Chapter 1

Introduction

All passengers travel at the hour most convenient to them. But it is not always possible to find a flight at the right time to fly them to their destination. In the case where service in any one time period is insufficient to meet air travel demanded, it may be expected that some unfilled demand passengers will either delay their flight or will advance it, thus adding to the effective demand of the adjoining time periods. Gunn (1964) defined this effect as “persistence of demand.”

The failure to satisfy demand at the right time results in two effects.

1. Few potential passengers who have a fairly inelastic demand for the trip in terms of its inconvenience; the majority of these travelers might be expected to advance or delay their departure from the ideal hour to an available hour if necessary.
2. A fair number of passengers have a fixed schedule for the day and might not be willing to depart at other than the convenient hour, choosing instead to go by alternative means or not travel at all.

The obvious alternate means of travel is a rental car. It takes a lot more time than flight, but it is readily available at any given time. This brings us to think of an airline system that will work in a similar fashion; A system that can be named an “Air Taxi System.” In the past, only the rich and famous had access to personal jets designed to whisk travelers from city to city without the inconvenience of crowded major airports. Now, however, with NASA’s support and the work of several companies determined to redefine personal air transport, flying direct to nearly any city from the closest local airport may soon become a viable option for everyone.

And that's where SATS comes in. The Small Aircraft Transportation System—currently being developed by NASA, and nearly 60 other aviation-related companies, agencies, and universities could, if its proponents prevail, revolutionize the way we travel. They intend to employ a new generation of inexpensive small business jets and an innovative computerized flight control network. This would help the air taxi companies provide direct service from and to any of the more than 3,400 public-use airports that pepper the national landscape but that have been unusable for commercial flights because they lack the staff and equipment necessary to handle heavy traffic, as well as takeoffs and landings in inclement weather (Scott 2002).

The SATS concept of operations utilizes small aircraft for personal and business transportation, for on-demand, point-to-point direct travel between smaller regional, reliever, general aviation and other landing facilities, including heliports. The architecture contemplates near-all-weather access to any landing facilities in the U.S. The systems would leverage Internet communications technologies for travel planning, scheduling, and optimizing and would minimize user uncertainty regarding destination services, including intermodal connectivity. SATS research is intended to create the possibility of using landing facilities that would not require control towers or radar surveillance. SATS architecture would be created to operate within the National Airspace System (NAS), initially between about 3400 existing public-use landing facilities. A total of over 9000 landing facilities serve the vast numbers of communities in the U.S; ultimately, essentially all of these facilities could employ SATS operating capabilities. The SATS aircraft include twin turbofan-powered, four- to six-place pressurized aircraft with revolutionary safety and affordability. There are also many new single-engine aircraft entering the fleet, also with safety features and cost previously unimagined. These new aircraft will possess near-all-weather operating capabilities and will be compatible with the modernization of the National Airspace System, including free flight. The aircraft will incorporate state-of-the-art advancements in avionics, airframes, engines, and advanced pilot training technologies (http://sats.nasa.gov/).

It will be close to a scenario where a passenger would simply logon to an air taxi reservation system and would book an aircraft and a pilot for himself. The passenger tells the pilot he wants to fly to a distant suburb, and he's on his way—cruising at 400 miles per hour and improving average door-to-door speed to 200 miles per hour.

This would mean a virtual highway in air space leading to a vast network. The network would be served by small aircraft flying from one city to another loading and unloading passengers. Such a large network having dynamic demand will have many issues to resolve before successfully launching a Small Aircraft Transportation System. One of the most important problems to solve is scheduling of aircraft for such a stochastic demand flow.

The objective of the research is to study a given set of airports with dynamic demand and known aircraft type. The major task will be to analyze the flow of passengers between each origin-destination pair and then schedule flights. The research will be to develop a schedule for a fixed set of airports with dynamic demand and known type of aircraft. The main objective is to maximize demand satisfaction. The study will also analyze the number of aircraft required for a given set of airports and find a method to schedule them.

Chapter 2

Literature Review

2.1 Flight Scheduling

The flight schedule is the central element of an airline's planning process, aimed at optimizing the deployment of the airline's resources in order to meet demands and maximize profits (Etschmaier 1984). The schedule construction phase takes into consideration only the aspects of primary importance. It provides only a rough first schedule which requires considerable modifications and improvement to become both operationally feasible and economically desirable.

A completed and adopted airline schedule is a working obligation for all those employed in the air carrier's services. Passengers are interested in the greatest flight frequency, departure times, short waiting time. The air carrier is interested in an airline schedule that results in a good airplane and good utilization of the existing transportation capacities. Certain passenger requests regarding the airline schedule inevitably conflict with the carrier's requests. The airline schedule design must reflect the best possible way to reconcile theses conflicting requirements.

The approaches taken in the schedule construction process can be divided into direct approaches and stepwise ones (Etschmaier 1984). The direct approaches use some heuristic procedure for composing a schedule, flight by flight. While some old models were entirely computer based, models currently in use provide a man-machine interactive environment in which the selection of flights is done by the planner.

Stepwise approaches start by selecting routes that are to be served and determining the frequency of service on each route. This step is called frequency planning. The second step determines departure times on the basis of the time-of-the-day variability of demand and of the possible connections of flights to other airlines. In the third step departure times are checked for operational feasibility. Aircraft rotation plans are developed to determine the number of aircraft required for executing the schedule. Also, changes may be identified which could lead to a reduction of the number of aircraft required.

Which approach is best for solving the aircraft scheduling problem for a particular airline depends on the structural characteristics of the airline, most importantly the route structure (linear vs. hub-and-spoke networks) and the market structure (density, volume and elasticity of demand). Various techniques used to solve these types of problems are:

- Time-of-day models
- Frequency planning models
- Aircraft Rotation models
- Direct Approaches

2.2 Time-of-Day models

A demand profile may be available that indicates how many people would like to fly at any particular interval of time. If only non-stop traffic is considered, then a given set of flights is positioned in such a way that some measure of time displacement from preferred departure times for the demand is minimized. The formulation incorporates the combinatorial features of an assignment problem. Solutions could be obtained by dynamic programming. However, for realistic situations the computational effort required is considerable. The situation is further complicated if the different aircraft types have different flying times. The air travel demanded represents the number of people per time period who would fly, provided there were a continuous supply of aircraft.

Teodorovic (1988) calls the number of passengers in a unit of time traveling from one city to another "Passenger Flow." The equations can be written

$$h_{ij}(t) = \frac{P_{ij}(t + dt) - P_{ij}(t)}{dt}$$

where $h_{ij}(t)$ is the passenger flow between city(i) and city(j) at time t,

. $P_{ij}(t)$ = Passenger arrival to time t.

Passenger flow is a value that changes over time. Changes are noticed by month, by week, by day in the week and finally by hour in the day. Monthly changes are of interest for their global view of the scope of traffic. In terms of monthly changes in passenger flows, most air routes can be divided into "business" or "tourist." To schedule a small chartered flight it is more important to understand the behavior of passenger flow by hour in the day. It is extremely important to monitor passenger flows by day in the week and particularly by hour in the day to solve the problem of determining flight frequency and departure time.

The monthly passenger demand changes and shows seasonality. Rise in demand is observed in summer and December. This monthly demand can be split into weekly depending on the traffic on weekdays and weekends. Finally, the daily demand being split hourly illustrates peaks in morning and evening.

