the third feasible solution taxi 1 serves request 1, taxi 2 serves request 2 and taxi 3 serves request 3. The second solution was found to be the optimum solution.

The optimal solution is given below:

Trip allocation:

Table 4.5 Trip allocation for optimal solution

Taxi ID	Sequence of trips for optimal solution
1	Request 1
2	Request 2, 3

Performance measures:

Table 4.6 Performance measures

Parameter	Value
Revenue	Rs. 420
Fleet size required	2 taxis
Total vehicle kms traveled	63
Dead mileage/trip	7 km

4.7 CONCLUSION

This type of a global optimization method results in giving a measure of optimal fleet requirement, maximum revenue that can be generated, measure of taxi idle time and dead mileage incurred. One can then estimate of gap between the global optimal solution and the heuristic solution used in practice. The next chapter discusses the heuristics solutions.

CHAPTER 5

HEURISTIC APPROACH

5.1 INTRODUCTION

In chapter 4, we have seen that an optimal solution to the routing and scheduling problem of the taxi can be solved provided the request set is known apriori, but in the real is impossible to have the request set apriori and therefore have the optimal scheduling strategy. In situations like these, normally one resorts to simple and advanced heuristics for the operations. This chapter will introduce you to some such heuristics which can be used in practice to improve the efficiency of the system.

5.2 PROPOSED HEURISTICS

5.2.1Introduction

Two heuristics are being proposed. The first one is called ‘The wait at the node’ heuristic (WN) and the second heuristic is called ‘Move to the desirable node’ heuristic (MDN). The system can be operated by adopting any one of the heuristics. In order to get a quantitative measure of the goodness of the heuristic one should run parallel simulations and the performance measures will indicate the efficiency of the system when a certain heuristic is adopted. So to quantify the goodness, the performance measures described in the chapter 1 have been modeled and used. The following section of this chapter will describe the heuristics in detail.

5.2.2 Logic behind using the heuristics

The only thing under the control of the operator is what the taxis do when they are not attending engaged in providing the service. In other words by judicious decision making of which taxi to allocate to a particular request and what that particular taxi should do when passenger is dropped off is off primary importance in making the system efficient. So it is clear that the following two crucial decision epoch exists where heuristic defines the action to be taken at these epochs.

- Which free taxi should be allocated to the request
- What does the taxi do after the drop-off

5.2.3 Wait at the node heuristic (WN)

In this heuristic to address the first decision epoch the heuristic directs the operator to allocate the nearest available free taxi to the request and the taxi remains station at the drop-off node until it gets allocated to a sub-sequent request. This is the most basic heuristic that can be adopted.

5.2.4 Move to the desirable node heuristic (MDN)

The directions for the first decision epoch remain the same, but for the second decision epoch the taxi moves to the most desirable node after the drop-off. For the successful implementation of this heuristic the desirability of all the nodes must be calculated.

The desirability of a node is both a function of the number of calls that get initiated at this node on an average per day in a month and the distance that has to be traversed to reach this node. While the number of calls factor is a positive effect implying that it contributes to the desirability of that node, the distance that has to be traversed is a negative one. Different weights were given to these factors to get multiple sets of desirability values. Each set was tested using a simulation to find out the set which gives the closest possible solution to the apriori optimum. The trip tables generated in the data extraction process was used in the process of calculating desirability.

5.3METHODOLOGY & IMPLEMENTATION OF THE SIMULATION

5.3.1 Procedure

The following step-by-step process is followed:

Step1: The passenger sends a request to the operator. The request will include details of the pick-up place (origin), drop-off place (destination), and pick-up time.

Step 2: Exactly at a time equal to call-offset time (usually 30 min) this request is processed. A feasibility check (fleet size) as to whether a taxi can be allocated to this request at this particular time is performed.

Step 3: If yes, then the closest free taxi available to the origin of this request is dispatched to the appropriate pick-up place of this particular request. If no, then the passenger is notified that his request cannot be fulfilled due to insufficient fleet size.

Step 4: After the taxi drops off the passenger at his destination, it uses the heuristic to govern its further course. If WN heuristic is used, the taxi waits at this particular node until it gets allocated to a sub-sequent request. If MDN heuristic is used, the taxi moves to the best (desirable) node closest to this node and waits at this node for a sub-sequent request.

Step 5: All the taxi return to the source node at the end of the day.

Step 6: Performance measures are calculated at the end of the day like the revenue generated, car-waiting time of the taxi, % of calls rejected, dead-mileage incurred in the process.

5.4 ILLUSTRATION

Inputs:

Consider the physical network given in the figure 4.1 and its corresponding adjacency, travel time and request matrices given in the tables 4.1, 4.2 and 4.3.

Call – off set time: 60 minutes , Fleet size: 2 taxis

The WN strategy is illustrated here:

Performance measures (output):

Table 5.1 Performance measures of WN strategy

Parameter	Value
Revenue	Rs. 420
Average dead mileage (per trip)	7 km
Number of calls late serviced	1
Average car waiting time	24 min

Simulation events:

Table 5.2 List of events in the simulation

Time Stamp	Event
0	Start simulation
40	Taxi 1 allocated to request 1
40	Taxi 1 reaches origin node – 1 of request 1
50	Taxi 2 allocated to request 2
100	Taxi 2 reaches origin node – 4 of request 2
100	Taxi 1 starts serving request 1- Taxi marked busy
111	Taxi 2 starts serving request 2- Taxi marked busy
127	Taxi 1 reaches destination node – 3of request 1 – Taxi marked free
145	Taxi 2 reaches destination node – 5 of request 2 – Taxi marked free
160	Taxi 2 allocated to request 3
219	Taxi 2 reaches origin node – 2 of request 3
219	Taxi 2 starts serving request 3- Taxi marked busy
273	Taxi 2 reaches destination node – 2 of request 3– Taxi marked free
273	All request served – All taxis move towards the hub node -1
286	Taxi 2 reaches hub node -1
300	Taxi 1 reaches hub node -1
300	End of simulation

It can be observed that the performance of the Apriori and the WN models are the same. But, significant difference can be seen as the number of request increases. The results for the real time data for the month of January 2006 are tabulated and discussed in the next chapter.

5.5 CONCLUSION

In the real world when the demands are not known apriori, one has to inevitable resort to the use of heuristics to manage the operations. The efficiency of the system depends on the goodness of the heuristics. The above described heuristics are some such examples. The simulation has shown that they provide a very good solution to the problem, which is not very far from the optimal solution generated by the apriori model in terms of the revenue generated and other such performance measures.

CHAPTER 6

EXPERIMENTAL DESIGN AND PROCEDURES

6.1 INTRODUCTION

In this chapter we run and analyze the results of a thirty day data set (Jan 1st, 2006 to Jan 31st, 2006) on the three different models explained in the previous chapters namely the apriori, wait-at-node and move-to-desirable-node models. A thorough sensitivity analysis across the various models to various factors and their influence on relevant performance measures is presented.

6.2 FACTORS CHOSEN AND THEIR EFFECT ON PERFORMANCE OF THE SYSTEM

The important factors that affect performance of the system are:

1. Demand size
2. Fleet size
3. Call-offset time

6.2.1 Demand size

It was observed that on an average, the operator receives 168 requests each day. The minimum recorded was 83 calls and maximum was 211 calls. The distribution pattern across the each month using historical data can be studied to appropriately hire and maintain fleet size at the hub node. This will result in a huge savings in terms of reduced operational costs. The pattern observed for the month of January, 2006 is as shown in the figure below.

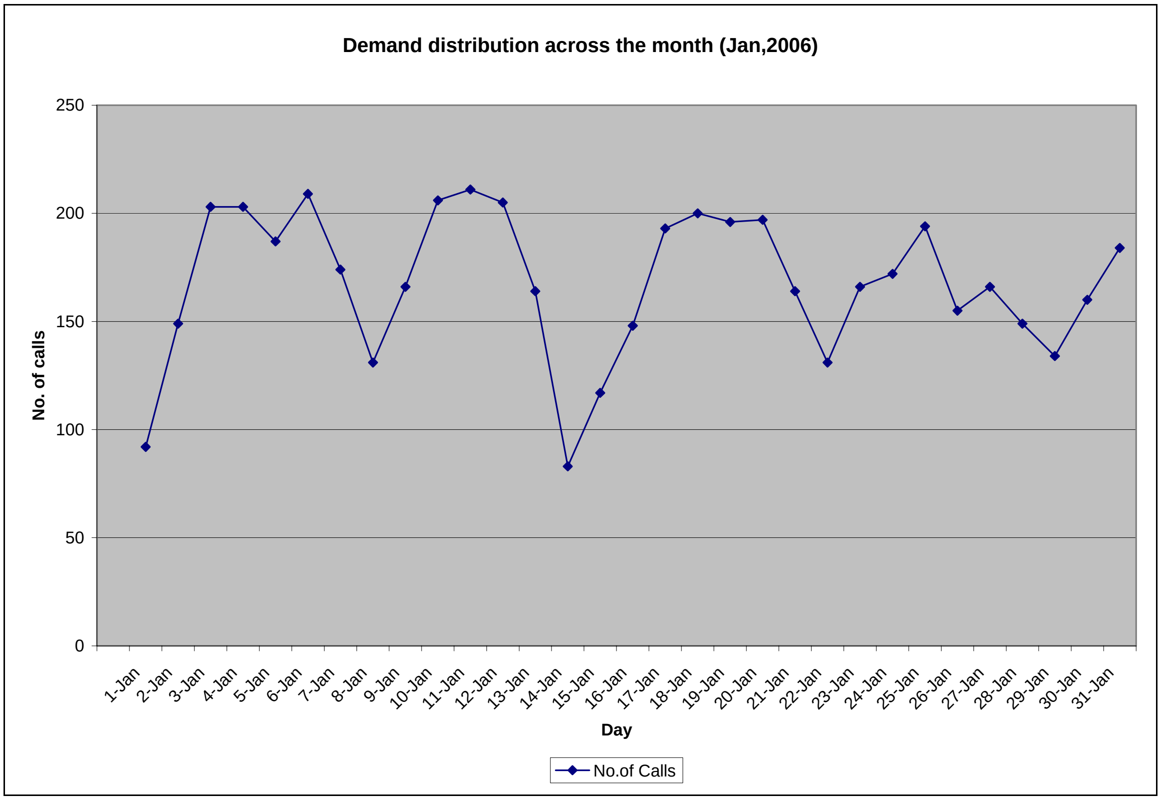

Figure 6.1 Demand-distribution across the month

Inferences:

It can be observed that the demand rises steadily in the initial days and remains steady for the next 10 days and falls steeply when it reaches the middle of the month and later it continues to fall towards the end of the month. This can be to a certain extent attributed to the cash-flows of the individuals. The affordability for the services increases at the beginning of the month because usually the salaries pour in and the individual is rich in cash during this period. Later, there is a fall in the demand and it reaches the minimum due to increased scarcity of finances during the mid-month period and also this was holiday time (Pongal vacation). Later on, the average demand slightly increases but falls again with a much lesser gradient towards the end of the month. This slight increase in the demand may be attributed to the increased activity of the individuals to commute as it reaches the end of the month. However, patterns for atleast another 10 to 15 months need to be observed to validate this theory. This was not possible due to the constraint of having obtained data for a single month. Several other variables like nationals holidays, important exam days also affect this pattern to a great extent.

The sensitivity of this demand variation to various performance measures (revenue generated and fleet size required) is analyzed in the further sections.

6.2.2 Fleet size

The operator can plan to lease fleet according to a pattern to carry his operations depending on the variation of the demand size expected for a particular month. So a good analysis of the historical data will give him on an average, the fleet size distribution he would require to run the operations. This will result in maintaining the minimum fleet and hence run the operations with the least possible capital investment and operational costs.

6.2.3 Call offset time

Call offset time should be carefully chosen. A higher call offset time will result in an unnecessary car waiting times at the origin of the requests and a lower call-offset time will result in a delay in the pick-up of the passenger at the origin, since the low call-offset time might not be sufficient to send the taxi to the origin from another node where the taxi is currently located at that particular point of time.

6.3 SIGNIFICANCE OF THE VARIOUS PERFORMANCE MEASURES AND THE VARIOUS TRADE – OFFS BETWEEN THEM

Performance measures quantify the efficiency of the system. The various performance measure chosen for this purpose are:

1) Revenue generated
2) Average dead mileage
3) % of calls rejected due to fleet constraint
4) % Calls late serviced
5) Taxi waiting time

The definition and computation procedure of the above performance measures has been explained in chapter 5. Here the significance and the various trade-offs that exists between them are elaborated.

6.3.1 Revenue generated

The operator strives for generating the maximum revenue while maintaining a good level of service (minimum % of calls rejected and minimum % of calls late serviced). Higher revenue generated for a given set of calls indicates that a good strategy has been adopted. Revenue is directly proportional to the goodness of the strategy adopted and the efficient allocation of the taxis to the requests. More revenue can be generated by pricing the services at an optimal rate due to increased demand and further increase the size of operations allowing economies of scale in the process.

6.3.2 Average dead mileage

Average dead mileage as explained is an indicator of the empty run of the taxis. If the dead mileage incurred due to adopting a particular strategy is very high then the operations based on this strategy will not be economically feasible. It is a function of the strategy adopted for example in the MDN model there is an additional component in the dead mileage incurred vis-à-vis moving to the desirable node after the drop off. This might contribute to the loss of revenue due to stretch of empty run. But this increased dead mileage will have other benefits like easier and quicker allocations of the request. In other words % of calls late serviced will not increase significantly with decrease in call-offset time. So there exists a subtle trade-off between dead mileage and ease and efficient allocation of taxis to the requests.

6.3.3 % Calls rejected due to fleet constraint

This contributes largely to the level of service from the view point of the passenger requesting a service. This becomes a critical issue only in the peak hour. If the operator chooses to have a large fleet size to maintain a zero percentage of calls rejected, he has to incur a huge capital as initial investment cost and moreover a very large fraction of the taxis will remain idle during the non-peak hours which will a huge loss to the operator. So there is a trade off between percentage of call rejected and the fleet size maintained. An optimal fleet size for an accepted level of percentage rejection is the best approach to be followed. The thresholds for it, however is entirely dependent on the operator.

6.3.4 % Calls late serviced

This is a function of the call-off set time, more the call-offset time earlier a taxi will be allocated to the request and can reach the passenger well before time but this has the disadvantage of resulting in large taxi idle times, which other can be used to serve other request. This again is an issue in during high demand peak hours. If the operator has very large fleet then the operator can afford to have a large call-offset time and hence very less or no calls late serviced.

6.3.5 Taxi idle times

This gives a qualitative estimate of the efficiency of allocation of the taxis to the request. Lesser the average taxi idle time better will be the performance and efficiency of the system. The allocation strategy should aim at minimizing the average taxi idle time of the system.

6.4 RESULTS & ANALYSIS – COMPARISON ACROSS VARIOUS MODELS

The results for the various models have been tabulated, analyzed and inferences are drawn in the subsequent sections. As mentioned earlier, call request data for the month of January 2006 has been collected and used for this analysis.

6.4.1 Comparison of revenue generated across various models and its inferences

Table 6.1 Model Vs Revenue generated

Date	No. of Calls	Revenue in Rupees (Closing)	
		Apriori	WN Model (Fl=25, CO=45)
1-Jan	92	4464	4296
2-Jan	149	7124	6876
3-Jan	203	9964	9480
4-Jan	203	9963	9468
5-Jan	187	9580	8772
6-Jan	209	11004	10572
7-Jan	174	8196	7724
8-Jan	131	6112	5760
9-Jan	166	8512	8032
10-Jan	206	10104	8908
11-Jan	211	12060	11368
12-Jan	205	10804	9688
13-Jan	164	8056	7460
14-Jan	83	3596	2872
15-Jan	117	5176	4848
16-Jan	148	6980	6174
17-Jan	193	9640	8940
18-Jan	200	10056	9480
19-Jan	196	9060	8652
20-Jan	197	9468	8564
21-Jan	164	7628	7280
22-Jan	131	5568	4760
23-Jan	166	7688	6932
24-Jan	172	8500	8404
25-Jan	194	9408	8820
26-Jan	155	7340	6988
27-Jan	166	8104	7624
28-Jan	149	7080	7000
29-Jan	134	6268	6124
30-Jan	160	8300	7400
31-Jan	184	8536	8200

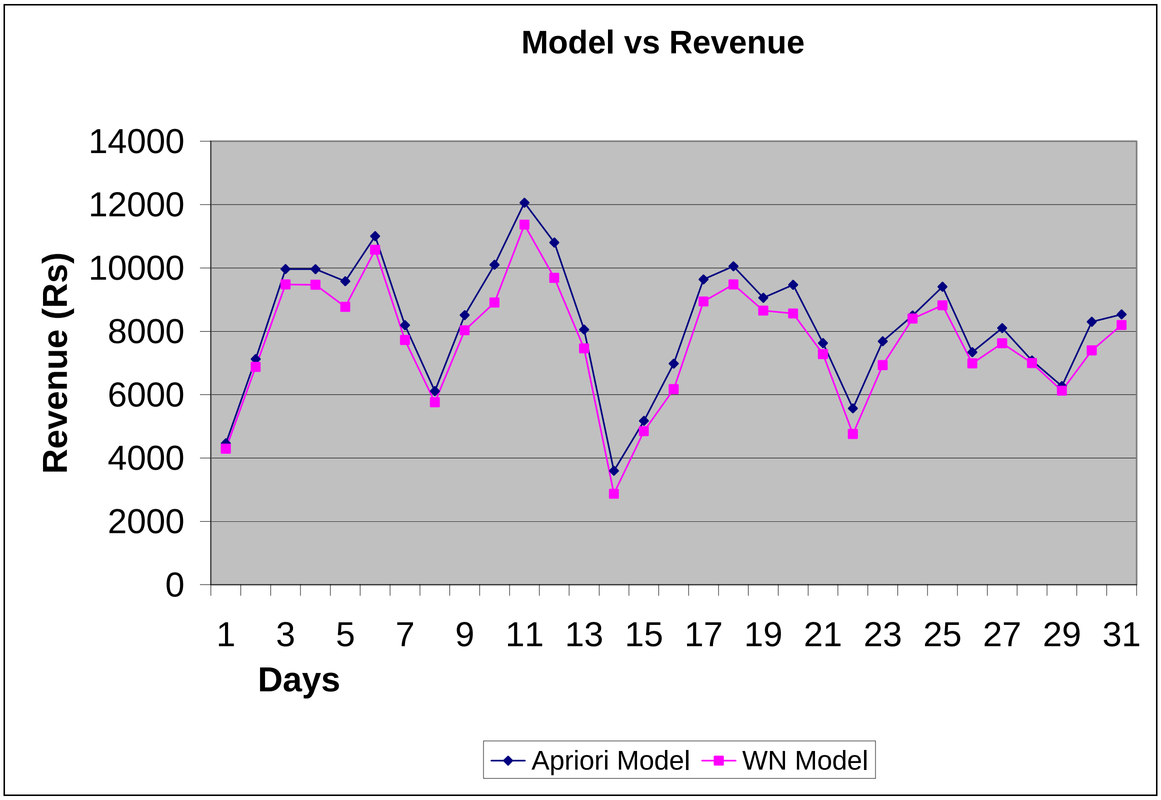

Figure 6.2 Model Vs Revenue generated

Inferences:

- The average gap between the apriori and the wait at node model is 8 % which amounts to a gap of Rs. 650 every day. This will amount to Rs.19000 per month. This is a significant gap and there is a scope for refining this heuristic to reduce the gap. But this heuristic is the easiest and the simplest in terms of adoption and implementation.
- It can be observed that on some days when the number of calls is above 180 the gaps is as low as 2 %, so this model is very close to optimum on these days. On days when the number of calls is relatively lesser, the gap between the revenues might be as high as 25%. So it can be concluded that if the demand is huge on any particular day this strategy can be adopted to give good results.

6.4.2 Comparison of fleet size requirement across various models and its inferences

Table 6.2 Model Vs Fleet requirement to manage the operations

Date	No. of Calls	Optimal Fleet Required	For 0 % rejection WN
1-Jan	92	13	15
2-Jan	149	20	25
3-Jan	203	24	25
4-Jan	203	23	25
5-Jan	187	24	25
6-Jan	209	27	25
7-Jan	174	20	20
8-Jan	131	17	20
9-Jan	166	23	25
10-Jan	206	27	25
11-Jan	211	24	20
12-Jan	205	24	20
13-Jan	164	23	20
14-Jan	83	13	10
15-Jan	117	13	15
16-Jan	148	20	20
17-Jan	193	24	25
18-Jan	200	22	20
19-Jan	196	23	25
20-Jan	197	21	20
21-Jan	164	22	15
22-Jan	131	17	15
23-Jan	166	22	20
24-Jan	172	23	25
25-Jan	194	21	25
26-Jan	155	19	20
27-Jan	166	23	20
28-Jan	149	23	15
29-Jan	134	16	20
30-Jan	160	20	25
31-Jan	184	22	25

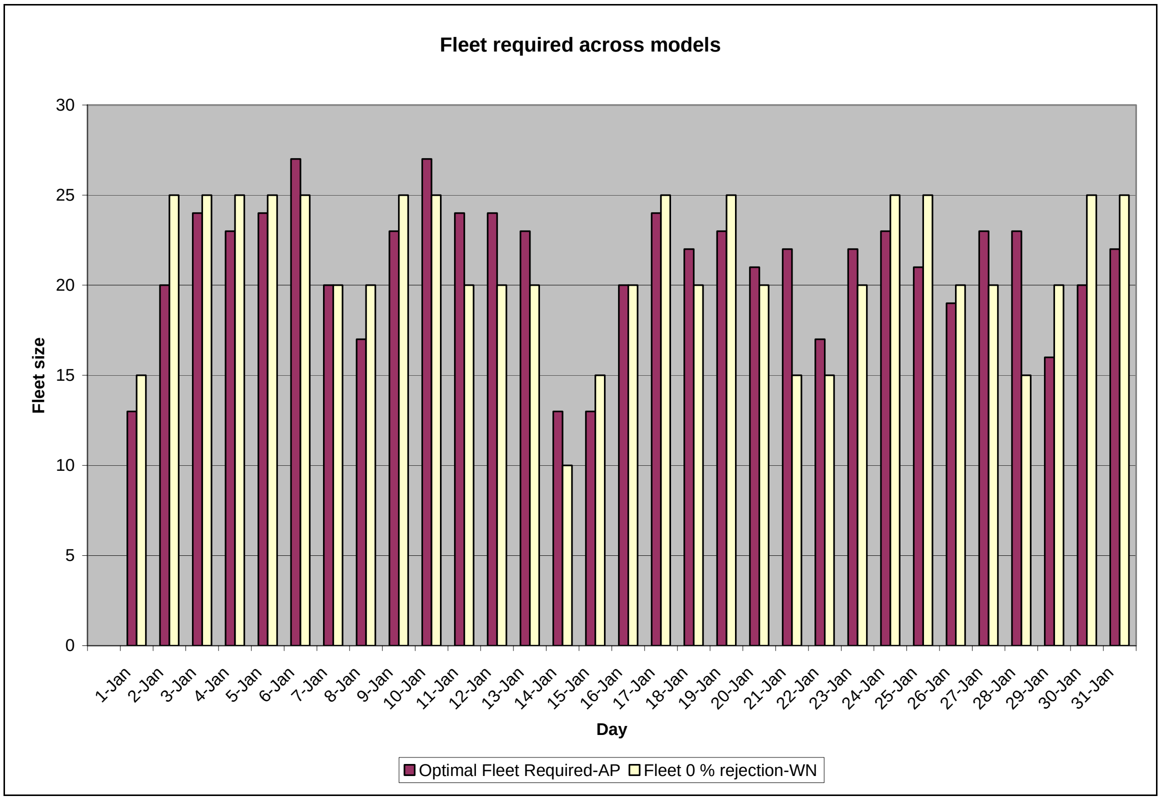

Figure 6.3 Model Vs Fleet requirement to manage the operations

Inferences:

- It can be observed that the optimal fleet size required in the apriori model is lesser than that required for other models to maintain a zero percent of rejection of calls due to fleet constraint.
- Sometimes the optimal fleet size is more in the case of apriori model for providing the services with the minimum cost because of the assumption that the capital investment in starting the services with a certain fixed fleet is a sunk cost.
- The minimum and the maximum fleet required in the month ideally (apriori model) is 13 and 27 and for the WN model is 15 and 25 respectively. So based on this information the operator can maintain a fleet size based on his initial capital

constraint and based on a marginal advantage of adding an extra taxi to his fleet (additional number of days with a zero % rejection).

6.4.3 Comparison of Average dead mileage across various models and its inferences

Table 6.3 Model Vs ADM

Date	No. of Calls	ADM/Trip - AP	WN-ADM/Trip for 0 % rejection
1-Jan	92	5	4
2-Jan	149	4	3
3-Jan	203	4	3
4-Jan	203	4	4
5-Jan	187	4	4
6-Jan	209	4	3
7-Jan	174	3	3
8-Jan	131	4	3
9-Jan	166	3	3
10-Jan	206	4	4
11-Jan	211	3	4
12-Jan	205	4	5
13-Jan	164	4	4
14-Jan	83	5	5
15-Jan	117	5	4
16-Jan	148	4	4
17-Jan	193	3	3
18-Jan	200	4	4
19-Jan	196	4	3
20-Jan	197	3	4
21-Jan	164	4	5
22-Jan	131	4	4
23-Jan	166	4	4
24-Jan	172	4	3
25-Jan	194	4	4
26-Jan	155	4	4
27-Jan	166	4	4
28-Jan	149	4	4
29-Jan	134	4	3
30-Jan	160	4	4
31-Jan	184	4	3

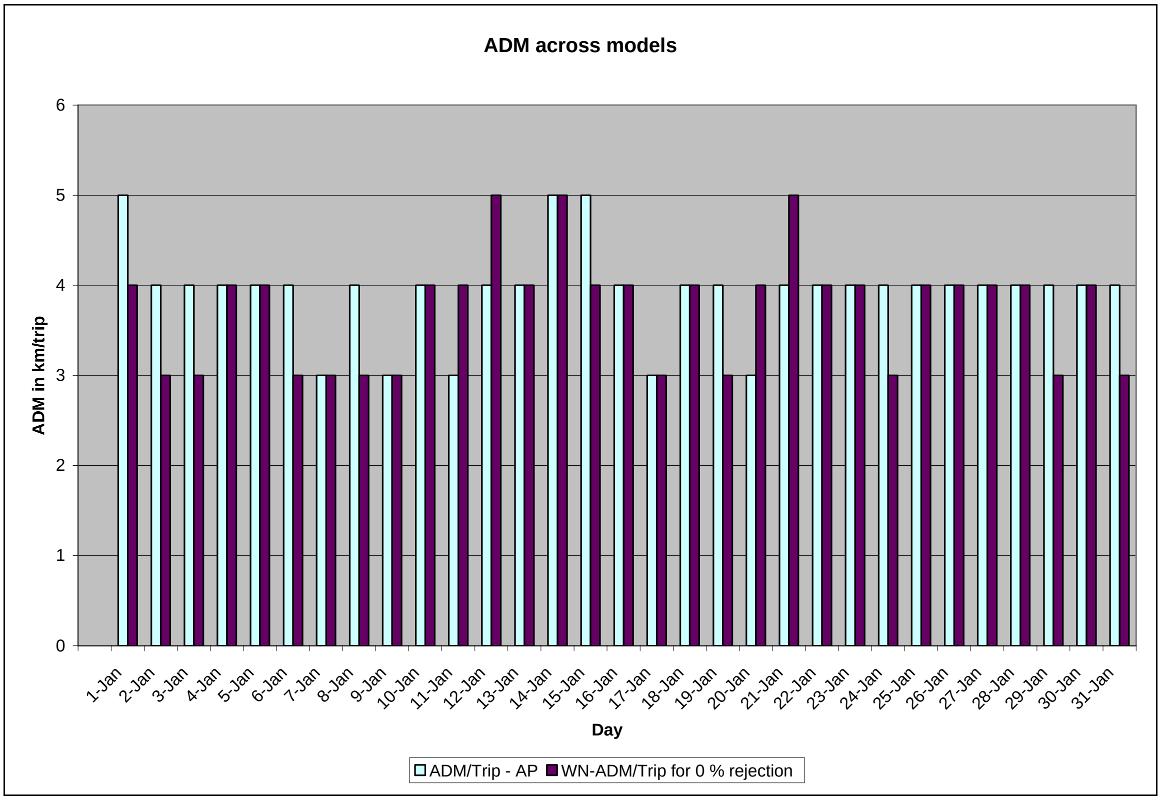

Figure 6.4 Model Vs ADM

Inferences:

- The average dead mileage (ADM) per trip for the apriori and the WN model (with an operating fleet which ensures 0 % rejection of calls) are 3.94 and 3.74 km respectively. The higher ADM for the apriori model can be explained since a smaller fleet is used as compared to that used in WN model to cater the same number of calls. A larger fleet size will have a greater probability of finding a taxi much close by to assign to the request when required.
- The difference in the dead mileage 0.2 km /trip means a savings of Rs 4200 (0.2*168(average number of calls per day)*31 (number of days in a month)* 4 (operational cost)) on a monthly basis.