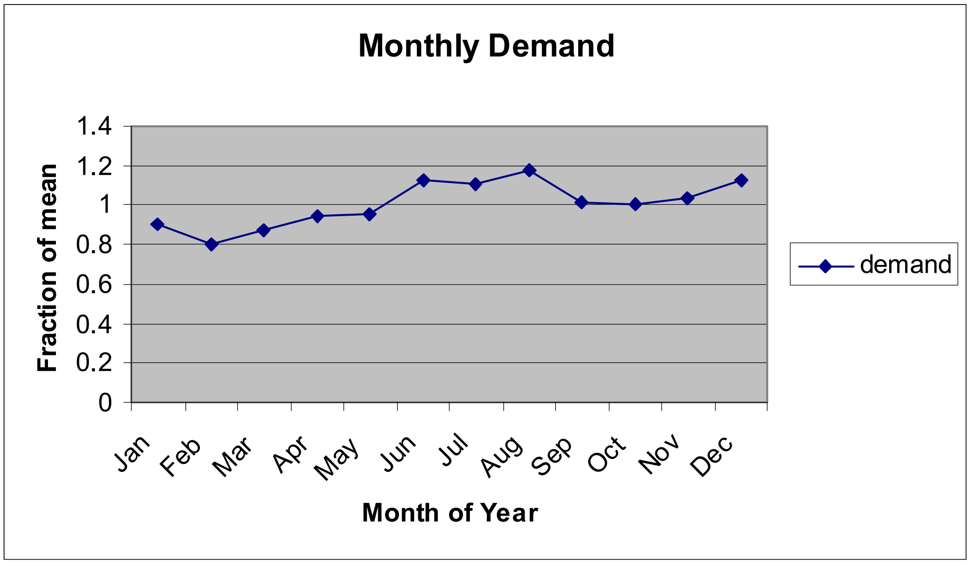


Figure 2.1 Passenger demand per month as a fraction of mean for each year

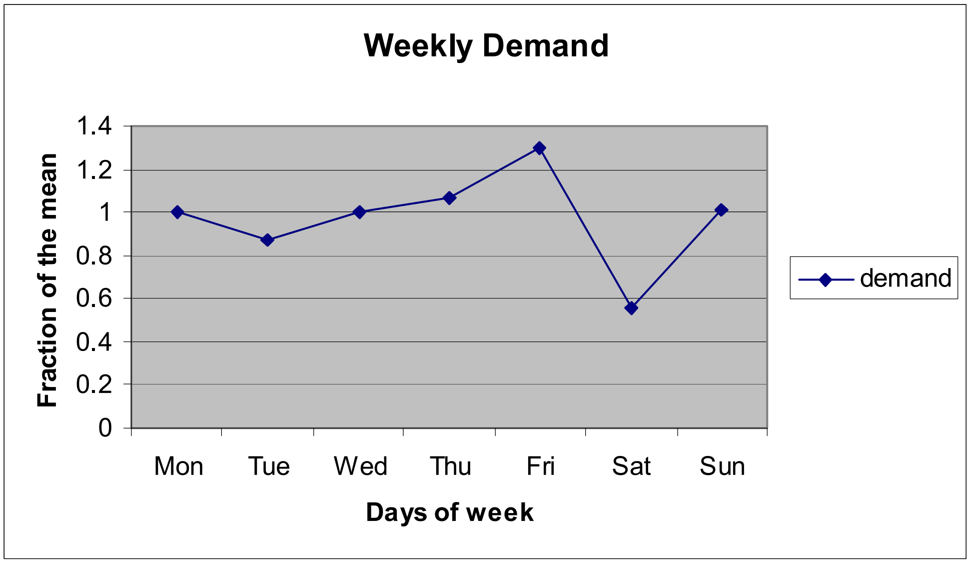


Figure 2.2 Passenger demand per day as a fraction of mean for each

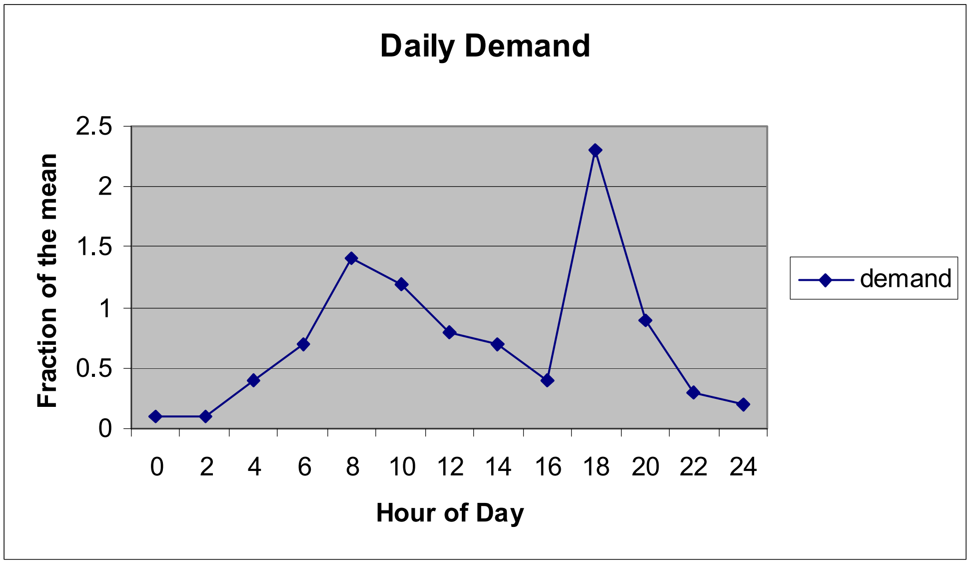


Figure 2.3 Passenger demand on hourly basis as a fraction of mean for each day

The hourly demand graph shows a functional dependence between passenger flow and the time of the day with two peaks – in the morning and in the evening. The bar graph as drawn does not measure air travel demanded on any particular route but indicates a typical, expected distribution, indeed, we would anticipate that different routes would have different air travel demand distributions. Whatever the distribution of air travel demanded, it should be apparent that the number of passengers actually traveling is the function of the timing as well as the number of flights (Teodorovic 1988).

If one aspect of routing efficiency is to be how well consumers are satisfied, we must determine a measure of consumer satisfaction and how it might vary with the distributions of flights throughout the day. Not all passengers travel at the hour most convenient to them. In the case where service in any one time period is insufficient to meet air travel demanded, it may be expected that some unfilled demand passengers will either delay their flight or will advance it, thus adding to the effective demand of the adjoining time periods. Gunn(1964) defined this effect as “persistence of demand,” and his group concluded after discussions with several airlines that persistence of demand could be estimated rather accurately, based on the next best means of transportation, taken to be private automobile driving times.

The passengers can be categorized into two groups:

a. passengers who are ready to change their time of flight, and
b. passengers who want to travel at a fixed time.

On balance, then, some – but not all – of unfilled demand will be willing to advance or delay departure should it become necessary. What proportion of passengers unable to secure a flight at their convenient hour would be willing to advance or delay departure? If the sole objective were to arrive at the destination as soon as possible after the convenient departure time, then passengers would be willing to delay departure by the difference between the transport time of the next best alternative and the length of time taken by the flight. Alternatively, the passengers might be willing to accept the inconvenience of a flight departing earlier than the optimal hour (Gunn 1964).

Surely those travelers who depart at a non-optimal hour have less consumer surplus than if they departed at the ideal hour, but there is a question as to whether average consumer surplus for inconvenienced passengers is significantly less than average consumer surplus for passengers leaving at the optimal hour. The reason is that those willing to advance or delay departure, if need be, are those with relatively more inelastic demand for the flight in terms of its inconvenience, and consequently they would have reaped higher than average consumer surplus had they been able to depart at the convenient hour.

The model presented by James C. Miller III (1966) uses the time-of-day structure of demand to solve the airline scheduling problem by linear programming for a single city pair. The model is outlined below. The cost parameters of mail and cargo are eliminated here but are considered in the paper.

Parameters:

S_i = seating capacity if plane type i(i= 1 for 2-engine jet, etc.)

C_i = flight cost for plane type I, on flight of 775 miles.

K_i = daily equipment cost plus daily cost of capital for one aircraft type i.

D_i = average air travel demanded during time period t

Variables:

X^t_i = number of i-engine jet flights per day during time period t.

E^t = number of empty seats flown per day during time period t.

Y = total number of passengers flown per day.

Z_i = number of i-engine jet aircraft.

C_p = per passenger revenue

$$\sum_{t=1}^{8}[C_p(\sum_{i=1}^{3} S_i X_i^t - E_t)] -$$

$$\{\sum_{i=1}^{3} C_i X_i^t + 0.003509*775*(\sum_{i=1}^{3} S_i X_i^t - E_t) + 0.019603*775*(\sum_{i=1}^{3} S_i X_i^t - E_t) + \sum_{i=1}^{3} K_i Z_i\} +$$

$$20Y$$

$subject\_to$

$$(\sum_{i=1}^{3} S_i X_i^t - E_t)] \leq 0.5[D_{t-1} - (\sum_{i=1}^{3} S_i X_i^{t-1} - E_{t-1})] + D_t + 0.25[D_{t+1} - \sum_{i=1}^{3} S_i X_i^{t+1} - E_{t+1})]$$

$$X_i^t \leq Z_i$$

$$X_i^1, X_i^2, \ldots.., X_i^8 \text{ int}eger$$

$$Y = \sum_{t=1}^{8}(\sum_{i=1}^{3} S_i X_i^t - E_t).$$

The problem is divided into 8 time periods and has 3 types of aircrafts. The objective function is the total daily revenue and also the total daily cost of air service provided. The first element inside the brackets is the revenue derived from all passengers flying at time period t, and consists of the price of the ticket (Cp) times the number of passengers flown (i.e. total seats minus the empty seats). The next bracket is total operating cost at time period t, and consists simply of cost per flight times number of flights summed over the three types of aircraft. The second element is indirect operating cost and includes nonflight investment; it consists of a cost per revenue passenger mile multiplied by revenue passenger miles (i.e. 775 miles * actual number of passengers). These two elements are summed over the eight time periods to give total daily costs. The remaining element is flight equipment cost plus cost of capital on flight investment and consists of the daily cost multiplied by the number of planes of each type in the fleet and then summed over the three types of aircraft.

The term (20Y) consists of passenger surplus ($20) multiplied by Y, the total number of daily passengers.

2.3 Frequency planning models

A frequency plan specifies the design of the network (radial vs. linear structure, nonstop and multistop routes), the frequency of service, and the type of aircraft assigned to each flight. Given a set of origin-destination markets, (with either a static or variable demand) a set of candidate routes, aircraft types, yield and operating cost functions, and operating restrictions, the objective is to find the set of supply decisions (a three-dimensional vector of frequency, routing and aircraft type) for a single period that maximizes revenues minus costs (Etschmaier 1984). The problem is modeled as a series of linear equations that are solved by mathematical programming techniques. Fixed competitive conditions and fixed market shares are generally assumed for all markets.