6.5 SENSITIVITY ANALYSIS

6.5.1 Sensitivity of revenue generated to demand size

The figure below shows the sensitivity of the revenue generated through the operations to the size of the demand that exists for the services:

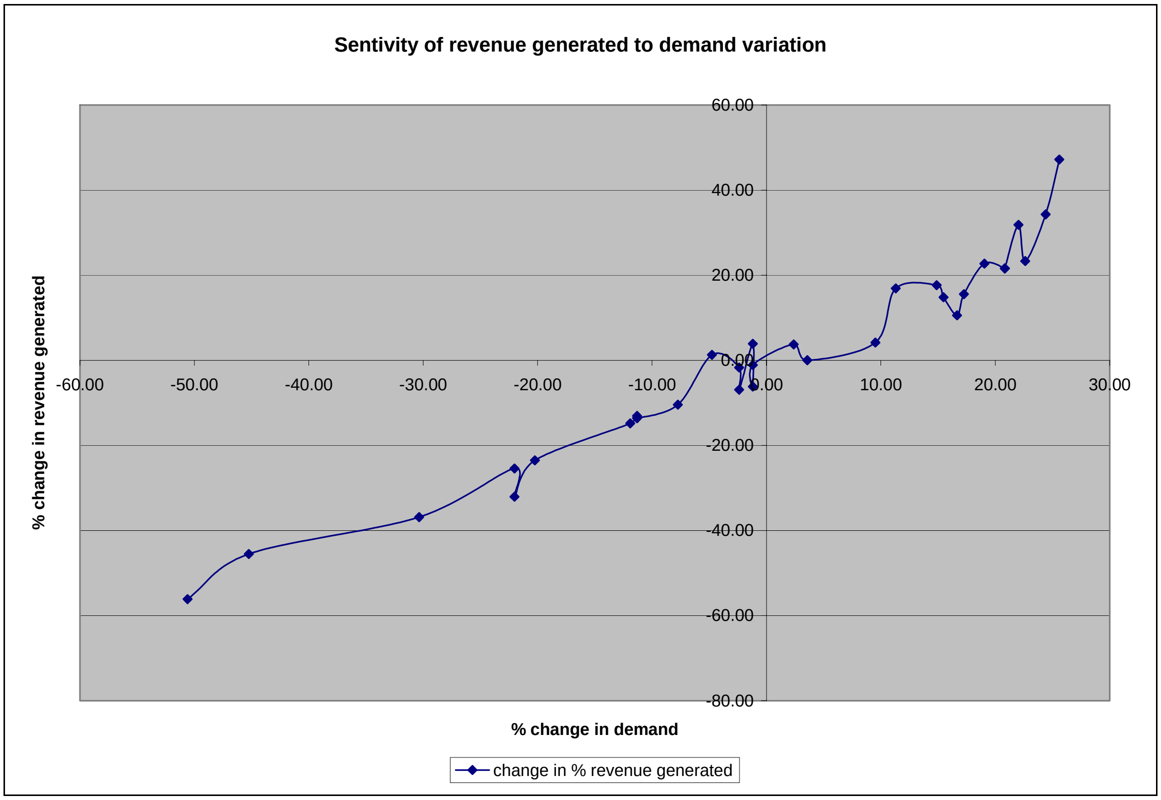

Figure 6.5 Sensitivity of revenue generated to demand

The mean demand (no. of calls) and revenue was 168 calls and Rs. 8194. Percentage change was calculated with respect to these observed means. Revenue as generated by apriori model is considered.

Inference:

- The graph is linearly symmetric about the origin, which means that a % change in demand will linearly affect % change in revenue generated to the same extent.

6.5.2 Sensitivity of fleet size requirement to demand size

The figure below shows the sensitivity of the optimal fleet size required to run the operations to the size of the demand that exists for the services:

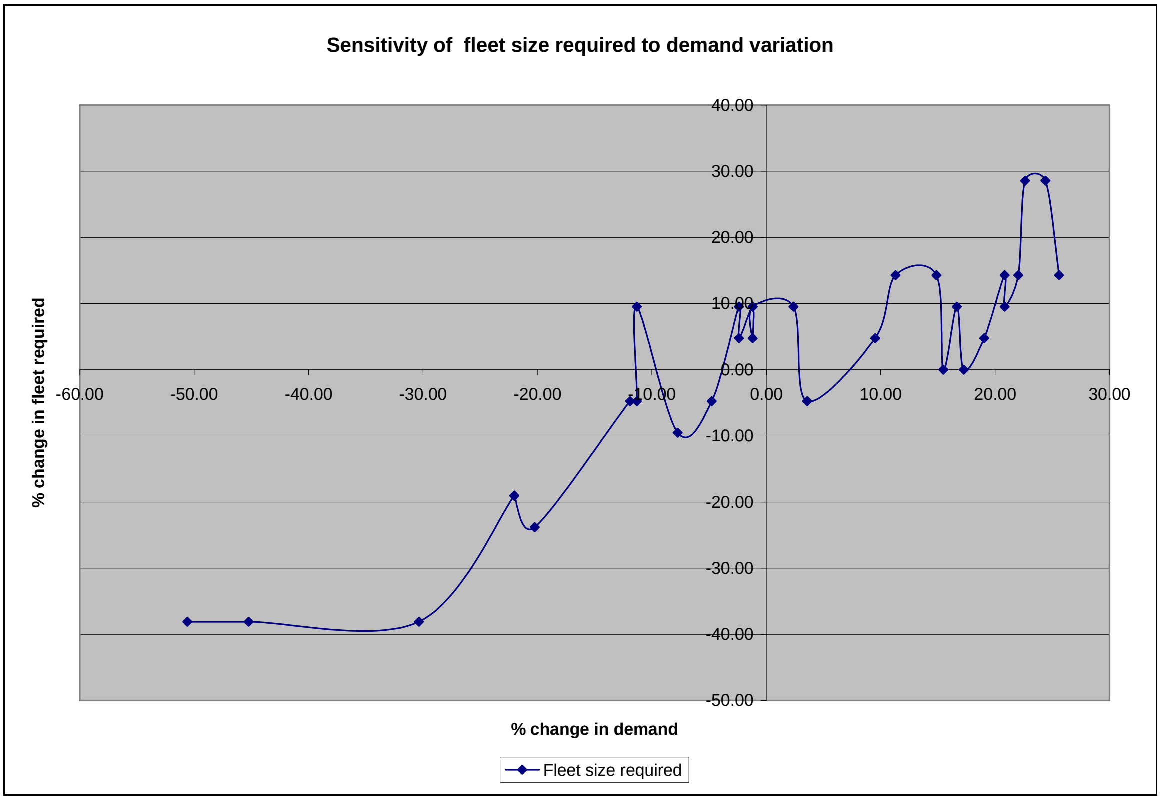

Figure 6.6 Sensitivity of fleet size required to demand

Inference:

- This graph is more interesting, this graph indicates that, the fleet size required to run the operations becomes insensitive to % decrease in demand is more than 30 % . This implies that fleet corresponding that point is the threshold fleet size (the minimum fleet size required to run the operations).

The sensitivity analysis for the WN model is analyzed further and tabulated:

6.5.3 Sensitivity of revenue generated to fleet size maintained in WN model

Table 6.4 Sensitivity of revenue generated to fleet size maintained in WN model

		Revenue generated (Closing) in Rupees			
Fleet size➔		25	20	15	10
Date	No. of calls				
1-Jan	92	4296	4296	4332	4060
2-Jan	149	6876	6800	6536	5096
3-Jan	203	9480	8836	7952	6060
4-Jan	203	9468	9128	7504	5876
5-Jan	187	8772	8676	7124	5108
6-Jan	209	10572	9912	8596	6396
7-Jan	174	7724	7632	7000	4896
8-Jan	131	5760	5760	5668	4644
9-Jan	166	8032	8004	7432	5980
10-Jan	206	8908	8744	7248	5672
11-Jan	211	11368	11024	9684	6584
12-Jan	205	9688	9280	8116	5928
13-Jan	164	7460	7480	6800	5204
14-Jan	83	3864	3772	3536	3436
15-Jan	117	4848	4896	4920	4364
16-Jan	148	6164	6132	5988	4772
17-Jan	193	8940	8292	7344	5356
18-Jan	200	9480	9336	8264	6372
19-Jan	196	8652	8212	7096	5528
20-Jan	197	8564	8532	7652	5740
21-Jan	164	7280	7256	6776	5028
22-Jan	131	4760	4944	5128	4404
23-Jan	166	6932	6928	6420	5252
24-Jan	172	8404	8128	7616	5572
25-Jan	194	8820	8668	8012	6192
26-Jan	155	6988	7000	6800	5604
27-Jan	166	7624	7496	6716	5660
28-Jan	149	7000	6900	6424	5168
29-Jan	134	6124	6124	6012	4824
30-Jan	160	7400	7180	6508	5260
31-Jan	184	8200	7716	6820	5412

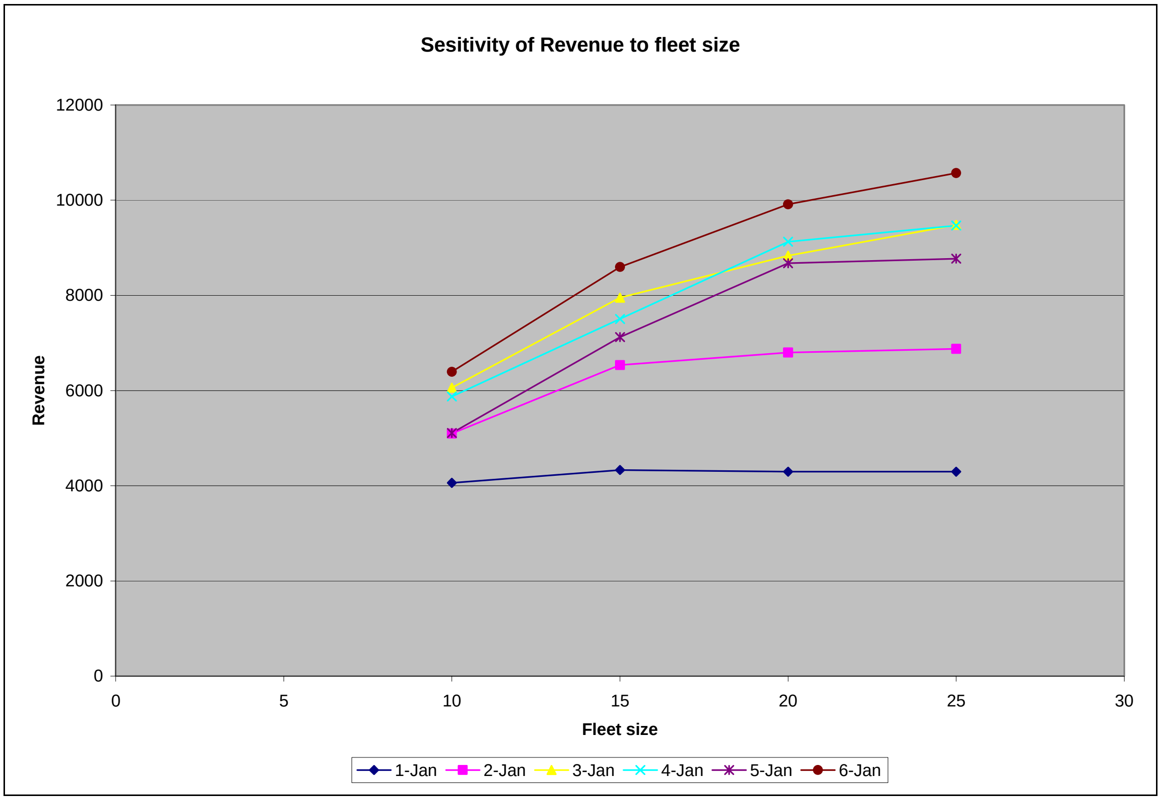

Figure 6.7 Sensitivity of revenue generated to fleet size maintained in WN model

Inference:

- It can be clearly seen that, the revenues increase very steeply between 10 and 15 fleet sizes and does not increase by the same rate after 15. So, if the operator is having a capital constraint on the number of vehicles to maintain, this input will indicate that maintaining a fleet size of 15 will lead to more marginal benefits.
- The revenue for the demand stabilizes after fleet size 30. This indicates that for the given fixed demand 30 vehicles are more than sufficient to satisfy them. In other words, there will be no rejection of calls due to fleet constraint.

6.5.4 Sensitivity of ADM to fleet size maintained in WN model

Table 6.5 Sensitivity of ADM to fleet size maintained in WN model

		ADM in km /trip			
Fleet size ➔		**25**	**20**	**15**	**10**
Date	**No. of calls**				
1-Jan	92	3	3	4	5
2-Jan	149	3	3	4	5
3-Jan	203	3	4	5	6
4-Jan	203	3	4	5	6
5-Jan	187	4	4	5	7
6-Jan	209	3	4	5	6
7-Jan	174	3	3	4	6
8-Jan	131	3	3	3	5
9-Jan	166	3	3	4	5
10-Jan	206	4	4	5	6
11-Jan	211	3	4	4	6
12-Jan	205	4	5	6	7
13-Jan	164	3	4	5	7
14-Jan	83	4	4	5	5
15-Jan	117	3	4	4	5
16-Jan	148	3	4	4	5
17-Jan	193	3	4	5	6
18-Jan	200	3	4	5	6
19-Jan	196	3	4	5	6
20-Jan	197	3	4	5	6
21-Jan	164	4	4	5	6
22-Jan	131	4	4	4	5
23-Jan	166	4	4	4	5
24-Jan	172	3	3	4	5
25-Jan	194	4	4	4	5
26-Jan	155	3	4	4	5
27-Jan	166	3	4	5	5
28-Jan	149	3	3	4	5
29-Jan	134	3	3	3	4
30-Jan	160	4	4	5	6
31-Jan	184	3	3	4	5

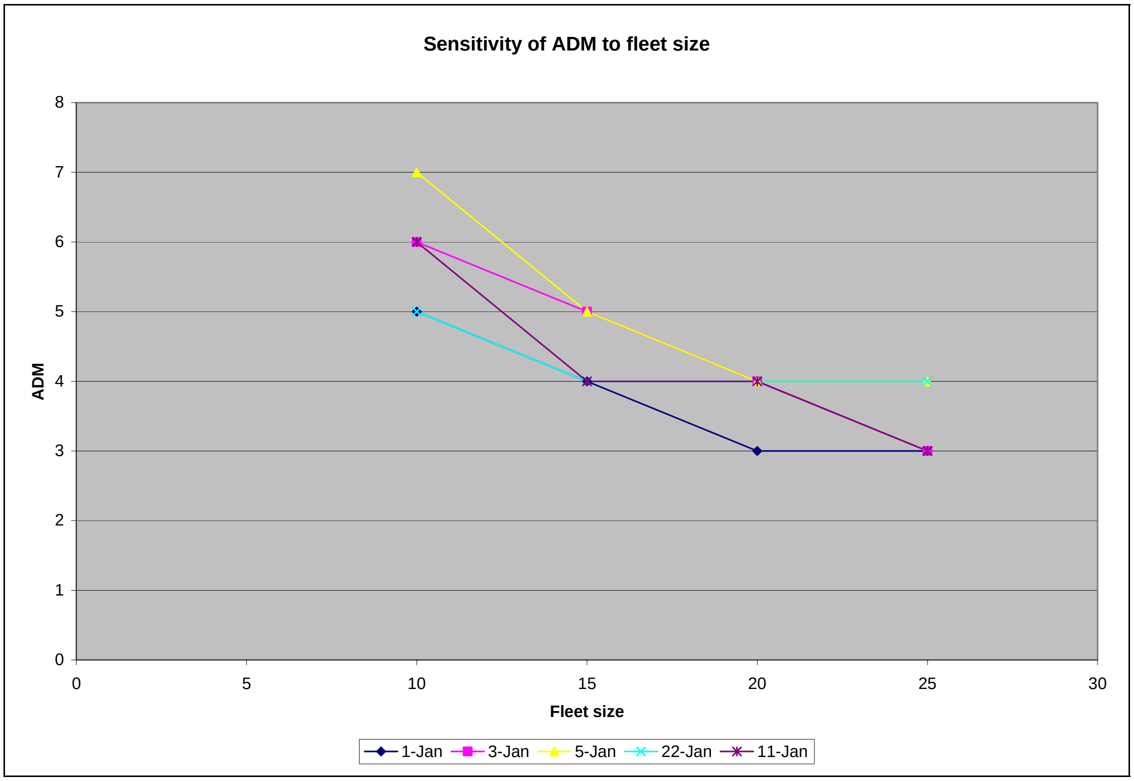

Figure 6.8 Sensitivity of ADM to fleet size maintained in WN model

Inferences:

- ADM (in km/trip) sharply falls when fleet is increased from 10 to 20 and thereafter remains constant. This is intuitionally correct and the effect can be easily visualized.
- As the fleet size increases, more number of taxis will be available over the network and can cater to the demands quickly and from closer destinations. This leads to the observed decrease in the average dead mileage incurred by the taxis.
- But, above a certain limit, since constant demand is being assumed to be served on each day, the increase does not further contribute to decrease in ADM.

6.5.5 Sensitivity of % Calls rejected to fleet size maintained in WN model

Table 6.6 Sensitivity of % calls rejected to fleet size maintained in WN model

		% Calls rejected			
Fleet size ➔		25	20	15	10
Date	No. of calls				
1-Jan	92	0	0	0	3
2-Jan	149	0	3	8	18
3-Jan	203	0	1	7	24
4-Jan	203	0	0	7	26
5-Jan	187	0	0	7	25
6-Jan	209	0	3	11	26
7-Jan	174	0	0	2	17
8-Jan	131	0	0	1	12
9-Jan	166	0	1	4	17
10-Jan	206	0	2	11	25
11-Jan	211	0	0	7	25
12-Jan	205	0	0	4	22
13-Jan	164	0	0	7	15
14-Jan	83	0	0	0	0
15-Jan	117	0	0	0	9
16-Jan	148	0	0	1	15
17-Jan	193	0	1	6	24
18-Jan	200	0	0	4	22
19-Jan	196	0	1	7	22
20-Jan	197	0	0	4	20
21-Jan	164	0	0	0	15
22-Jan	131	0	0	0	11
23-Jan	166	0	0	4	19
24-Jan	172	0	2	6	20
25-Jan	194	0	2	7	23
26-Jan	155	0	0	1	14
27-Jan	166	0	0	3	18
28-Jan	149	0	0	0	12
29-Jan	134	0	0	3	16
30-Jan	160	1	5	10	23
31-Jan	184	1	4	10	23

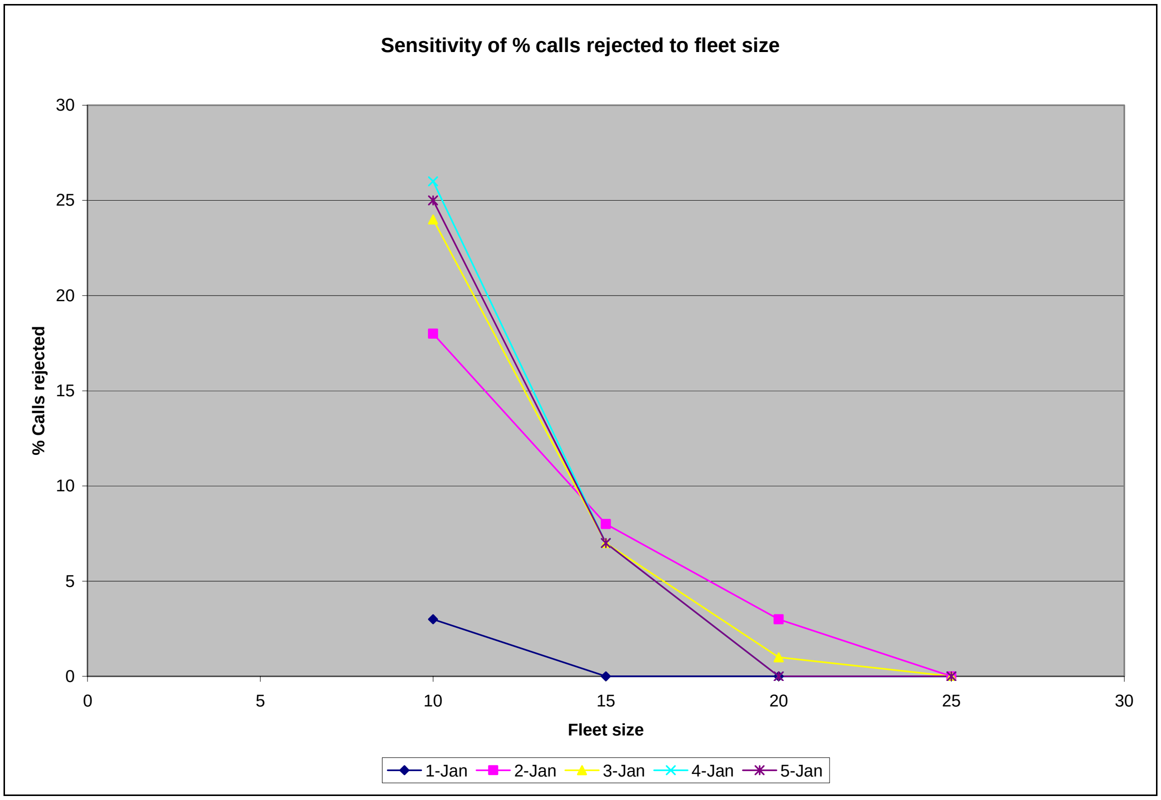

Figure 6.9 Sensitivity of % calls rejected to fleet size maintained in WN model

Inferences:

- Section 6.5.3 indicated that the % rejection of calls will end when the fleet size reaches a limit. The above graph validates and reinforces that prediction.
- The above graph can be used to find out the limiting fleet size. It can be seen that after a fleet size of 25, the % rejection of calls reaches the value zero, which means in this case the limiting fleet is 25.

6.5.6 Sensitivity of Taxi waiting times to call-offset time in WN model

Table 6.7 Sensitivity of taxi waiting times to call-offset time in WN model

		Taxi waiting time in minutes		
Call offset time➔		45	30	15
Date	No. of calls			
1-Jan	92	33	18	8
2-Jan	149	32	18	8
3-Jan	203	31	17	9
4-Jan	203	33	18	8
5-Jan	187	31	18	8
6-Jan	209	33	19	8
7-Jan	174	34	20	8
8-Jan	131	33	18	7
9-Jan	166	34	18	8
10-Jan	206	31	18	9
11-Jan	211	33	19	8
12-Jan	205	31	18	9
13-Jan	164	32	17	8
14-Jan	83	31	17	8
15-Jan	117	32	17	7
16-Jan	148	32	17	8
17-Jan	193	33	19	8
18-Jan	200	33	19	9
19-Jan	196	33	18	8
20-Jan	197	32	19	8
21-Jan	164	32	16	6
22-Jan	131	31	17	8
23-Jan	166	32	19	8
24-Jan	172	32	17	7
25-Jan	194	32	18	9
26-Jan	155	33	19	8
27-Jan	166	33	19	8
28-Jan	149	34	19	3
29-Jan	134	33	19	8
30-Jan	160	34	17	7
31-Jan	184	34	19	8

Figure 6.10 Sensitivity of taxi waiting times to call-offset time in WN model

Inferences:

- The time the taxi waits at the origin of a request is the taxi-waiting time. It is a function of timely allocation of a taxi to the request. It is directly proportional to the call-offset time.
- The call-offset time should be maintained as low as possible all the while maintaining the level of service to the customers (no-delay).
- The next section will provide more inputs on how to choose an optimal call-offset time.

6.5.7 Sensitivity of % Calls late serviced to call-offset time in WN model

Table 6.8 Sensitivity of % calls late serviced to call-offset time in WN model

		% Calls late serviced		
Call offset time➔		45	30	15
Date	No. of calls			
1-Jan	92	0	0	30
2-Jan	149	1	2	30
3-Jan	203	1	2	33
4-Jan	203	1	1	26
5-Jan	187	1	1	42
6-Jan	209	4	5	25
7-Jan	174	2	2	20
8-Jan	131	0	0	25
9-Jan	166	0	0	22
10-Jan	206	3	6	29
11-Jan	211	3	3	31
12-Jan	205	4	6	29
13-Jan	164	2	3	35
14-Jan	83	2	4	42
15-Jan	117	0	1	33
16-Jan	148	1	2	28
17-Jan	193	1	3	30
18-Jan	200	4	5	33
19-Jan	196	3	3	32
20-Jan	197	2	2	20
21-Jan	164	4	6	39
22-Jan	131	0	0	25
23-Jan	166	1	3	20
24-Jan	172	1	1	37
25-Jan	194	3	2	27
26-Jan	155	2	5	27
27-Jan	166	1	1	25
28-Jan	149	2	3	24
29-Jan	134	0	2	24
30-Jan	160	0	0	38
31-Jan	184	1	5	22

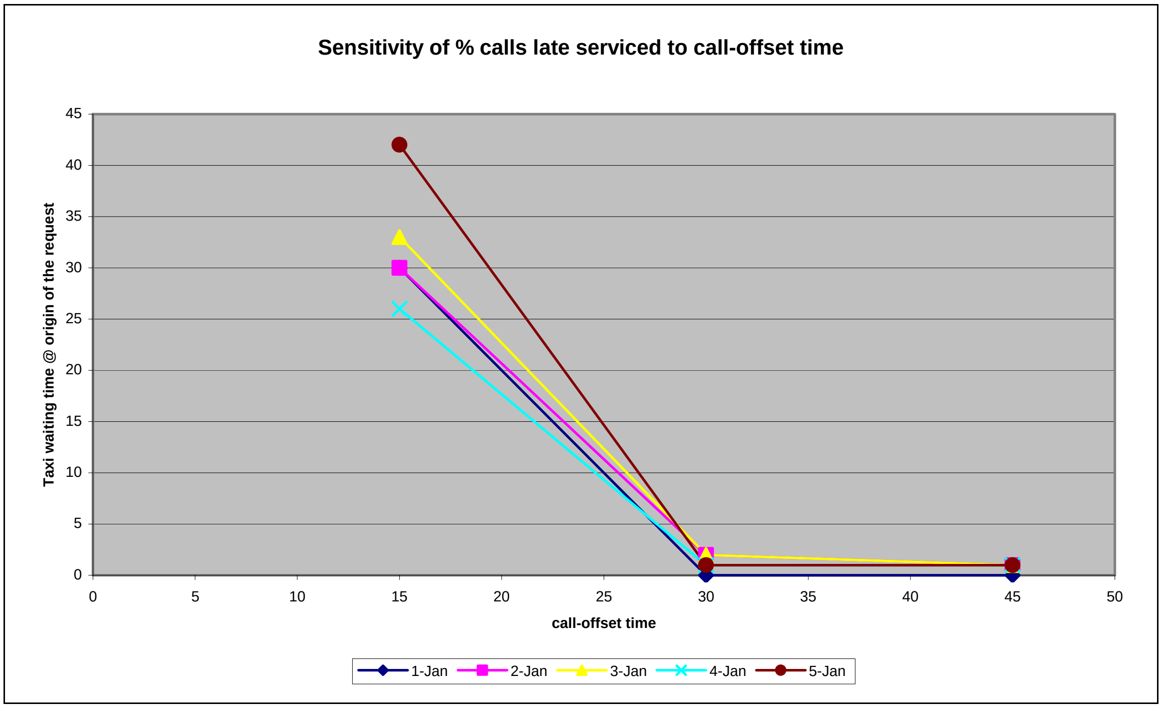

Figure 6.11 Sensitivity of % calls late serviced to call-offset time in WN model

Inferences:

- The above graph provides very valuable input on the call-offset time. It shows consistently that % of calls late serviced very sharply increases if call-offset time goes below 30 min.
- It also indicates that having a call-offset time of 45 minutes over 30 minutes does not provide any marginal benefit in terms of delay caused to customers.
- So it can be concluded from the above two sections that call-offset time of 30 minutes needs to be chosen. This threshold call-offset time is a function of the distance between the zones in the network.

6.6 SUMMARY

The above analysis gives us insights into the operational and management aspects of a typical call - taxi operations. The goodness of various heuristics and how the heuristics can be intelligently used to reduce the gap between the heuristic and the global optimal solution can be inferred. This analysis enables efficient planning of the operations (fleet size, call-offset time, type of heuristics) thus, resulting in a better performance of the system. In summary the above analysis serves as a good decision support application.

CHAPTER 7

CONCLUSIONS

7.1 SUMMARY

The type of strategies adopted to make the routing and scheduling decisions to run a call-taxi system will determine its operational efficiency. The primary objective of this study was to propose and develop several such strategies which will out-perform the existing crude methods and practices used in the field.

In order to accomplish the objective two heuristics strategies “Wait at node” and “Move to a desirable node” have been developed and tested. Furthermore, a global optimization model for the hind-sight problem, “The apriori model” has also been developed to let compare the heuristic solutions with this best possible optimal solution at the end of the day to day operations. Simulations in Java were coded to run and validate these real-time strategies and optimization models using real data obtained from a call taxi firm operating in Chennai city.