2.3.1 Linear Programming Models for Frequency Planning

Under the assumption of time insensitive demand and cost function, comparatively simple models can be formulated to determine optimal frequency plans. The optimization model in this case is only a part of the bigger airline objective function.

Costs are a linear function of frequency for each aircraft type on each route. The limited availability of aircraft is expressed by constraining the number of flight hours for each aircraft type. Frequencies are assumed to be continuous variables and solutions are obtained by linear programming. In addition to the aircraft capacity and utilization constraints already mentioned one may want to impose upper and lower bounds on the frequencies per aircraft type or on the total frequencies on the route. Also one may want to limit the total number of frequencies into a city. Again this may be done either for selected aircraft types or for the whole fleet. The formulation assumes symmetrical demand and frequencies in both directions of the route. If the frequencies are not symmetrical, additional constraints have to be introduced to assure continuity in aircraft movements (Etschmaier 1984).

A model of the frequency optimization type is being used for the U.S. domestic air transportation system to minimize cost. The solution was obtained by linear programming with resulting fractional frequencies. Clearly, the assumption of linearity leads to a simplistic representation of consumer behavior.

The model by Elce (1970) considers the alternate ways passengers travel from their origin to their destination by putting the passenger's choice outside the model. The origin to destination demand figures are split up and assigned in fixed proportions to the most desirable routes available. Thus the frequency optimization model deals with a fixed demand figure for every segment of each route. To a certain degree, this sounds like a self-fulfilling prophecy, since the planner will base the assignment of demand on his expectations of the frequencies.

A number of improvements to the basic formulation for frequency planning have been considered over time. The most interesting achievements permit demand functions that vary with price and level of service. The assumption of fixed market share is remote from reality in all environments except for the airlines which operate in monopoly markets. Variable demand functions are attempts at including competitive effects. A number of models have been designed to include market share – frequency share relationships. The model developed by Swan (1977) is based on the assumption that the market share of an airline on a route is proportional to the share of frequencies on that route.

Swan (1977) represents the market share – frequency share function by an S-shaped demand frequency relationship. The relationship is approximated by a convex, piece wise linear curve and is solved by linear programming. Mathaisel and DeLamotte (1983) formulate a goal programming model in which the demand is sensitive to the price and the frequency of service. The nonlinearity of the revenue function (the product of the variable demand and price function) is resolved by splitting the maximization part of the objective into two goals: maximum demand and maximum price. A minimum cost goal is also included.

A more sophisticated model was developed to find the approximate mix of vehicles, routes, schedules and terminal facilities that would satisfy intercity passenger and cargo demands at a minimal social and economic cost. The cost function includes such elements as the value of time for the passengers and cargo; the cost of owning and operating aircraft for the normal flight times and for delays due to air traffic congestion; the terminal operating costs; and the cost of dissatisfied demand. The waiting times for scheduled departures and cost of delays due to air traffic congestion are convex functions of the frequencies and thus can be approximated by piecewise linear functions. This further increases the size of the problem.

Teodorovic (1988) gives a different way to determine the flight frequency between two cities. He shows that the time difference between the actual and desired time of departure can be approximately expressed in the function of flight frequency.

2.3.2 Integer Programming Models for Frequency Planning

All the frequency models discussed so far have one common deficiency: the resulting frequencies may be, and usually are, fractional. While this may be of negligible consequence for an airline with only high frequency routes, it can pose considerable problems for most airlines. The important problems resulting from continuous frequencies are given below.

1. For large networks the rounding operation is not as simple as it seems. Even in the simplest case with only non-stop traffic and symmetrical frequencies on each route, one has to balance aircraft availability and demand satisfaction between the routes with fractional frequencies. The complexity of the rounding process increases with the number of constraints. Elce (1970) embeds the frequency model in an interactive scheme where the demand that corresponds to small fractional parts of frequencies is reassigned to other routes. The frequency model is solved with the adjusted demand value and this process is continued until it stabilizes. Soudarovich (1971) goes one step further and, after each rounding operation determines the actual traffic that can be attracted by the frequencies.
2. There is no guarantee that the non integer solutions are anywhere near optimal. The result may, for example, suggest half a frequency of a large aircraft over one frequency for a smaller aircraft. The problem is not just one of rounding the fractional part of the frequency. On routes with small frequencies it is easy to see

that, instead of adjusting the result for the fractional part, it may be preferable to assign all frequencies to another aircraft type.

The obvious answer to the difficulties arising from fractional solutions is the use of mixed integer linear programming. The passengers in such a model are considered continuous variables whereas the frequencies are restricted to integer values. The problem is that MILP algorithms lag far behind LP algorithms in computational efficiency and require large computer memory. For large scale cases one might consider a Lagrangian relaxation solution to the integer problem.

2.4 Minimizing Waiting Time of Passengers

Passenger waiting time is a crucial factor to consider. The passenger demand is never steady over the period of day. It varies from hour to hour, and it is very important that the flights are scheduled to fulfill all the demand.

Passengers can be categorized as “business” or “tourist.” The Small Aircraft Transportation System will primarily serve the business class. This class will require the service to be fast and with no schedule changes. In other words, these type of passengers will have to be served as soon as possible. The waiting time or delay per passenger will indicate the quality of the transportation service.

Ross (2000) considers passenger arrival to be a Poisson process having rate λ and denotes the time of the first event by T_1. Further, for $i > 1$, let T_i denote the time elapsed between the $(i - 1)^{st}$ and the i^{th} event. The event $\{T_1 > t\}$ takes place if and only if no arrivals occur in the interval $[0, t]$. Ross shows that T_2 is also an exponential random variable with mean $1/\lambda$, and furthermore, that T_2 is independent of T_1. This proves that T_i, $i = 1, 2, \ldots$, are independent identically distributed exponential random variables having mean 1/λ. Teodorovic (1988) explains the effect of flight frequency on airline schedule delay.

Let f(t) be the probability density function of random variables T_i and let the mean and standard deviation of these random variables be denoted by μ and σ, respectively.

We will consider a time period T where x_i is the airplane departure times during period (0, T). Assuming that the passengers will choose the flight closest to their desired departure time, all passengers who choose departure x_i will be the passengers who arrive during interval $(\frac{x_{i-1}+x_i}{2}, \frac{x_i+x_{i+1}}{2})$. The arrival of passengers will be a normal distribution with mean x_i. Let 2m be the number of passengers on every flight, i.e., m passengers during $\frac{x_{i-1}+x_i}{2}$ and m passengers during $\frac{x_i+x_{i+1}}{2}$.

We denote W_i as the absolute time deviation between the actual and desired time of i^{th} passenger.

$W_1 = T_2 + T_3 + \ldots + T_m + W_m$
$W_2 = T_3 + T_4 + \ldots + T_m + W_m$
.. ..
W_m
W_{m+1}
$W_{m+2} = T_{m+2} + W_{m+1}$

$W_{m+3} = T_{m+3} + T_{m+2} + W_{m+1}$

.. ..

$W_{2m} = T_{2m} + T_{2m-1} + T_{2m-2} \ldots + T_{m+1} + W_{m+1}$

where $W_m + W_{m+1} = T_{m+1}$

The total waiting time is

$$W = \sum_{i=1}^{2m} W_i .$$

Teodorovic proves the frequency of flight for minimizing the waiting time for period (0, T). Teodorovic (1988) gives the average schedule delay per passenger, D, as

$$D = \frac{T}{4N}$$

where, T = Total time period in minutes, and

N = Flight frequency.

Ross (2000) solves the optimization problem of minimizing the total expected wait. Ross considers a processing plant that has items arriving according to Poisson process with rate λ. At a fixed time T, all items are dispatched from the system. The problem chooses an intermediate time, $t \in (0, T)$, at which all items in the system are dispatched with minimum expected wait time. Ross considers a time t, $0 < t < T$, then the expected total wait of all items will be

$$\frac{\lambda t^2}{2} + \frac{\lambda (T - t)^2}{2} .$$

The expected number of arrivals in (0, t) is λt, and each arrival is uniformly distributed on (0, t), and hence has expected wait t/2. Thus, the expected total wait of the items arriving in (0, t) is $\lambda t^2/2$. Similar reasoning holds for arrivals in (t, T), and the above follows. To minimize this quantity, we differentiate with respect to t to obtain

$$\frac{d}{dt}\left[\lambda\frac{t^2}{2}+\lambda\frac{(T-t)^2}{2}\right]=\lambda t-\lambda(T-t)\,.$$

Equating this to 0 shows that the dispatch time that minimizes the expected total wait is t = T/2.

2.5 Minimum Number of Aircraft Required

One of the fundamental problems when designing schedules is determining the minimum number of vehicles needed to service a given schedule. This problem is relatively easy to solve on smaller transportation networks. However, for larger networks, the optimal solution must be chosen from the very large number of possible solutions.