The performance measures obtained from various strategies were compared to identify the extent of improvement over the strategies used in practice. In order to get insights, sensitivity analysis was also performed on various parameters like fleet size and call-off set time to predict the effect on the performance measures like revenue generated, average dead mileage, taxi waiting time and percentage of calls rejected.

Some of the important findings are listed below:

1. Demand is high in the start and the end of the month and it falls sharply on weekends and national holidays.
2. A very large portion of the demand is directed around the airport and the central railway stations zones.
3. Heuristic-optimal values of fleet size and call-offset time are 25 taxis and 30 minutes respectively for an average demand size of 160-180 calls per day.
4. Significant and steep reduction in % of calls rejected when the fleet size increase from 10 to 15 and the improvement reaches saturation at the heuristic optimal value of the fleet size i.e. 25 for a demand size of 160-180 calls/day.
5. More the number of taxis maintained, lesser will be the ADM, but the capital investment cost and monthly maintenance cost shoots up.
6. Taxi waiting times increase linearly as call-offset time increases, but this is essential to maintain the level of service to the customer (causing no delay)
7. There exists an average of just 8 % gap between the WN model and the apriori solution which means this strategy is very close to the optimal solution and can be adopted over the existing strategies in practice to see immediate jump in the performance of the system.

Through this study the relation and trade-offs between different decision variable and performance measures were analyzed and understood. These inputs enable the application developed to serve as a good decision support system for making planning and as well as operational level decisions.

7.2 DIRECTIONS FOR FURTHER RESEARCH

Future work can focus on the following issues:

1. The model can be applied to various other cities to quantify the effect of variations in demand patterns, difference in network components (number of nodes, links and corresponding travel times), different cost structures (driver and fuel) and changes in other local conditions.
2. A model which quantifies the effect of variations in demand across a month (weekdays, weekends) and within a day (peak hour and non-peak hour) needs to be developed. More advanced and far-sighted strategies, which take into account the stochastic characteristics of the future demands which involve stochastic programming, can also be explored.
3. An elaborate model to solve a more complicated instance of the problem which use all real-time information available and involve real-time congestion on the network links, sudden break-down of vehicles, driver behavior, passenger behavior (cancellations) will add a lot of value to the existing literature.
4. Applications and tools which integrate the mathematical models with the technological advancements in the fields of ITS and GPS should be developed.
5. These models developed for call-taxi services can be extended into developing car – pool and call-bus systems. These types of services would ensure the high level of service and also would be more affordable by a larger section of the population.

REFERENCES

1. Hyunmyung Kim, Jun-Seok Oh, and Jayakrishnan R (2005), "Effect of Taxi Information System on Efficiency and Quality of Taxi Services", *Transportation Research Record: Journal of Transportation Research*, No. 1903, pp. 96 – 104.
2. Yang J., Jaillet P. and Mahmassani H.S. (1999), “On-line Algorithms for Truck Fleet Assignment and Scheduling under Real-time Information”, *Transportation Research Record*, 1667, pp. 107-113.
3. Kim, Jaillet, P. and Mahmassani, H.S. (2001), “Dynamic Truck Operation Strategies for Truckload Pickup and Delivery Problems with Time Windows,” *Proceedings of the TRISTAN IV Conference, Triennial Symposium on Transportation Analysis*, Azores Island, Portugal June 10-22.
4. Yang H. and Wong S.C. (1998), “A Network Equilibrium Model of Urban Taxi Services”, Paper presented at the *77th TRB Annual Meeting*, 11-15 January, Washington D.C., USA.
5. Wong K.I., Wong S.C. and Yang H. (2000), “Modeling Urban Taxi Services in Congested Road Networks with Elastic Demand”, Paper presented at the *79th TRB Annual Meeting*, 9-13 January, Washington D.C., USA.
6. Wong K.I., Wong S.C. and Yang H. (2000), “Computational Efficiency of Two Solution Algorithms for an Urban Taxi Model”, *Proceeding of the Fifth Conference of Hong Kong Society for Transportation Studies*, 2 December, Hong Kong, pp 245-254.

7. Wong K.I., Wong S.C., Yang H. and Wu J.H. (2001), "Formulation of a Multi-class Model for Urban Taxi Services", *Proceedings of the Sixth Conference of Hong Kong Society for Transportation Studies* , 1 December, Hong Kong, pp 240-249.
8. Wong K.I., Wong S.C., Yang H. and Wu J.H. (2004), "Modeling Urban Taxi Services with Multiple User Classes and Vehicle Modes", *Proceedings of the Croucher Advanced Study Institute (ASI) on Advanced Modeling for Transit Supply and Passenger Demand*, 19-20 January, Hong Kong, pp A1.1-A1.28.
9. Prakash S. and Madhav N. (2003), "Fleet Management using a GPS-GIS Integrated Approach", *Map India Conference*, Noida, India.
10. Alla, N. (2004), "Optimization of Call-taxi Operations using Simulation", Bachelor's thesis, Indian Institute of Technology, Madras.
11. Paul E. Black, (2005). Floyd - Warshall algorithm. Dictionary of algorithms and Data Structures. http://www.nist.gov/dads/HTML/floydWars.hall.html. Accessed January 2006.
12. Dinakar A, (August 24, 2001), "Call – taxis calling the shots in Chennai", http://www.hinduonnet.com/2001/08/24/stories/0424401x.htm. Accessed September 2006.
13. Ramakrishnan N, (March 16, 2002), "Hugely popular and win – win deal for all", http://www.thehindubusinessline.com/2002/03/16/stories/200203160208190.htm. Accessed September 2006.

14. Philip S, (November 24, 2002), “Taxi services in Chennai city”, http://www.chennaionline.com/cityfeature/calltaxi.asp. Accessed September 2006.

Source codes:

The source codes of the simulations models and relevant software can be downloaded from the following URL: www.k.abishek.googlepages.com/mtpdownloads

APPENDIX

TABLE A.1 TRAVEL DISTANCE BETWEEN ZONAL NODES (in km)

	Destination →											
Origin	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1.45	4.8	2.54	3.89	3.89	4.51	6.5	7.9	9.86	10.29	11.36
2	1.45	0	3.51	1.74	2.44	2.44	4.22	5.32	6.72	8.68	9.11	10.18
3	4.8	3.51	0	2.79	1.36	2.75	3	3.24	4.64	6.6	7.03	8.1
4	2.54	1.74	2.79	0	2.21	2.29	3.53	4.6	6	7.96	8.39	9.46
5	3.89	2.44	1.36	2.21	0	1.5	2.9	3.37	4.77	6.73	7.16	8.23
6	3.89	2.44	2.75	2.29	1.5	0	4.4	4.87	6.27	8.23	8.66	9.73
7	4.51	4.22	3	3.53	2.9	4.41	0	2.55	3.95	5.91	6.34	7.41
8	6.5	5.32	3.24	4.6	3.37	4.87	2.55	0	1.92	3.88	4.31	5.38
9	7.9	6.72	4.64	6	4.77	6.27	9.95	1.92	0	2.86	2.39	3.61
10	9.86	8.68	6.6	7.96	6.73	8.23	5.91	3.88	2.86	0	3.41	4.63
11	10.29	9.11	7.03	8.39	7.16	8.66	6.34	4.31	2.39	3.41	0	2.4
12	11.36	10.18	8.1	9.46	8.23	9.73	7.41	5.38	3.61	4.63	2.4	0
13	10.19	9.01	6.74	8.29	7.06	6.75	6.24	4.21	3.53	5.49	5.92	5.73
14	11.69	10.51	8.43	9.79	8.56	8.97	7.74	5.71	4.39	5.93	5.46	4.39
15	4.63	5.61	8.36	6.59	7.29	7.29	8.52	10.17	11.57	13.53	13.96	15.03
16	3.32	4.51	7.53	5.6	6.46	6.46	7.21	9.2	10.6	12.56	12.99	14.06
17	3.12	4.05	6.8	5.03	5.73	5.73	7.01	8.61	10.01	11.97	12.4	13.47
18	4.3	3.91	5.96	4.19	4.89	4.89	7.3	7.77	9.17	11.13	11.56	12.63
19	5.81	5.9	7.95	6.18	6.88	6.73	9.29	9.76	11.16	13.12	13.55	14.62
20	4.7	4.31	6.36	4.59	5.29	5.28	7.7	8.17	9.57	11.53	11.96	13.03
21	6.57	5.75	6.93	6.03	5.68	5.14	8.58	9.02	10.42	12.38	12.81	13.88
22	4.61	3.37	4.55	3.65	3.3	2.76	6.2	6.64	8.04	10	10.43	11.5
23	4.82	3.58	4.76	3.86	3.51	2.97	6.41	6.85	8.25	10.21	10.64	11.71
24	6.73	5.49	5.1	5.77	5.2	4.05	6.77	6.34	7.74	9.7	10.13	11.2
25	7.27	6.03	6.28	6.31	5.96	5.23	8.33	7.9	9.3	11.26	11.69	12.76
26	7.49	7.64	10.2	8.62	9.28	8.74	11.04	12.01	13.41	15.37	15.8	16.87
27	5.47	4.23	3.9	4.51	4	2.85	6.06	5.63	7.03	8.99	9.42	10.49
28	6.77	5.5	3.55	5.35	4.212	3.06	5.02	4.59	5.99	7.95	8.38	9.45
29	7.59	6.35	5.43	6.63	5.53	4.38	7.1	6.67	8.07	10.03	10.46	11.53
30	8.34	6.96	4.89	6.81	5.67	4.52	6.36	5.93	7.23	9.19	9.62	10.69
31	7.76	6.31	3.99	6.16	4.92	4	5.37	3.34	2.66	4.62	5.05	6.12
32	6.27	4.82	2.5	4.67	3.43	2.51	3.97	3.54	4.94	6.9	7.33	8.4
33	9.88	8.5	6.43	8.35	7.21	6.06	7.9	6.23	5.55	7.51	7.94	9.01
34	12.2	10.82	8.75	10.67	9.53	8.38	10.22	8.73	8.05	10.01	10.44	11.51
35	8.61	7.37	5.74	7.54	6.4	5.25	7.21	6.78	8.18	10.14	10.57	11.64
36	10.38	9.16	7.88	9.44	8.34	7.19	9.35	8.92	10.32	12.28	12.71	13.78
37	7.78	6.54	6.79	6.82	6.47	5.74	8.58	8.15	9.55	11.51	11.94	13.01
38	8.41	8.05	7.13	8.33	7.23	6.08	8.8	8.37	9.77	11.73	12.16	13.23
39	11.08	9.86	8.58	10.14	9.04	7.89	10.05	9.62	11.02	12.98	13.41	14.48
40	12.43	11.05	8.98	10.9	9.76	8.61	10.45	8.78	8.1	10.06	10.49	11.56
41	13.33	11.95	9.88	11.8	10.66	9.51	11.35	9.86	9.18	11.14	11.57	12.64
42	8.56	7.72	7.6	8	7.65	6.55	9.27	8.84	10.24	12.2	12.63	13.7
43	7.87	8.02	8.41	8.81	8.46	7.36	10.08	9.65	11.05	13.01	13.44	14.51
44	6.27	6.42	9.17	7.4	8.1	8.1	9.82	10.98	12.38	14.34	14.77	15.84
45	2.18	1.55	3.48	1.48	2.9	3.22	2.89	4.88	6.28	8.24	8.67	9.74
46	3.7	3.85	6.6	4.83	5.53	5.53	7.25	8.41	9.81	11.77	12.2	13.27
47	5.96	5.57	7.62	5.85	6.55	6.34	8.96	9.43	10.83	12.79	13.22	14.29
48	12.66	12.32	14.37	12.6	13.21	12.67	15.71	16.18	17.58	19.54	19.97	21.04
49	7.36	6.47	7.65	6.75	6.4	5.86	9.3	9.74	11.14	13.1	13.53	14.6
50	8.56	8.71	11.46	9.69	10.39	10.39	12.11	13.27	14.67	16.63	17.06	18.13
51	7.23	7.38	8.75	8.18	7.83	7.29	10.73	10.56	11.96	13.92	14.35	15.42
52	9.77	9.92	12.32	10.9	11.23	10.69	13.32	14.13	15.53	17.49	17.92	18.99
53	8.57	7.9	9.08	8.18	7.83	7.29	10.73	11.17	12.57	14.53	14.96	16.03
54	10	9.61	11.39	9.89	10.14	9.6	13	13.47	14.87	16.83	17.26	18.33
55	8.92	9.07	11.82	10.05	10.75	10.75	12.47	13.63	15.03	16.99	17.42	18.49
56	9.34	9.49	12.24	10.47	11.17	11.17	12.89	14.05	15.45	17.41	17.84	18.9
57	9.64	9.66	11.71	9.94	10.55	10.01	13.05	13.52	14.92	16.88	17.31	18.38
58	7.94	8.09	10.84	9.07	9.77	9.77	11.49	12.65	14.05	16.01	16.44	17.51
59	5.84	6.9	9.92	7.99	8.85	8.85	9.73	11.72	13.12	15.08	15.51	16.58
60	10.28	11.34	14.22	12.43	13.15	13.15	14.17	16.03	17.43	19.39	19.82	20.89

TABLE A.1 TRAVEL DISTANCE BETWEEN ZONAL NODES (in km)

	Destination →												
Origin	13	14	15	16	17	18	19	20	21	22	23	24	25
1	10.19	11.69	4.63	3.32	3.12	4.3	5.81	4.7	6.57	4.61	4.82	6.73	7.27
2	9.01	10.51	5.61	4.51	4.05	3.91	5.9	4.31	5.75	3.37	3.58	5.49	6.03
3	6.59	8.43	8.36	7.53	6.8	5.96	7.95	6.36	6.93	4.55	4.76	5.1	6.28
4	8.29	9.79	6.59	5.6	5.03	4.19	6.18	4.59	6.03	3.65	3.86	5.77	6.31
5	7.06	8.56	7.29	6.46	5.73	4.89	6.88	5.29	5.66	3.3	6.51	5.2	5.96
6	6.75	8.97	7.29	6.46	5.73	4.89	6.73	5.28	5.14	2.76	2.97	4.05	5.23
7	6.24	7.74	8.52	7.21	7.01	7.31	9.29	7.7	8.58	6.21	6.42	6.77	8.33
8	4.21	5.71	10.17	9.2	8.61	7.77	9.76	8.17	9.02	6.64	6.85	6.34	7.9
9	3.53	4.39	11.57	10.6	10.01	9.17	11.16	9.57	10.42	8.04	8.25	7.74	9.3
10	5.49	5.93	13.53	12.56	11.97	11.13	13.12	11.53	12.38	10	10.21	9.7	11.26
11	5.92	5.46	13.96	12.99	12.4	11.56	13.55	11.96	12.81	10.43	10.64	10.13	11.69
12	5.73	4.39	15.03	14.06	13.47	12.63	14.62	13.03	13.88	11.5	11.71	11.2	12.76
13	0	3.14	13.43	12.6	11.87	10.84	12.35	10.9	10.76	3.83	8.59	8.08	9.64
14	3.14	0	15.36	14.39	13.8	12.96	14.57	13.12	12.98	10.6	10.81	10.3	11.86
15	13.43	15.36	0	2.19	2.02	3.61	3.23	4.18	4.78	5.76	6.76	9.11	7.9
16	12.6	14.39	2.19	0	1.41	3.22	4.34	3.83	5.7	5.37	6.37	8.76	8.39
17	11.87	13.8	2.02	1.41	0	2.05	3.47	2.66	4.53	4.2	5.2	8.03	7.22
18	10.84	12.96	3.61	3.22	2.05	0	2.97	1.38	3.25	2.7	3.7	6.8	6.02
19	12.35	14.57	3.23	4.34	3.47	2.97	0	1.59	2.19	4.21	4.45	7.32	6.03
20	10.9	13.12	4.18	3.83	2.66	1.38	1.59	0	2.26	2.76	3.17	6.27	5.43
21	10.76	12.98	4.78	5.7	4.53	3.25	2.19	2.26	0	3.47	2.57	5.67	3.84
22	8.38	10.6	5.76	5.37	4.2	2.7	4.21	2.76	3.47	0	1.41	4.54	4.6
23	8.59	10.81	6.76	6.37	5.2	3.7	4.45	3.17	2.57	1.14	0	3.97	3.7
24	8.08	10.3	9.11	8.76	8.03	6.8	7.32	6.27	5.67	4.54	3.97	0	2.48
25	9.64	11.86	7.9	8.39	7.22	6.02	6.03	5.43	3.84	4.6	3.7	2.48	0
26	13.75	15.97	5.83	6.32	5.15	5.64	4.04	4.99	4.56	7.07	6.17	6.59	4.11
27	7.37	9.54	8.33	7.5	6.77	5.57	6.32	5.04	4.44	3.28	2.73	2.05	2.99
28	6.33	8.55	9.63	8.8	8.07	7.04	8.24	6.96	6.36	4.58	4.65	3.13	4.69
29	8.41	10.63	9.14	9.62	8.46	7.86	7.35	7.42	6.82	5.4	5.25	2.21	4.01
30	7.32	9.54	11.2	10.37	9.64	8.61	9.33	8.6	8	6.15	6.36	4	5.56
31	2.75	4.97	10.68	9.85	9.12	8.09	9.6	8.15	8.01	5.63	5.84	5.33	6.89
32	5.28	7.5	9.19	8.36	7.63	6.6	8.11	6.66	6.52	4.14	4.35	3.5	5.06
33	5.64	7.86	11.27	11.76	10.59	10.15	9.48	10.14	9.54	7.69	7.9	5.81	7.37
34	8.14	10.36	12.66	13.15	11.98	12.47	10.87	11.82	11.86	10.01	10.22	7.54	9.08
35	8.32	10.54	9.47	9.96	8.79	8.88	7.68	8.44	7.84	6.42	6.27	3.47	5.03
36	10.66	12.88	8.72	9.21	8.04	8.53	6.92	7.88	8.48	8.21	8.06	4.92	6.07
37	9.89	12.11	8.48	8.97	7.8	7.85	6.69	7.32	6.42	5.59	5.02	1.81	2.58
38	10.11	12.33	6.75	7.24	6.07	6.56	4.96	5.91	6.51	7.1	6.81	3.55	4.57
39	11.36	13.58	9.42	9.91	8.74	9.23	7.63	8.58	9.18	8.91	8.76	5.62	6.77
40	8.19	10.41	13.21	13.7	12.53	12.7	11.42	12.37	12.09	10.24	10.45	8.09	9.63
41	9.27	11.49	13.79	14.28	13.11	13.6	12	12.95	12.99	11.14	11.35	8.67	10.21
42	10.58	12.8	6.9	7.39	6.22	6.71	5.11	6.06	6.66	6.77	6.2	2.99	3.08
43	11.39	13.61	6.21	6.7	5.53	6.02	4.42	5.37	5.97	7.58	7.01	3.8	4.67
44	13.39	15.61	4.61	5.1	3.93	4.42	2.82	3.77	4.37	6.39	6.62	5.8	4.56
45	8.57	10.07	6.19	4.88	4.68	4.95	6.69	5.35	6.79	4.41	4.62	6.53	7.07
46	11.67	13.6	2.47	2.53	1.36	1.85	2.11	2.46	3.66	4	5	7.07	5.86
47	11.96	14.18	3.78	4.89	3.92	2.64	1.19	1.26	1.2	3.88	3.77	6.87	5.04
48	18.29	20.51	9.06	11.13	10.32	9.39	7.31	8.01	7.53	10.63	10.1	11.6	9.12
49	11.48	13.7	6.29	6.49	5.32	4.04	3.7	3.1	1.51	4.19	3.29	5.5	3.02
50	15.86	18.08	6.9	7.39	6.22	6.71	5.11	6.06	6.66	8.68	8.28	8.7	6.22
51	12.3	14.52	5.57	6.06	4.89	5.38	3.78	4.73	3.56	5.62	4.72	5.14	2.66
52	15.87	18.09	8.11	8.6	7.43	7.95	5.87	6.57	6.09	9.02	8.12	8.71	6.23
53	12.91	15.13	6.39	7.5	6.53	5.25	3.17	3.87	2.87	5.62	4.72	7.82	5.03
54	15.22	17.44	7.82	8.93	7.96	6.68	4.6	5.3	4.82	7.92	7.03	10.13	7.34
55	16.22	18.44	7.26	7.75	6.58	7.07	5.47	6.42	7.02	9.04	8.64	9.06	6.58
56	16.64	18.86	7.68	8.17	7	7.49	5.89	6.84	7.1	9.46	8.95	9.48	7
57	15.63	17.85	5.18	7.25	6.78	6.73	4.65	5.35	4.87	7.97	7.44	10.54	8.59
58	15.91	17.84	3.48	5.55	5.08	6.09	5.62	6.57	7.17	8.24	9.24	11.5	10.29
59	14.99	16.91	2.37	3.21	3.73	5.54	5.29	6.15	6.84	7.69	8.69	11.15	9.96
60	19.29	21.22	6.86	7.65	8.17	9.47	9	9.95	10.55	11.62	12.62	14.88	13.67

TABLE A.1 TRAVEL DISTANCE BETWEEN ZONAL NODES (in km)

	Destination →											
Origin	25	26	27	28	29	30	31	32	33	34	35	36
1	7.27	7.49	5.47	6.77	7.59	8.34	7.76	6.27	9.9	12.2	8.61	10.3
2	6.03	7.64	4.23	5.5	6.35	6.96	6.31	4.82	8.52	10.82	7.37	9.16
3	6.28	10.2	3.9	3.4	5.43	4.74	3.84	2.35	6.3	8.6	5.59	7.73
4	6.31	8.62	4.51	5.42	6.63	6.81	6.16	4.67	8.44	10.67	7.54	9.44
5	5.96	9.28	4	4.21	5.53	5.67	4.92	3.43	7.23	9.53	6.4	8.34
6	5.23	8.74	2.85	3.06	4.38	4.52	4	2.51	6.08	8.38	5.25	7.19
7	8.33	11.04	6.06	5.02	7.1	6.36	5.37	3.97	7.92	10.22	7.21	9.35
8	7.9	12.01	5.63	4.59	6.67	5.93	3.34	3.54	6.23	8.73	8.78	8.92
9	9.3	13.41	7.03	5.99	8.07	7.21	2.66	4.94	5.55	8.05	8.18	10.32
10	11.26	15.37	8.99	7.95	10.03	9.17	4.62	6.9	7.51	10.01	10.14	12.28
11	11.69	15.8	9.42	8.36	10.46	9.6	5.05	7.33	7.94	10.44	10.57	12.71
12	12.76	16.87	10.49	9.45	11.53	10.67	6.12	8.4	9.01	11.51	11.64	13.78
13	9.64	13.75	7.37	6.33	8.41	7.3	2.75	5.28	5.64	8.14	8.32	10.66
14	11.86	15.97	9.59	8.55	10.63	9.52	4.97	7.5	7.86	10.36	10.54	12.88
15	7.9	5.83	8.33	9.63	9.14	11.12	10.68	9.19	11.27	12.66	9.47	8.72
16	8.39	6.32	7.5	8.8	9.62	10.37	9.85	8.36	11.76	13.15	9.96	9.21
17	7.22	5.15	6.77	8.07	8.46	9.64	9.12	7.63	10.59	11.98	8.79	8.04
18	6.02	5.64	5.57	7.04	7.86	8.61	8.09	6.6	10.17	12.47	8.88	8.53
19	6.03	4.04	6.32	8.24	7.35	9.33	9.6	8.11	9.48	10.87	7.68	6.93
20	5.43	4.99	5.04	6.96	7.42	8.6	8.15	6.66	10.16	11.82	8.44	7.88
21	3.84	4.56	4.44	6.36	6.82	8	8.01	6.52	9.56	11.86	7.84	8.48
22	4.6	7.07	3.28	4.58	5.4	6.16	5.63	4.14	7.71	10.01	6.42	8.21
23	3.7	6.17	2.73	4.65	5.11	6.29	5.84	4.35	7.85	10.15	6.13	7.92
24	2.48	6.59	2.05	3.13	2.21	4	5.33	3.5	5.83	7.54	3.47	4.92
25	0	4.11	2.99	4.69	4.01	5.56	6.89	5.06	7.39	9.08	5.03	6.07
26	4.11	0	7.1	8.8	8.12	9.67	11	9.17	10.52	11.91	8.72	7.97
27	2.99	7.1	0	2.42	2.38	3.56	4.62	2.79	5.12	7.42	3.4	5.19
28	4.69	8.8	2.42	0	3.24	1.77	3.58	1.5	3.33	5.63	2.81	5.26
29	4.01	8.12	2.38	3.24	0	2.88	5.66	3.83	4.24	5.63	2.35	3.11
30	5.56	9.67	3.56	1.77	2.88	0	4.92	2.39	2.47	3.86	2.45	4.9
31	6.89	11	4.62	3.58	5.66	4.92	0	2.53	4.41	6.91	5.77	7.91
32	5.06	9.17	2.79	1.5	3.83	2.39	2.53	0	3.95	6.25	3.66	6.08
33	7.37	10.52	5.1	3.31	4.24	2.47	4.41	3.93	0	2.86	2.68	5.05
34	9.08	11.91	7.42	5.63	5.63	3.86	6.91	6.25	2.86	0	4.07	6.44
35	5.03	8.72	3.4	2.81	2.35	2.45	5.77	3.66	2.68	4.07	0	3.25
36	6.07	7.97	5.19	5.26	3.11	4.9	7.91	6.08	5.05	6.44	3.25	0
37	2.58	6.69	3.5	4.94	2.79	4.83	7.14	5.31	5.98	7.37	4.18	4.36
38	4.57	6	4.08	4.73	2.55	4.37	7.36	5.53	4.52	5.91	2.72	1.97
39	6.77	8.67	5.89	5.96	3.81	5.6	8.61	6.78	5.75	7.14	3.95	2.18
40	9.63	12.46	7.65	5.86	6.18	4.41	6.96	6.48	2.91	2.67	4.62	6.99
41	10.21	13.04	8.55	6.76	6.76	4.99	8.04	7.38	3.99	3.25	5.2	7.57
42	3.08	6.15	4.55	5.63	3.02	5.41	7.83	6	6.01	7.4	4.21	3.46
43	4.67	5.46	5.36	6.17	3.83	5.81	8.64	6.81	5.96	7.35	4.16	3.41
44	4.56	3.3	7.36	8.17	5.83	7.81	10.64	8.81	47.96	9.35	6.16	5.41
45	7.07	8.37	5.27	6.28	7.39	7.74	7.09	5.6	9.3	11.6	8.41	10.2
46	5.86	3.79	6.57	7.87	7.1	9.08	8.92	7.43	9.23	10.62	7.43	6.68
47	5.04	4.59	5.64	7.56	7.9	9.2	9.21	7.72	10.03	11.42	8.23	7.48
48	9.12	6.33	11.97	13.81	13.13	14.68	15.54	14.05	15.69	17.08	13.89	13.14
49	3.02	3.05	5.16	7.08	7.03	8.58	8.73	7.24	10.28	12.1	8.05	9.09
50	6.22	2.67	9.21	10.91	9.46	11.44	13.11	11.28	11.59	12.98	9.79	9.04
51	2.66	3.37	5.65	7.35	6.67	8.22	9.55	7.72	9.86	11.25	7.69	7.31
52	6.23	3.44	9.22	10.92	10.24	11.79	13.12	11.29	12.8	14.19	11	10.25
53	5.03	4.35	6.59	8.51	8.97	10.15	10.16	8.67	11.71	14.01	9.99	10.09
54	7.34	5.47	8.9	10.82	11.28	12.46	12.47	10.98	14.02	15.46	12.27	11.52
55	6.58	3.03	9.57	11.27	9.82	11.8	13.47	11.64	11.95	13.34	10.15	9.4
56	7	3.45	9.99	11.69	10.24	12.222	13.89	12.06	12.37	13.76	10.57	9.82
57	8.59	7.91	9.31	11.23	11.53	12.87	12.88	11.39	13.66	15.05	11.86	11.11
58	10.29	8.22	10.81	12.11	11.53	13.51	13.16	11.67	13.66	15.05	11.86	11.11
59	9.96	7.89	9.89	11.19	11.2	12.76	12.24	10.75	13.33	14.72	11.53	10.78
60	13.67	11.6	14.19	15.49	14.91	16.89	16.54	15.05	17.04	18.43	15.24	14.49

TABLE A.1 TRAVEL DISTANCE BETWEEN ZONAL NODES (in km)