The figure 2.4 shows a space-time diagram with 14 flights (trips) to be carried out between cities A and B, B and C, C and B, B and A. It is obvious that we can distribute the vehicles to carry out the planned flights in different ways. For example, a vehicle can take flight 4, then flight 6, flight 11 and finally flight 14. Figure 2.6 shows a network in which nodes represent planned flights (trips) to be made.

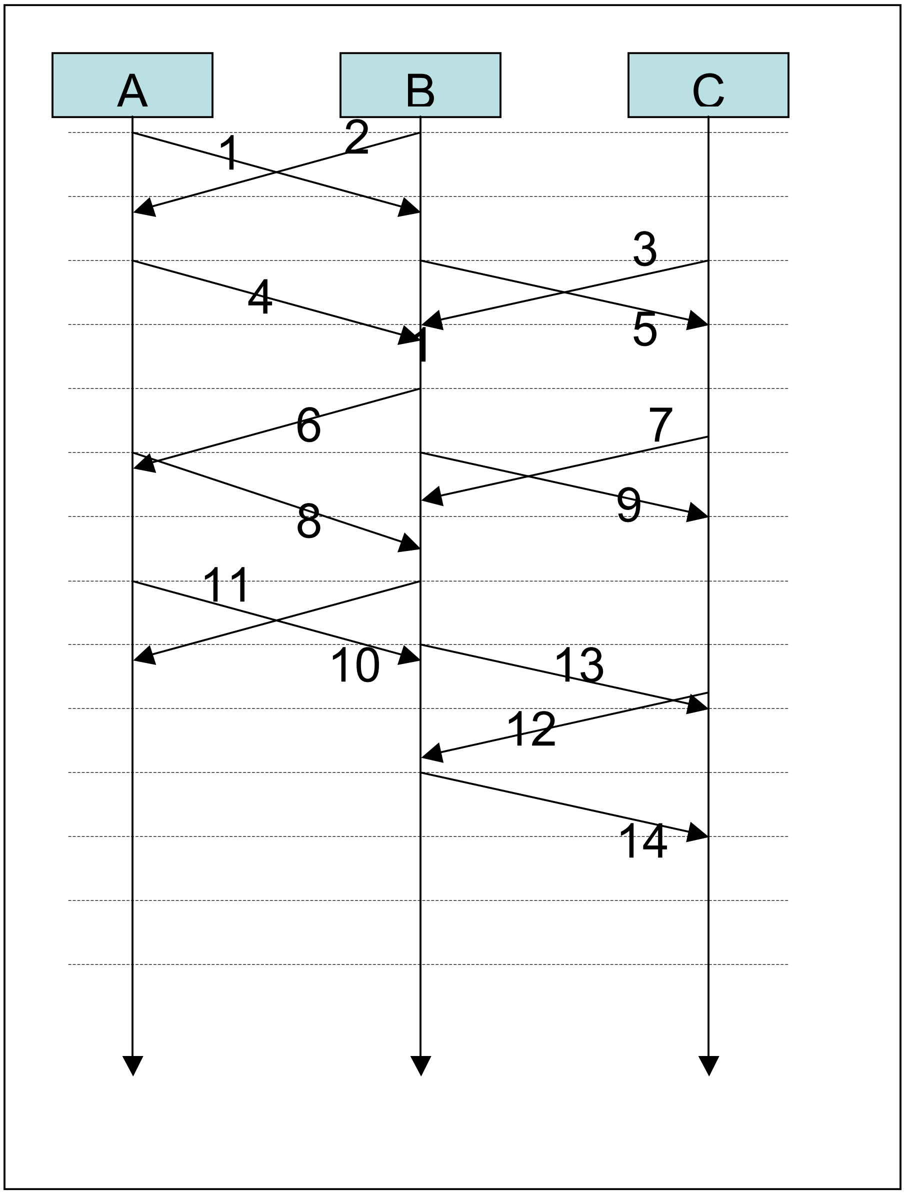


Figure 2.4 Space-Time Diagram

In this network, a branch is directed from node x_i towards node x_j only if flight x_j can be carried out after flight x_i. Flight x_j can be made after flight x_i if flight x_j starts in the city where flight x_i finishes and if the planned departure time of flight x_j is after the finishing time of flight x_i.

Since chains represent vehicle routes, the minimum number of vehicles needed to service a given schedule on the transportation network equals the minimum number of chains into which the acyclic oriented graph can be decomposed, with each node representing flights to be made.

Therefore, by discovering the answer to our question on the minimum number of chains into which an acyclic oriented graph can be decomposed, we also answer the question on the least number of vehicles needed to service a given schedule.

Let $|C|$ denote the minimum number of vehicles required.
Let $|D|$ be the number of branches in the network.
Let $|N|$ be the number of nodes in the graph.
We have $|N| = |D| + |C|$

To solve the problem, we start from node s_1 and construct all branches starting from this node. Figure 2.5 shows that the flight 1 travels from city A to city B. The only flights that it can now fly are flight 5, 6, 9, 10, 13, and 14, i.e., flights that fly from city B after flight 1 reaches city B. We allocate a value of 1 to the first branch (s_1, t_5) as it is the earliest flight after flight 1 reaches city B.

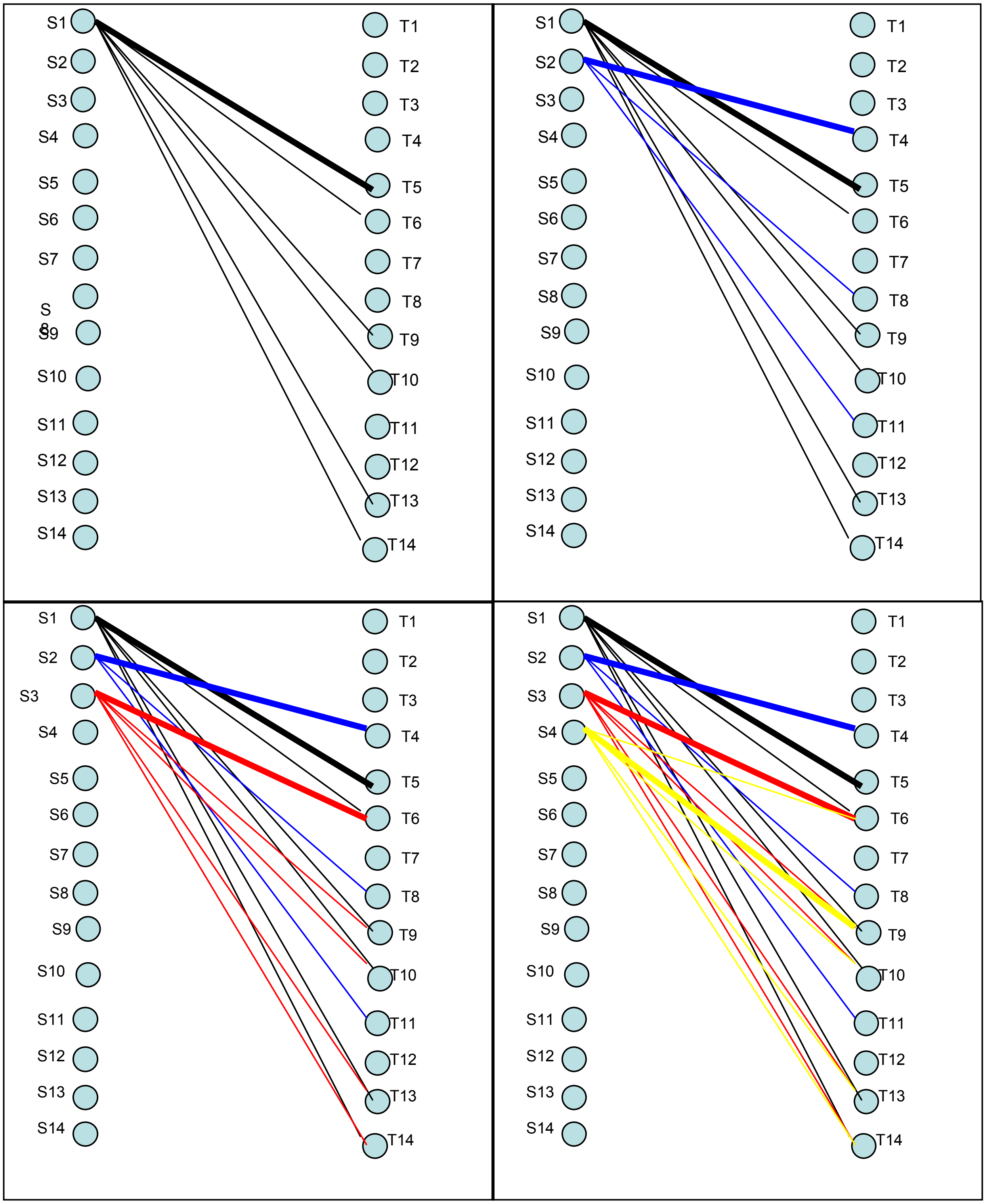


Figure 2.5 Bipartite Graph for branches from S1, S2, S3, S4

The value of 1 to t_5 signifies that flight 5 has been allocated. Since a flow with the highest value of 1 can appear from node s_1, this means that branches (s_1, t_6), (s_1, t_9), (s_1, t_{10}), (s_1, t_{13}), (s_1, t_{14}) are left with flow value of 0. We also know that in the future all branches arriving at node t_5 will be without a flow since node t_5 has already received flow of 1.