	Destination →											
Origin	37	38	39	40	41	42	43	44	45	46	47	48
1	7.78	8.41	11.08	12.45	13.33	8.56	7.87	6.27	2.18	3.7	5.96	12.62
2	6.54	8.05	9.86	11.07	11.95	7.72	8.02	6.42	1.55	3.85	5.57	12.32
3	6.79	7.13	8.43	8.85	9.73	7.6	8.41	9.17	3.48	6.6	7.62	14.37
4	6.82	8.33	10.14	10.92	11.8	8	8.81	7.4	1.48	4.83	5.85	12.6
5	6.47	7.23	9.04	9.78	10.66	7.65	8.46	8.1	2.9	5.53	6.55	13.21
6	5.74	6.08	7.89	8.63	9.51	6.55	7.36	8.1	3.22	5.53	6.34	12.67
7	8.58	8.8	10.05	10.47	11.35	9.27	10.08	9.82	2.89	7.25	8.96	15.71
8	8.15	8.37	9.62	8.78	9.86	8.84	9.65	10.98	4.88	8.41	9.43	16.18
9	9.55	9.77	11.02	8.1	9.18	10.24	11.05	12.38	6.28	9.81	10.83	17.58
10	11.51	11.73	12.98	10.06	11.14	12.2	13.01	14.34	8.24	11.77	12.79	19.54
11	11.94	12.16	13.41	10.49	11.57	12.63	13.44	14.77	8.67	12.2	13.22	19.97
12	13.01	13.23	14.48	11.56	12.64	13.7	14.51	15.84	9.74	13.27	14.29	21.04
13	9.89	10.11	11.36	8.19	9.27	10.58	11.39	13.39	8.57	11.67	11.96	18.29
14	12.11	12.33	13.58	10.41	11.49	12.8	13.61	15.61	10.07	13.6	14.18	20.51
15	8.48	6.75	9.42	13.21	13.79	6.9	6.21	4.61	6.19	2.47	3.78	9.06
16	8.97	7.24	9.91	13.7	14.28	7.39	6.7	5.1	4.88	2.53	4.89	11.13
17	7.8	6.07	8.74	12.53	13.11	6.22	5.53	3.93	4.68	1.36	3.92	10.28
18	7.85	6.56	9.23	12.72	13.6	6.71	6.02	4.42	4.95	1.85	2.64	9.39
19	6.69	4.96	7.63	11.42	12	5.11	4.42	2.82	6.69	2.11	1.19	7.31
20	7.32	5.91	8.58	12.37	12.95	6.06	5.37	3.77	5.35	2.46	1.26	8.01
21	6.42	6.51	9.18	12.11	12.99	6.66	5.97	4.37	6.79	3.66	1.2	7.53
22	5.59	7.1	8.91	10.26	11.14	6.77	7.58	6.39	4.41	4	3.88	10.63
23	5.02	6.81	8.62	10.4	11.28	6.2	7.01	6.62	4.62	5	3.77	10.1
24	1.81	3.55	5.62	8.09	8.67	2.99	3.8	5.8	6.53	7.07	6.87	11.6
25	2.58	4.57	6.77	9.63	10.21	3.08	4.67	4.56	7.07	5.86	5.04	9.12
26	6.69	6	8.67	12.46	13.04	6.15	5.46	3.3	8.37	3.79	4.59	6.29
27	3.5	4.08	5.89	7.67	8.55	4.55	5.36	7.36	5.27	6.57	5.64	11.97
28	4.94	4.73	5.96	5.88	6.76	5.63	6.17	8.17	6.28	7.87	7.56	13.81
29	2.79	2.55	3.81	6.18	6.76	3.02	3.83	5.83	7.39	7.1	7.9	13.13
30	4.83	0.37	5.6	4.41	4.99	5.41	5.81	7.81	7.74	9.08	9.2	14.68
31	7.14	7.36	8.61	6.96	8.04	7.83	8.64	10.64	7.09	8.92	9.21	15.54
32	5.31	5.53	6.78	6.5	7.38	6	6.81	8.81	5.6	7.43	7.72	14.05
33	5.98	4.52	5.75	2.91	3.99	6.01	5.96	7.96	9.28	9.23	10.03	15.65
34	7.37	5.91	7.14	2.67	3.25	7.4	7.35	9.35	11.6	10.62	11.42	17.04
35	4.18	2.72	3.95	4.62	5.2	4.21	4.16	6.16	8.41	7.43	8.23	13.85
36	4.36	1.97	2.18	6.99	7.57	3.46	3.41	5.41	10.2	6.68	7.48	13.1
37	0	2.99	5.06	7.92	8.5	2.36	3.17	5.17	7.58	6.44	7.24	11.7
38	2.99	0	2.67	6.46	7.04	1.49	1.44	3.44	9.09	4.71	5.51	11.13
39	5.06	2.67	0	7.69	8.27	4.16	4.11	6.11	10.9	7.38	8.18	13.8
40	7.92	6.46	7.69	0	2.28	7.95	7.9	9.9	11.83	11.17	11.97	17.59
41	8.5	7.04	8.27	2.28	0	8.53	8.48	10.48	12.73	11.75	12.55	18.17
42	2.36	1.49	4.16	7.95	8.53	0	1.59	3.59	8.76	4.86	5.66	11.28
43	3.17	1.44	4.11	7.9	8.48	1.59	0	2.9	8.75	4.17	4.97	10.59
44	5.17	3.44	6.11	9.9	10.48	3.59	2.9	0	7.15	2.57	3.37	8.43
45	7.58	9.09	10.9	11.85	12.73	8.76	8.75	7.15	0	4.58	6.61	13.36
46	6.44	4.71	7.38	11.17	11.75	4.86	4.17	2.57	4.58	0	2.66	8.92
47	7.24	5.51	8.18	11.97	12.55	5.66	4.97	3.37	6.61	2.66	0	6.75
48	11.7	11.17	13.84	17.63	18.21	11.32	10.63	8.47	13.36	8.96	6.75	0
49	5.6	7.59	9.79	12.65	13.23	6.1	7.48	4.64	7.51	5.12	2.71	8.06
50	8.8	7.07	9.74	13.53	14.11	7.22	6.53	4.37	9.44	4.86	5.66	5.96
51	5.24	5.34	8.01	11.8	12.38	5.49	4.8	1.9	8.11	3.53	4.33	8.38
52	8.81	8.28	10.95	14.74	15.32	8.43	7.74	5.58	10.65	6.07	5.31	2.89
53	7.61	8.12	10.79	14.26	15.14	8.11	7.58	5.98	8.94	5.27	2.61	5.97
54	9.92	9.55	12.22	16.01	16.59	9.7	9.01	7.41	10.65	6.7	4.04	3.66
55	9.16	7.43	10.1	13.89	14.47	7.58	6.89	4.73	9.8	5.22	6.02	7.16
56	9.58	7.85	10.52	14.31	14.89	8	7.31	5.15	10.22	5.64	6.44	5.18
57	10.87	9.14	11.81	15.6	16.18	9.29	8.6	7	10.52	6.29	4.09	3.88
58	10.87	9.14	11.81	15.6	16.18	9.29	8.6	7	8.82	4.86	6.17	10.04
59	10.54	8.81	11.48	15.27	15.85	8.96	8.27	6.67	7.4	4.53	5.84	9.71
60	14.25	12.52	15.19	18.98	19.56	12.67	11.98	10.38	11.84	8.24	9.55	13.42

TABLE A.1 TRAVEL DISTANCE BETWEEN ZONAL NODES (in km)

	Destination ➔											
Origin	49	50	51	52	53	54	55	56	57	58	59	60
1	7.36	8.56	7.23	9.73	8.57	10	8.92	9.34	9.64	7.94	5.84	10.28
2	6.47	8.71	7.38	9.88	7.9	9.61	9.07	9.49	9.66	8.09	6.9	11.34
3	7.65	11.46	8.75	12.32	9.08	11.39	11.82	12.24	11.71	10.84	9.92	14.22
4	6.75	9.69	8.18	10.86	8.18	9.89	10.05	10.47	9.94	9.07	7.99	12.43
5	6.4	10.39	7.83	11.23	7.83	10.14	10.75	11.17	10.55	9.77	8.85	13.15
6	5.86	10.39	7.29	10.69	7.29	9.6	10.75	11.17	10.01	9.77	8.85	13.15
7	9.31	12.11	10.73	13.28	10.73	13	12.47	12.89	13.05	11.49	9.73	14.17
8	9.74	13.27	10.56	14.13	11.17	13.47	13.63	14.05	13.52	12.65	11.72	16.03
9	11.14	14.67	11.96	15.53	12.57	14.87	15.03	15.45	14.92	14.05	13.12	17.43
10	13.1	16.63	13.92	17.49	14.53	16.83	16.99	17.41	16.88	16.01	15.08	19.39
11	13.53	17.06	14.35	17.92	14.96	17.26	17.42	17.84	17.31	16.44	15.51	19.82
12	14.6	18.13	15.42	18.99	16.03	18.33	18.49	18.91	18.38	1751	16.58	20.89
13	11.48	15.86	12.3	15.87	12.91	15.22	16.22	16.64	15.63	15.91	14.99	19.29
14	13.7	18.08	14.52	18.09	15.13	17.44	18.44	18.86	17.85	17.84	16.91	21.22
15	6.29	6.9	5.57	8.07	6.39	7.82	7.26	7.68	5.18	3.48	2.37	6.86
16	6.49	7.39	6.06	8.56	7.5	8.93	7.75	8.17	7.25	555	3.21	7.65
17	5.32	6.22	4.89	739	6.53	7.96	6.58	7	6.78	5.08	3.73	8.17
18	4.04	6.71	5.38	7.88	5.25	6.68	7.07	7.49	6.73	6.09	5.54	9.47
19	3.7	5.11	3.78	5.87	3.17	4.6	5.47	5.89	4.65	5.62	5.29	9
20	3.1	6.06	4.73	6.57	3.87	5.3	6.42	6.84	5.35	6.57	6.15	9.95
21	1.51	6.66	3.56	6.09	2.87	4.82	7.02	7.1	4.87	7.17	6.84	10.55
22	4.19	8.68	5.62	9.02	5.62	7.92	9.04	9.46	7.97	8.24	7.69	11.62
23	3.29	8.28	4.72	8.12	4.72	7.03	8.64	8.95	744	9.24	8.69	12.62
24	5.5	8.7	5.14	8.71	7.51	9.82	9.06	9.48	10.54	115	11.15	14.88
25	3.02	6.22	2.66	6.23	5.03	7.34	6.58	7	8.59	10.29	9.96	13.67
26	3.05	2.67	3.37	3.4	4.35	5.43	3.03	3.45	7.91	8.22	7.89	11.6
27	5.16	9.21	5.65	9.22	6.59	8.9	9.57	9.99	9.31	10.81	9.89	14.19
28	7.08	10.91	7.35	10.92	8.51	10.82	11.27	11.69	11.23	12.11	11.19	15.49
29	7.03	9.46	6.67	10.24	8.97	11.28	9.82	10.24	11.53	11.53	11.2	14.91
30	8.58	11.44	8.22	11.79	10.15	12.46	11.8	12.22	1287	13.51	12.76	16.89
31	8.73	13.11	9.55	13.12	10.16	12.47	13.47	13.89	12.88	13.16	12.24	16.54
32	7.24	11.28	7.72	11.29	8.67	10.98	11.64	12.06	11.39	11.67	10.75	15.05
33	10.26	11.59	9.86	12.76	11.69	14	11.95	12.37	13.66	13.66	13.33	17.04
34	12.1	12.98	11.25	14.15	14.01	15.46	13.34	13.76	15.05	15.05	14.72	18.43
35	8.05	9.79	7.69	10.96	9.99	12.27	10.15	10.57	11.86	11.86	11.53	15.24
36	9.09	9.04	7.31	10.21	10.09	11.52	9.4	9.82	11.11	11.11	10.78	14.49
37	5.6	8.8	5.24	8.81	7.61	9.92	9.16	9.58	10.87	10.87	10.54	14.25
38	7.59	7.07	5.34	8.24	8.12	9.55	7.43	7.85	9.14	9.14	8.81	12.52
39	9.79	9.74	8.01	10.91	10.79	12.22	10.1	10.52	11.81	11.81	11.48	15.19
40	12.65	13.53	11.8	14.7	14.24	16.01	13.89	14.31	15.6	15.6	15.27	18.98
41	13.23	14.11	12.38	15.28	15.14	16.59	14.47	14.89	16.18	16.18	15.85	19.56
42	6.1	7.22	5.49	8.39	8.11	9.7	7.58	8	9.29	9.29	8.96	12.67
43	7.48	6.53	4.8	7.7	7.58	9.01	6.89	7.31	8.6	8.6	8.27	11.98
44	4.64	4.37	1.9	5.54	5.98	7.41	4.73	5.15	7	7	6.67	10.38
45	7.51	9.44	8.11	10.61	8.94	10.65	9.8	10.22	10.52	8.82	7.4	11.84
46	5.12	4.86	3.53	6.03	5.27	6.7	5.22	5.64	6.29	4.86	4.53	8.24
47	2.71	5.66	4.33	5.31	2.61	4.04	6.02	6.44	4.09	6.17	5.84	9.55
48	8.06	5.96	8.38	2.89	5.97	3.66	7.2	5.18	3.88	10.04	9.71	13.42
49	0	5.16	2.74	5.17	2.7	5.01	5.52	5.94	6.26	8.68	8.35	12.06
50	5.16	0	4.89	3.07	6.15	5.1	3.08	2.02	8.75	9.29	8.96	12.67
51	2.74	4.89	0	5.49	4.75	7.06	5.25	5.67	7.96	7.96	7.63	11.34
52	5.17	3.07	5.49	0	3.58	2.53	4.31	2.29	5.68	10.42	10.09	13.8
53	2.7	6.15	4.75	3.58	0	2.31	6.53	5.37	3.56	8.3	7.97	11.68
54	5.01	5.1	7.06	2.53	2.31	0	6.34	4.32	4.35	9.09	8.76	12.47
55	5.52	3.08	5.25	4.27	6.53	6.3	0	3.86	9.65	9.65	9.32	13.03
56	5.94	2.02	5.67	2.29	5.37	4.32	3.86	0	7.97	10.07	9.74	13.45
57	6.26	8.75	7.96	5.68	3.56	4.35	9.65	7.97	0	6.16	5.83	9.54
58	8.68	9.29	7.96	10.42	8.3	9.09	9.65	10.07	6.16	0	4.13	4.09
59	8.35	8.96	7.63	10.09	7.97	8.76	9.32	9.74	5.83	4.13	0	5.86
60	12.06	12.67	11.34	13.8	11.68	12.47	13.03	13.45	9.54	4.09	5.86	0

TABLE A.2 O-D TRIPS BETWEEN ZONAL NODES (In Jan, 2006)