Now, we go to node s_2. We construct branches (s_2, t_4), (s_2, t_8) and (s_2, t_{11}) from this node. The first branch (s_2, t_4) is allocated a flow of 1. Similarly, the branches from s_3 are constructed as below and branch (s_3, t_6) is allocated with value 1.

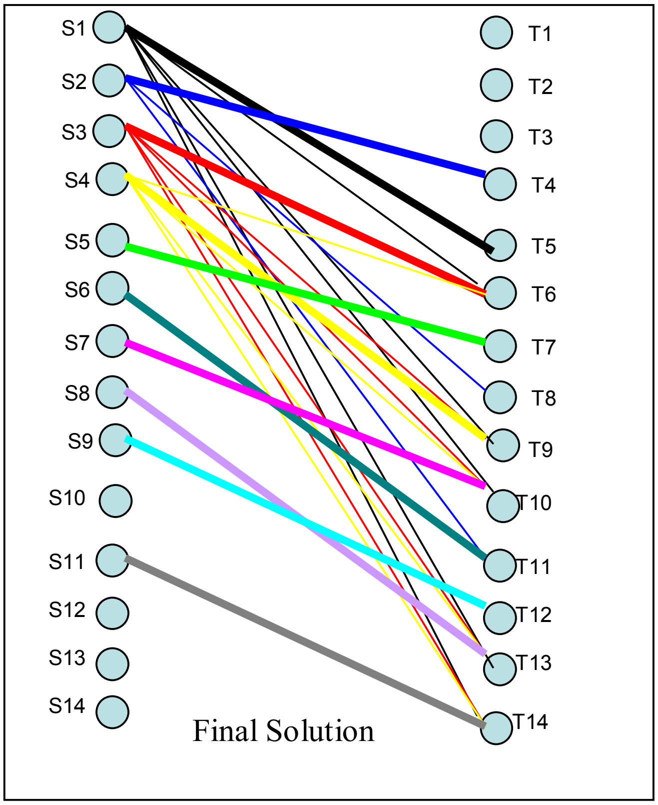


Figure 2.6 Final Solution

We now come to node s_4. Branches starting here include (s_4, t_6), (s_4, t_9), (s_4, t_{10}), (s_4, t_{13}) and (s_4, t_{14}). Here we see that the first branch (s_4, t_6) cannot be allocated as branch (s_3, t_6) arrives at node t_6 with flow of 1. Thus, we allocate the flow to branch (s_4, t_9).

Similarly, with the same procedure, we allocate flows of value 1 to branches (s_5, t_7), (s_6, t_{11}), (s_7, t_{10}), (s_8, t_{13}), (s_9, t_{12}) and (s_{11}, t_{14}). Figure 2.8 gives the final solution.

The bipartite has 10 branches with a flow value of 1. This means that the maximum number of total flows through the bipartite graph is

$|D| = 10$

$|N| = 14.$

Thus, the minimum number of vehicles needed to service the given schedule is

$|C| = |D| - |N|$

$|C| = 4.$

The minimum flights found by this method are not always feasible. The feasibility test can be made by heuristically running the flights in the network. The figure 2.7 shows the routes of all 4 vehicles.

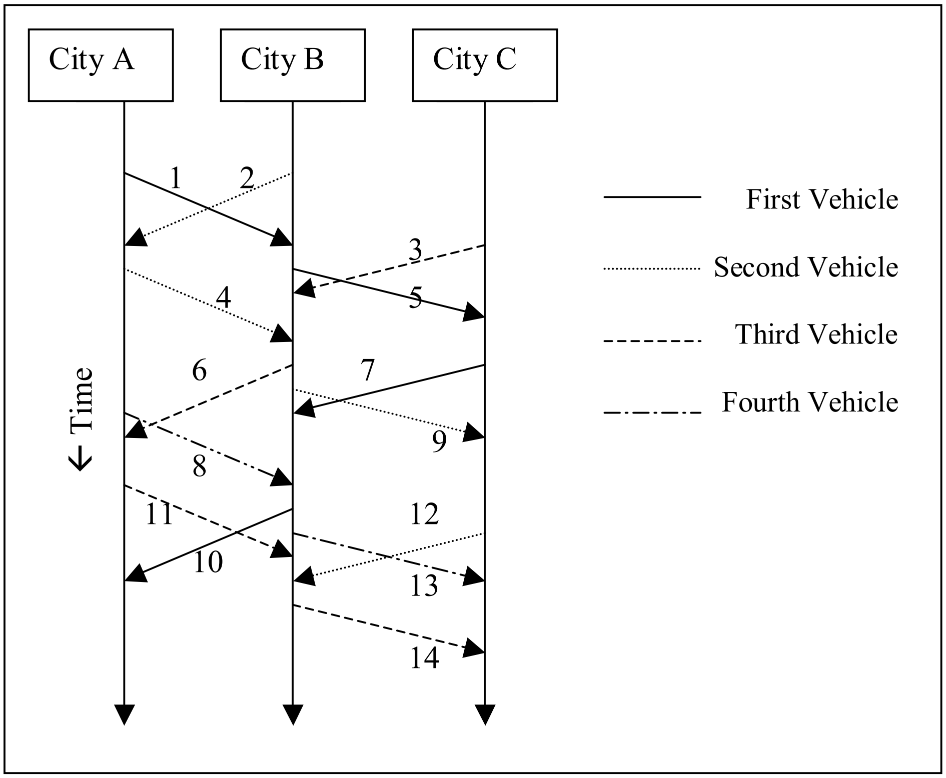


Figure 2.7 Routes for all 4 vehicles

2.6 Aircraft Rotation Models

Aircraft rotation planning is solved with the assignment of flights individual aircraft. The work to be discussed covers two complementary problems:

a. how a given schedule can be realized with minimum number of aircraft, and
b. how the schedule can be changed to reduce the number of aircraft required.

Minimization of the number of aircraft required, of course, conflicts with some of the objectives. The problem of aircraft rotation then would be to assign aircraft to flights and to identify changes in the schedule such that the schedule can be carried out with aircraft on hand. However, defining the problem in this way tends to obscure under utilization of the fleet and the potential for introducing more flights. In contrast, by showing idle aircraft, minimization of the number of aircraft required might facilitate substantial savings. Minimization also provides a much clearer objective function suitable for mathematical programming. If the objective were solely to find feasible solutions, secondary objectives would have to be introduced to select from among the possibly large number of feasible solutions.

Aircraft rotation models, to be realistic, have to take into consideration the maintenance requirements of the aircraft. In many airlines maintenance requirements have been defined in anticipation of a flight schedule so that most rotation plans will meet them automatically. It may be advantageous to exclude the maintenance constraints when constructing the rotations. The construction of aircraft rotations, which do not include maintenance constraints, has received considerable attention. Comparatively little work on the other hand has been devoted to models that include maintenance constraints.

Clarke, Johnson, Nemhauser and Feo (1997) study the maintenance location problem, which involves finding the minimum number of maintenance stations required to meet the specified 4-day requirement for a proposed flight schedule. They assume that the interim stops of flights during the day are unimportant. Using the one-day routings between overnight cities as input, they formulate the maintenance base location problem as a minimum cost, multicommodity network flow problem with integer restrictions on the variables. Kabbani and Patty(1992) study the maintenance routing problem for American Airlines where each aircraft needs to be checked every three days. They formulate the problem as a set partitioning model where a column represents a possible weeklong routing and a row of a flight. Gopalan and Talluri(1998) study the maintenance routing problem for US Air, where each aircraft needs to have a routine check every three days and a “balance-check” once in a while. They consider the problem for a daily schedule under the term “static infinite horizon model.”

2.7 Aircraft Scheduling

B. P. Loughran (1970) gives an airline schedule construction model which is called “Timetable Building Module.” The Timetable Building Module is comprised of two programs: the initial scheduling program and the fleet reduction program. Figure 2.10 shows the flowchart for the initial scheduling program that takes the output from the route retrieval routine (number of daily flights on each route) and, using this time-of-day demand distribution profile for each city pair, produces an initial timetable tailored to passenger demand desires but without regard to efficient usage of the fleet from the airline’s point of view. The fleet reduction program, operating on the initial timetable, then produces more efficient flight itineraries by shifting the arrival and departure times slightly to affect flight connections and thus reductions in aircraft required.

Scheduling for chartered aircraft must consider dynamic demand. The entire scheduling process is demand responsive.

In a survey of the general aircraft scheduling problem by Etschmaier and Mathaisel (1984), the concept of dynamic scheduling similar to Demand Driven Dispatch is discussed. Berge and Hopperstad(1993) give a demand driven dispatch model for dynamic aircraft capacity assignment. Kikuchi and Jong-Ho(1989) solve the demand responsive transportation system and the method used can be characterized by the following elements.

1. Vehicle schedules are built one vehicle at a time.
2. Insertion possibility is examined for a group of trips simultaneously.
3. A trip may be inserted into any place on the time axis as long as it satisfies all the time constraints.

Horn (2002) describes a software system to manage the deployment of a fleet of demand-responsive passenger vehicles. Apart from its scheduling functions, the system includes automated vehicle dispatching procedures designed to achieve a favorable combination of customer service and vehicle deployment efficiency.

Charter aircraft scheduling is discussed by Ronen (2000). The paper deals with scheduling several fleets of small air jet aircraft by a charter operator, and describes a computerized decision support aid. The problem handled is a fleet schedule of several types of aircraft. They use an Elastic Set Partitioning (ESP) model, which is very appealing to problems where costs are non-linear and discrete, and complex operational rules are involved. Keskinocak and Tayur (1998) consider scheduling time-shared jet aircraft. They show that the jet aircraft scheduling problem is NP complete and formulate the problem as a 0-1 integer program and solve small and medium size problems by Cplex.

One of the earliest computerized schedule construction studies was conducted by R. W. Simpson (1966). Simpson determines the frequency pattern for non-stop services and then constructs a timetable to assign departure times for each service on every route. The computer program constructs an initial timetable given

1. the frequency pattern,
2. block time, and
3. data describing the daily variation in demand for each city pair.

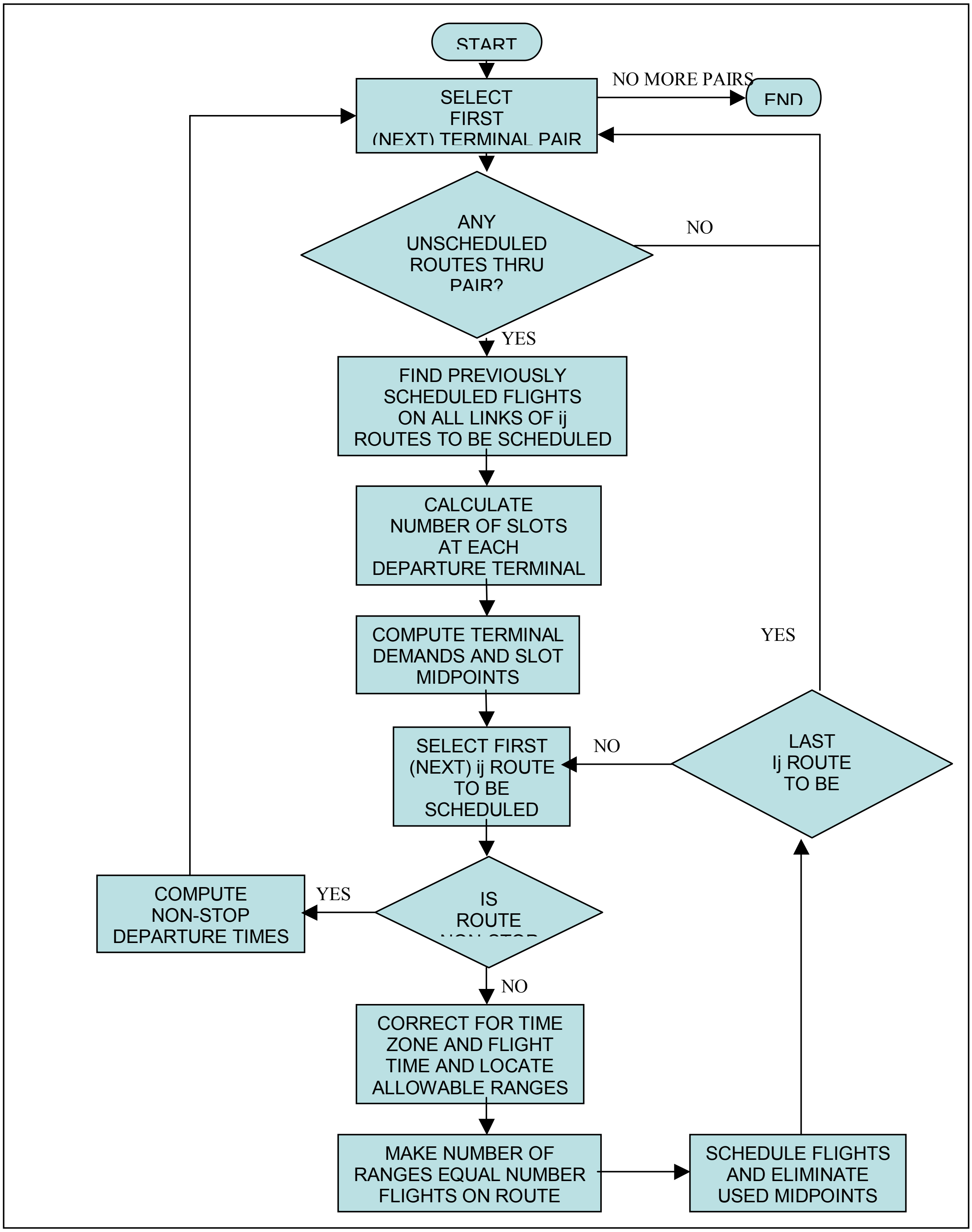


Figure 2.8 Initial Scheduling Program

Chapter 3

Methodology

3.1 Problem

In the previous chapter, we have seen various scheduling techniques for conventional airlines. However, the scheduling problem addressed in this thesis is for an air taxi system. Scheduling air taxi differs from conventional aircraft scheduling in the following ways.

1. Conventional airlines have a fixed schedule, and passengers plan their travel according to this schedule. However, the air taxi system schedules flights only if there is demand. Thus, the air taxi system will plan its schedule according to passenger demand.
2. The schedule for conventional airlines is fixed for some specific period of time, say a week or month. The air taxi system will have a schedule that will change daily depending on the demand for that particular day.
3. Conventional airlines have scheduled flights for each aircraft. However, the aircraft in an air taxi system may run a different flight schedule each day.

The air taxi service will be used primarily by business passengers who wish to fly from one city to another on any day and at any time. This type of demand is very difficult to predict. However, on the basis of previous studies, it is possible to approximate annual or monthly demand for a specific O-D pair. This demand can be further refined to estimate daily demand. The daily demand can be further refined to hourly demand. Passenger arrival in an air taxi system is nothing more than the time at which he requires the flight. Farrell (2001) gives probability of passenger arrival for each hour of the day. Table 3.1 shows the time of the day and also the probability of arrivals.

Time of the Day	Probability of Passenger Arrival
00:00 - 01:00	0.002579
01:00 - 02:00	0.001433
02:00 - 03:00	0.000860
03:00 - 04:00	0.001146
04:00 - 05:00	0.000573
05:00 - 06:00	0.003483
06:00 - 07:00	0.065043
07:00 - 08:00	0.078223
08:00 - 09:00	0.053582
09:00 - 10:00	0.071060
10:00 - 11:00	0.061891
11:00 - 12:00	0.045559
12:00 - 13:00	0.055587
13:00 - 14:00	0.050430
14:00 - 15:00	0.057020
15:00 - 16:00	0.057593
16:00 - 17:00	0.053009
17:00 - 18:00	0.087679
18:00 - 19:00	0.075358
19:00 - 20:00	0.064183
20:00 - 21:00	0.057020
21:00 - 22:00	0.032951
22:00 - 23:00	0.014900
23:00 - 24:00	0.008883

Table 3.1 Hour of the day and Probability of Passenger Arrival

Table 3.1 presents the probability mass function of daily passenger arrival.

3.2 SATS Aircraft

For our study, we will consider the Eclipse 500 as the sole aircraft in the air taxi service. The eclipse has been already documented by NASA in its vision for SATS. The performance specifications of the aircraft are presented in Table 3.2.