	Destination →											
Origin	1	2	3	4	5	6	7	8	9	10	11	12
1	1	0	2	0	2	0	0	0	15	0	5	0
2	0	0	0	0	0	0	0	0	1	0	0	0
3	2	0	0	1	0	1	0	0	3	0	5	0
4	0	0	1	0	0	0	0	0	1	1	2	0
5	21	0	0	0	1	0	0	0	2	0	10	0
6	13	1	0	6	0	0	0	0	0	0	4	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	26	12	10	1	0	0	2	0	1	0	41	0
10	0	0	1	0	0	0	0	0	0	0	2	0
11	0	1	1	0	5	0	0	0	27	0	1	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	5	2	4	1	6	0	0	0	3	0	2	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	5	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	5	1	0	0	0	0	0	0	2	0	11	0
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	1	0	1	0	0	0	0	0	6	0	1	0
22	2	0	0	0	1	0	0	0	5	0	3	0
23	5	2	3	2	2	0	0	0	6	0	2	0
24	4	0	2	0	1	0	0	0	0	0	1	0
25	2	1	0	0	0	1	0	0	1	0	1	0
26	4	1	5	1	25	5	0	0	2	0	1	0
27	3	2	1	0	0	0	0	0	1	9	4	0
28	2	0	2	0	1	0	0	0	0	0	2	0
29	6	1	4	2	9	0	0	0	2	0	7	0
30	3	0	1	0	0	0	0	0	0	0	4	0
31	0	0	0	0	0	0	0	0	0	0	1	0
32	0	1	0	0	0	0	0	0	1	0	1	0
33	8	0	2	3	0	1	0	0	1	0	14	0
34	0	0	0	0	0	0	0	0	0	0	0	0
35	10	5	4	2	2	0	0	0	10	0	15	0
36	2	0	1	0	1	0	0	0	0	0	1	0
37	7	3	7	0	3	1	0	0	5	0	10	0
38	11	8	0	1	24	0	0	0	1	0	4	0
39	6	1	1	0	0	0	0	0	0	0	0	0
40	1	0	0	0	0	0	0	0	2	0	0	0
41	10	0	3	1	1	0	0	0	7	1	9	0
42	2	1	0	1	0	0	0	0	0	0	1	0
43	3	1	3	2	0	0	0	0	4	0	3	0
44	3	0	2	0	1	0	0	0	2	0	2	0
45	0	0	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0	0	0	0	0	0
47	0	0	0	0	0	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0	0	0
49	0	19	0	0	0	0	0	0	1	0	1	0
50	0	0	0	0	0	0	0	0	0	0	0	0
51	1	0	1	1	0	0	0	0	0	0	0	0
52	2	0	1	0	0	0	0	0	0	0	3	0
53	0	0	0	0	0	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0	0	1	0	0	0
55	1	1	0	0	9	0	0	0	1	0	1	0
56	32	3	7	0	15	0	0	0	18	0	18	0
57	1	0	0	0	0	0	0	0	9	0	1	0
58	0	0	0	0	14	0	0	0	1	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0

TABLE A.2 O-D TRIPS BETWEEN ZONAL NODES (In Jan, 2006)

	Destination →											
Origin	13	14	15	16	17	18	19	20	21	22	23	24
1	5	0	0	0	17	2	0	0	3	13	6	4
2	0	0	0	0	0	1	0	0	0	0	1	0
3	4	0	0	0	3	1	0	0	1	0	3	0
4	1	0	0	0	1	0	0	0	0	0	0	0
5	11	1	0	2	6	1	0	0	0	0	0	0
6	4	0	2	1	0	3	0	0	0	0	8	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	32	0	2	5	11	11	0	0	12	1	11	15
10	0	0	0	0	29	2	0	0	0	0	4	0
11	2	0	0	0	5	4	0	0	0	0	0	3
12	0	0	0	0	0	0	0	0	0	0	0	0
13	1	0	0	0	11	22	0	1	2	2	5	0
14	0	0	0	0	0	1	0	0	0	0	1	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	1	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	17	0	0	0	2	0	0	0	0	0	22	2
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	8	0	0	1	2	0	0	0	0	0	2	0
22	3	0	0	0	1	0	0	0	0	0	0	0
23	7	1	0	2	18	18	0	0	1	0	4	6
24	1	0	0	1	20	9	0	0	0	2	1	0
25	2	0	0	0	4	2	0	1	1	1	0	0
26	3	0	0	0	16	5	0	0	0	1	3	1
27	6	0	0	2	8	3	0	0	1	0	2	0
28	2	0	0	1	3	0	0	0	0	0	0	1
29	9	0	1	0	31	18	0	0	3	3	2	0
30	6	0	0	3	14	11	0	0	1	1	1	1
31	0	0	0	0	5	0	0	0	0	0	0	0
32	1	0	0	0	2	0	0	0	0	0	0	0
33	5	2	0	1	3	0	1	0	1	1	6	1
34	0	0	0	0	0	0	0	0	0	0	0	0
35	10	1	0	3	72	22	0	0	1	1	13	3
36	0	0	0	0	0	1	0	0	0	0	1	0
37	3	1	0	3	25	8	3	0	1	7	19	1
38	4	0	0	5	44	15	0	0	0	0	7	1
39	2	1	0	3	10	7	0	0	0	1	7	1
40	0	0	0	0	1	2	0	1	0	0	0	0
41	4	0	0	0	11	3	0	0	2	0	7	3
42	0	0	0	0	7	1	0	0	0	0	1	2
43	13	0	0	1	22	7	0	0	0	2	2	0
44	3	0	0	0	4	4	0	0	1	1	2	0
45	0	0	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0	0	0	0	0	0
47	0	0	0	0	2	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0	0	0
49	1	0	0	0	2	1	0	0	1	0	0	0
50	0	0	0	0	1	0	0	0	0	0	1	0
51	0	0	0	0	0	4	0	0	0	1	1	0
52	0	0	0	0	1	0	0	0	1	0	1	0
53	0	0	0	0	1	1	0	0	1	0	0	0
54	0	0	0	0	0	0	0	0	0	0	0	0
55	0	0	0	0	2	4	0	0	1	0	1	0
56	11	0	1	1	3	13	0	0	16	23	23	5
57	1	0	0	0	1	0	0	0	0	0	0	2
58	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0

TABLE A.2 O-D TRIPS BETWEEN ZONAL NODES (In Jan, 2006)

Destination →												
Origin	25	26	27	28	29	30	31	32	33	34	35	36
1	1	2	4	0	0	1	0	0	4	0	0	0
2	1	2	3	0	0	0	0	1	0	0	0	0
3	0	1	2	0	1	0	1	0	1	0	2	0
4	0	4	0	0	0	0	0	0	3	0	0	0
5	0	4	2	0	21	0	0	0	0	0	1	0
6	5	10	0	0	6	0	0	0	2	0	1	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	4	34	17	2	4	3	0	0	3	0	2	0
10	0	3	3	1	0	0	0	1	0	0	0	0
11	0	4	3	3	2	1	2	0	1	0	2	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	2	7	10	1	0	0	0	0	0	0	2	0
14	0	0	0	1	0	0	0	0	0	0	0	0
15	0	1	0	0	0	0	0	0	0	0	0	0
16	0	0	0	1	0	1	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	1	0
18	0	6	0	1	1	2	0	0	1	0	2	0
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	0	3	1	1	0	0	3	0	2	0	0	0
22	1	3	1	0	1	2	1	0	1	0	0	0
23	0	5	6	0	0	3	0	0	2	0	6	0
24	0	2	1	0	0	2	0	0	0	0	0	0
25	0	0	0	0	1	0	0	0	0	0	0	0
26	0	6	3	1	1	2	0	0	5	0	2	3
27	0	4	2	1	2	0	0	0	0	0	2	3
28	0	0	3	0	0	0	0	0	0	0	1	0
29	1	8	0	2	0	0	0	0	1	0	0	0
30	0	8	4	0	0	0	0	0	1	0	0	0
31	0	0	0	2	1	0	0	0	0	0	0	0
32	2	0	0	0	0	0	0	2	0	0	0	0
33	2	6	4	3	2	1	0	0	0	0	2	5
34	0	0	0	0	0	0	0	0	0	0	0	0
35	0	24	4	7	2	1	0	2	0	0	2	0
36	0	3	5	1	0	0	0	0	4	0	0	0
37	0	7	6	1	4	3	0	0	2	0	5	0
38	1	2	4	2	1	2	0	0	1	0	3	0
39	1	0	0	1	0	0	0	0	0	0	0	0
40	1	1	0	0	0	0	0	0	0	0	1	0
41	0	4	5	2	4	1	0	1	0	0	6	0
42	0	0	0	0	0	0	0	0	0	0	0	0
43	0	1	12	19	0	1	0	0	0	0	0	0
44	1	0	1	3	0	0	0	0	0	0	1	0
45	0	0	0	0	0	0	0	0	0	0	0	0
46	0	1	0	0	0	0	0	0	0	0	0	0
47	0	0	0	0	0	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0	0	0
49	0	0	2	0	0	0	0	1	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0	0	0
51	0	0	0	1	0	0	0	0	0	0	0	0
52	0	3	1	1	0	0	1	0	0	0	0	0
53	0	0	0	0	1	0	0	0	0	0	0	0
54	0	0	1	0	0	0	0	0	0	0	0	0
55	0	0	0	1	0	1	0	0	0	0	0	0
56	11	21	5	1	23	3	0	0	13	0	12	0
57	0	1	0	0	1	0	0	0	0	0	0	0
58	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0

TABLE A.2 O-D TRIPS BETWEEN ZONAL NODES (In Jan, 2006)

	Destination ➔											
Origin	37	38	39	40	41	42	43	44	45	46	47	48
1	7	2	0	1	60	0	2	2	0	0	0	0
2	2	0	0	0	40	0	0	3	0	0	0	0
3	3	3	0	0	43	0	1	1	0	0	0	0
4	0	0	0	0	19	0	0	0	0	0	0	0
5	5	25	0	1	47	4	3	0	0	0	0	0
6	10	4	0	0	2	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	23	12	0	20	103	5	4	1	1	0	2	0
10	0	0	0	0	29	0	1	5	0	0	0	0
11	3	1	0	1	14	0	0	1	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	4	8	0	0	40	0	14	3	0	0	0	0
14	0	0	0	0	6	0	0	0	0	0	0	0
15	1	0	0	0	1	0	0	0	0	0	0	0
16	0	0	0	0	2	0	0	1	0	0	0	0
17	0	0	0	0	1	0	0	0	0	0	0	0
18	4	1	0	0	35	1	0	1	0	0	1	1
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	6	0	0	0	27	1	0	3	0	4	0	0
22	11	2	0	1	8	0	2	2	0	0	0	0
23	15	7	1	1	108	0	2	4	0	0	0	0
24	5	0	0	0	49	0	0	0	0	3	0	1
25	0	0	0	1	10	0	0	1	0	0	0	1
26	2	8	0	0	42	0	0	1	0	0	0	0
27	4	0	0	0	61	0	1	4	1	0	0	0
28	4	0	3	1	18	0	1	0	0	0	0	0
29	2	0	2	4	59	0	0	5	0	0	0	1
30	4	0	0	1	27	1	1	6	0	0	1	0
31	1	0	0	0	3	0	0	0	0	0	0	0
32	2	0	3	0	5	0	1	0	0	0	0	0
33	8	0	0	0	55	3	0	4	0	0	1	0
34	0	0	0	0	0	0	0	0	0	0	0	0
35	14	0	0	2	131	2	0	5	0	1	1	0
36	1	0	0	0	12	0	0	2	0	0	0	0
37	5	0	1	0	65	1	6	7	0	0	0	0
38	1	0	0	3	55	1	0	2	0	2	0	0
39	3	1	0	0	27	1	0	2	0	0	0	0
40	1	0	0	0	6	0	0	2	0	0	0	0
41	3	3	0	1	8	0	1	2	0	1	0	0
42	0	0	0	0	14	0	0	0	0	0	0	0
43	5	0	0	0	26	1	0	1	0	0	0	1
44	2	0	0	0	37	0	0	1	0	0	0	0
45	0	0	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	4	0	0	0	0	0	0	0
47	0	0	0	0	7	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0	0	0
49	1	0	0	0	11	1	0	0	0	0	0	0
50	0	0	0	1	5	0	0	0	0	0	0	0
51	0	0	0	0	5	1	0	0	0	0	0	0
52	3	0	0	0	13	0	0	0	0	0	0	0
53	0	0	0	0	3	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0	0	0	0	0	0
55	1	0	0	0	12	0	0	0	0	0	0	0
56	38	10	1	8	26	27	5	13	0	0	1	0
57	2	0	0	0	5	0	0	0	0	0	0	0
58	0	0	0	0	0	1	0	0	0	0	0	0
59	0	0	0	0	2	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0

TABLE A.2 O-D TRIPS BETWEEN ZONAL NODES (In Jan, 2006)

	Destination ➔											
Origin	49	50	51	52	53	54	55	56	57	58	59	60
1	1	0	0	2	0	0	1	1	1	0	0	0
2	23	0	0	0	0	0	1	2	0	0	0	0
3	0	0	0	1	0	0	1	2	0	0	0	0
4	0	0	0	0	0	0	0	0	0	1	0	0
5	0	0	0	8	0	0	12	10	2	1	0	0
6	1	0	0	1	2	0	18	7	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	2	5	13	2	0	28	35	16	0	7	15
10	0	0	1	0	0	0	0	3	1	0	0	0
11	0	0	1	1	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	2	1	0	0	12	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	6	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	1	0	0	2	0	0	5	1	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	0	1	0	14	0	0	1	6	0	0	0	0
22	0	0	1	0	0	0	3	35	2	0	0	0
23	1	1	0	2	0	0	2	15	1	0	0	0
24	0	0	0	1	0	0	1	2	1	0	0	0
25	0	0	0	0	0	1	0	2	1	0	0	1
26	0	0	0	1	0	0	1	0	2	0	0	0
27	0	0	0	0	0	1	2	6	0	0	0	0
28	0	0	1	1	0	0	0	2	1	0	0	0
29	1	1	1	0	2	0	1	5	5	0	0	1
30	1	2	2	1	1	0	1	0	0	0	1	0
31	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	1	0	2	5	0	3	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0
35	0	2	0	1	3	0	3	4	3	2	1	0
36	0	0	0	0	0	0	4	1	0	0	0	0
37	0	0	0	4	1	0	3	17	2	0	3	0
38	0	0	0	0	0	0	3	7	3	1	0	0
39	0	0	0	0	0	0	0	2	1	1	0	0
40	1	0	0	0	0	0	0	0	0	0	0	0
41	0	0	0	4	2	0	0	3	2	0	0	0
42	0	0	0	0	0	0	0	3	0	0	0	0
43	0	0	0	1	0	0	0	0	0	0	0	0
44	0	0	0	1	0	0	1	5	0	0	0	0
45	0	0	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0	1	0	0	0	0
47	0	0	0	0	0	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0	0	0
49	0	0	0	0	1	0	2	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0	0	0
51	0	0	0	0	0	0	0	1	0	0	0	0
52	0	0	0	0	0	0	1	0	0	0	0	0
53	0	0	0	0	0	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0	0	0	0	0	0
55	0	0	1	1	0	0	0	0	0	0	0	0
56	3	7	1	8	3	0	14	2	9	0	10	0
57	0	0	0	1	0	0	0	0	0	0	0	0
58	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	0	1	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	1	0	0	0