Eclipse 500 Jet Performance		
	Imperial	Metric
Takeoff Distance	2,155 ft	657 m
Sea level, ISA to 50 ft @ MGTOW		
Landing Distance	2,040 ft	622 m
Sea level, ISA @ 4,600-lb landing weight		
Rate of Climb - 2 engines	2,990 ft/min	911 m/min
Rate of Climb - 1 engine	888 ft/min	271 m/min
Time to Climb - 35,000 ft	19 min	
Takeoff at 5,000 ft at 68°F	3,350 ft	1,021 m
(1,524 m at ISA +15)		
Single Engine Takeoff Climb at 5,000 ft at 68°F (1,524 m at ISA + 15)	293 ft/min	89 m/min
Single Engine Service Ceiling	25,000 ft	7,620 m
Cruise Speed	375 kt	694 km/hr
Max. Altitude	41,000 ft	12,497 m
Range, 4 occupants	1,395 nm	2,584 km
IFR 45-minute reserve		

Table 3.2 Eclipse 500 Performance

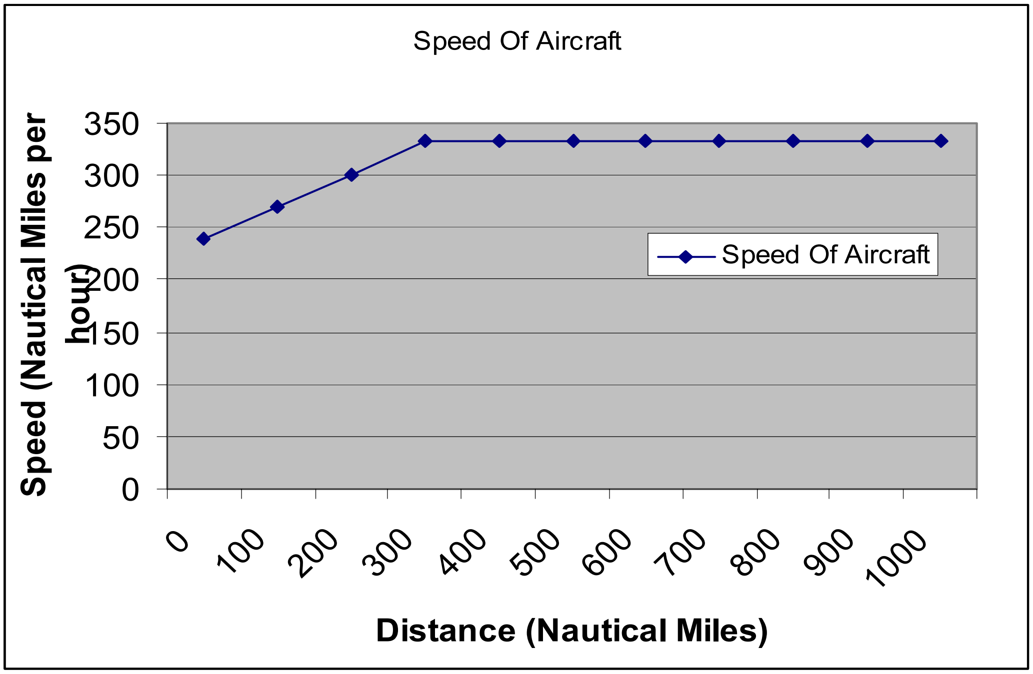


Figure 3.1 Speed of Aircraft

The figure 3.1 illustrates the speed of aircraft for different distances. It is observed that the aircraft cruises after it has traveled 300 nautical miles. The total time required for a flight can be found out by knowing the distance between two cities.

$$\text{Total time} = \frac{\text{Distance between the cities}}{\text{Speed of the aircraft}}$$

In this research, the aircraft service will be provided for cities within a range of 500 nautical miles. This means that flight time will be at most 1 hour and 30 minutes. The time taken by a flight from take off to landing is known as the “block time.” We assume the block time to be 1hour and 30 minutes (maximum).

3.3 Time Unit

The study will consider a select group of cities that are each separated by a distance of less than 500 Nautical Miles. Thus, the maximum time that the flight flies will be 1 hour 30 minutes to about 2 hours. Thus, we consider a time unit of 2 hours for convenience. This restriction does not affect the general procedure presented here.

The two hour time unit will be constant for all origin destination pairs. The entire service time is split into 2 hour time units beginning at 4 a.m., with the last flight departing at 10 p.m. Thus, for every origin destination pair, the time units and probability of passenger arrival are provided in Table 3.3.

Time of the Day	**Time Unit**	**Probability of Passenger Arrival**
04:00 - 06:00	1	0.007367
06:00 - 08:00	2	0.146577
08:00 - 10:00	3	0.127953
10:00 - 12:00	4	0.110761
12:00 - 14:00	5	0.109328
14:00 - 16:00	6	0.117924
16:00 - 18:00	7	0.143999
18:00 - 20:00	8	0.142852
20:00 - 22:00	9	0.093282

Table 3.3 Probability of Passenger Arrival by Time Unit

From Table 3.3, we can determine the probability density function for passenger arrivals.

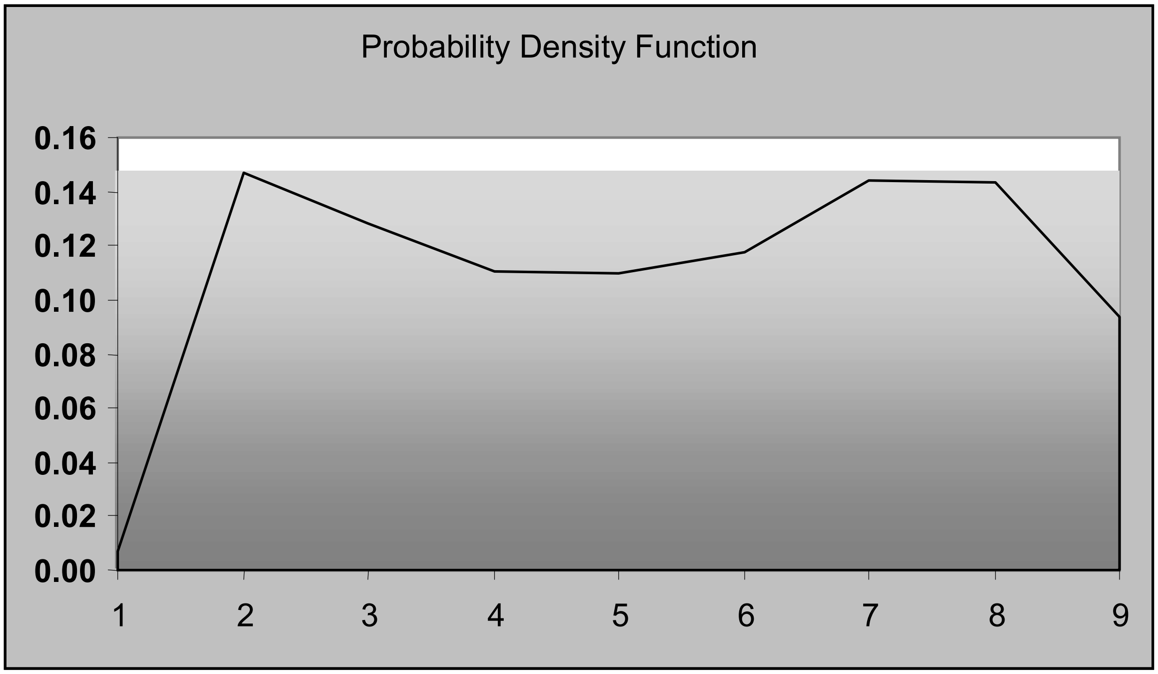


Figure 3.2 Probability Density Function of Passenger Arrival

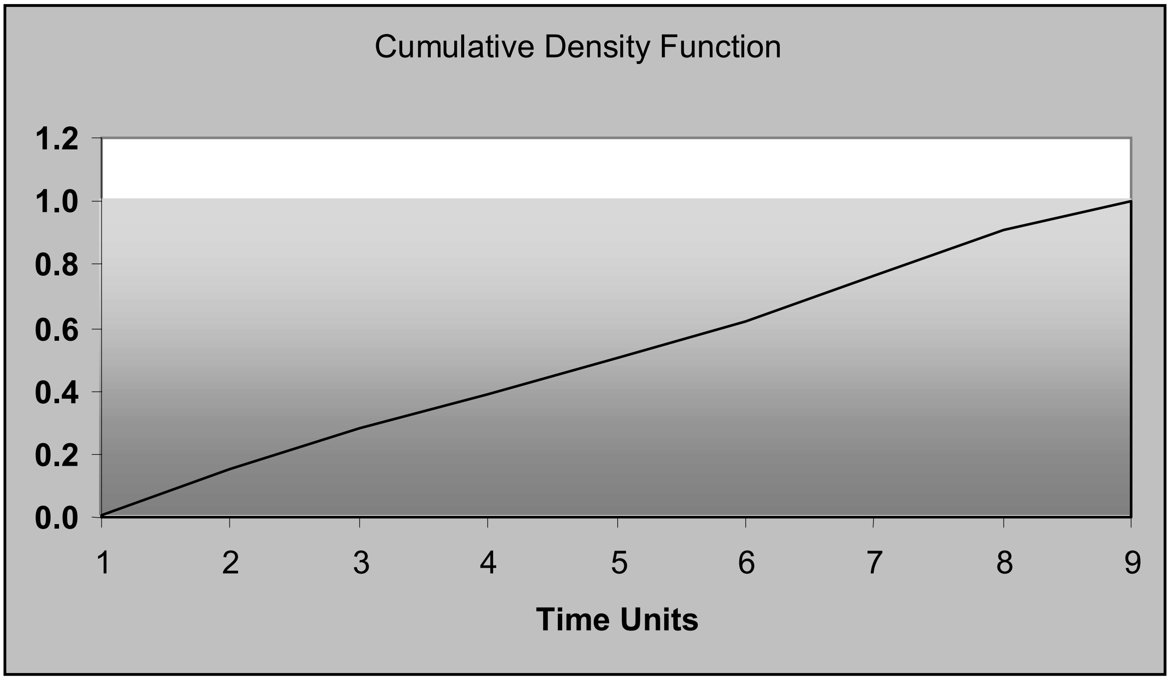


Figure 3.3 Cumulative Density Function of Passenger Arrival

Figure 3.4 shows the cumulative distribution function, which is independent of daily demand.

3.4 Demand and Flight Frequency

The demand for any O-D pair can be determined from the population and the travel trend between the two cities.

The daily demand for a city in the network is a dynamic generation of a number between 0 and 40. This daily demand is split into demand per time unit depending on Table 3.3. The minimum number of flights required for one day can be found from the following equations Teodorovic,1988.

Let
A = number of aircraft
N_{ijt} = number of flights assigned from origin i to destination j at time unit t
n = number of seats in each flight
λ_{ijt} = demand between origin i to destination j at time unit t
d_{ij} = distance between origin i to destination j
C_{ij} = cost to fly a flight from origin i to destination j
S = time required to take off and land (constant for all flights)
V = speed of the aircraft
U = maximum possible utilization of the aircraft

The objective function is to minimize the cost of flying the plane each time.

$$\textbf{Min} \sum_{t=1}^{9} \sum_{(i,j)} N_{ijt} C_{ij}$$

The constraints are described below.

a) Demand Fulfillment Constraint

Supply of seats offered > Demand

$nN_{ijt} \geq \min(\lambda_{ijt},6)$ for all (i, j) city pairs and all time units t.

The number of seats in each time unit for each O-D pair is a product of the number of seats per flight and number of flights per unit time

b) Number of flights in a time unit

$\sum_{(i,j)} N_{ijt} \leq A$ for t = 1, 2, …, 9.

This will ensure that the total number of flights per time unit for the network is less than or equal to the fleet size.

c) Non negativity Constraint

$N_{ijt} \geq 0$

The above equations will give the time units at which we should schedule a flight for any OD pair. It will also give the number of flights required for the network.

The network, consisting of a number of cities, will provide direct (non-stop) service from each city to every other city. As the number of cities increases, the number of O-D pairs will increase according to the following rule:

$$l(l-1)$$

where, l = number of cities.

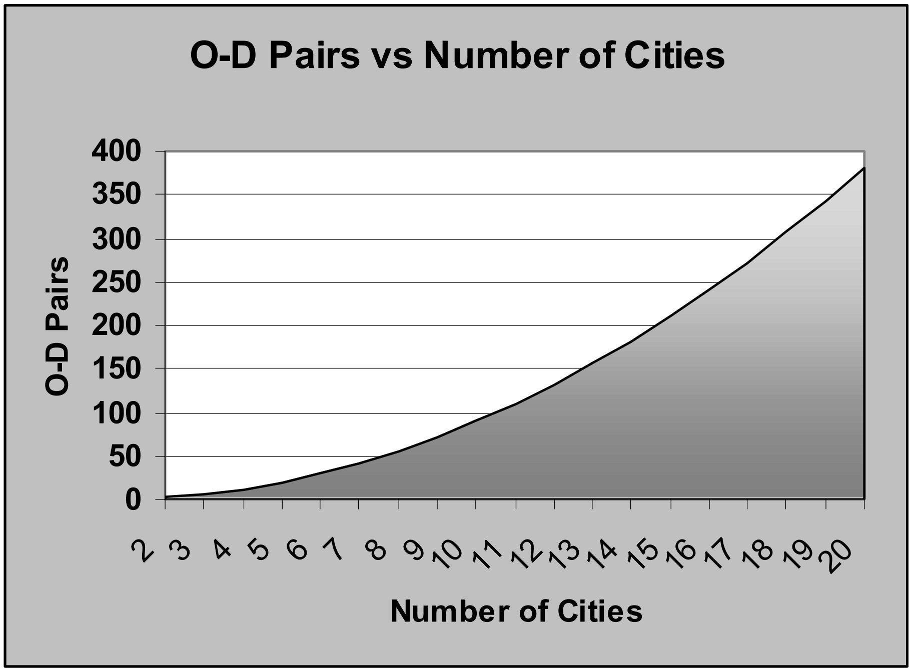


Figure 3.4 Relation of O-D pairs with the number of cities

Each O-D pair will have its own demand for the day. The demand for the day will determine the rate of passenger arrival for every two hour block. The next task is to determine the time units for the flights

3.5 Passenger Waiting Time

Teodorovic (1988) gives the average schedule delay per passenger, D, as

$$D = \frac{T}{4N}$$

where, T = Total time period in minutes, and

N = Flight Frequency.

The flight day is defined as 4 a.m. to 10 p.m. i.e. 18 hours = 1080 min.

If there is one flight flying per time unit, the average schedule delay will be

$$D = \frac{1080}{36} = 30 \text{ min.}$$

For the system defined here, a similar check can be performed except that each time unit must be handled differently. The waiting time for each passenger in every time unit must be found. To fly a flight in a specific time unit, it should minimize the waiting time of all the passengers that arrive in that time unit.

Consider a single time unit (two hours). Let the number of passengers requesting flights in this time unit be six (maximum seating capacity). Let H(t) be the cumulative distribution function as defined previously in figure 3.4. If the flight flies at the end of the time unit, then the waiting time is from start of the time unit to the end.

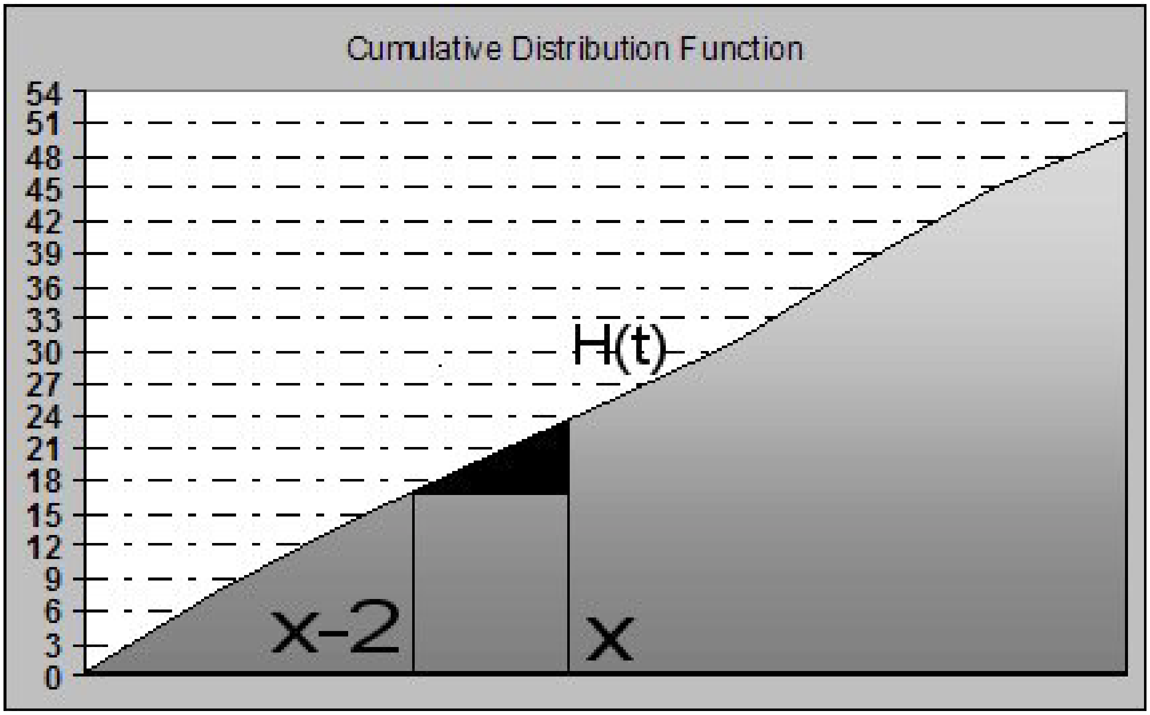


Figure 3.5 Waiting time for passenger

For example, if a flight flies at time x, then maximum waiting time is the shaded portion in Figure 3.6.

$$W = \int_{x-2}^{x} H(t)dt - 2 * H(x-2)$$

where, H(x-2) = total passengers arrived by time (x-2).

The average waiting time will be the total waiting time divided by the number of passengers served.

3.6 Flight Assignment and Routing

Flight assignment is the most crucial part of airline schedule. The algorithm that will find the entire schedule is given below.

3.6.1 Algorithm 1 – Generate a matrix that will provide data regarding the proximity of cities from a particular city individually.

Step 1:

The algorithm requires the distance between each city. This information is stored in a square matrix with row and columns equal to number of cities. The matrix will have the distance from a city to every other city. Matrix F shows the cities compared, and matrix G shows the values, for a network of four cities.

$$F = \begin{bmatrix} 1-1 & 1-2 & 1-3 & 1-4 \\ 2-1 & 2-2 & 2-3 & 2-4 \\ 3-1 & 3-2 & 3-3 & 3-4 \\ 4-1 & 4-2 & 4-3 & 4-4 \end{bmatrix}$$

$$G = \begin{bmatrix} 0 & 23 & 65 & 76 \\ 23 & 0 & 78 & 22 \\ 65 & 78 & 0 & 12 \\ 76 & 22 & 12 & 0 \end{bmatrix}$$

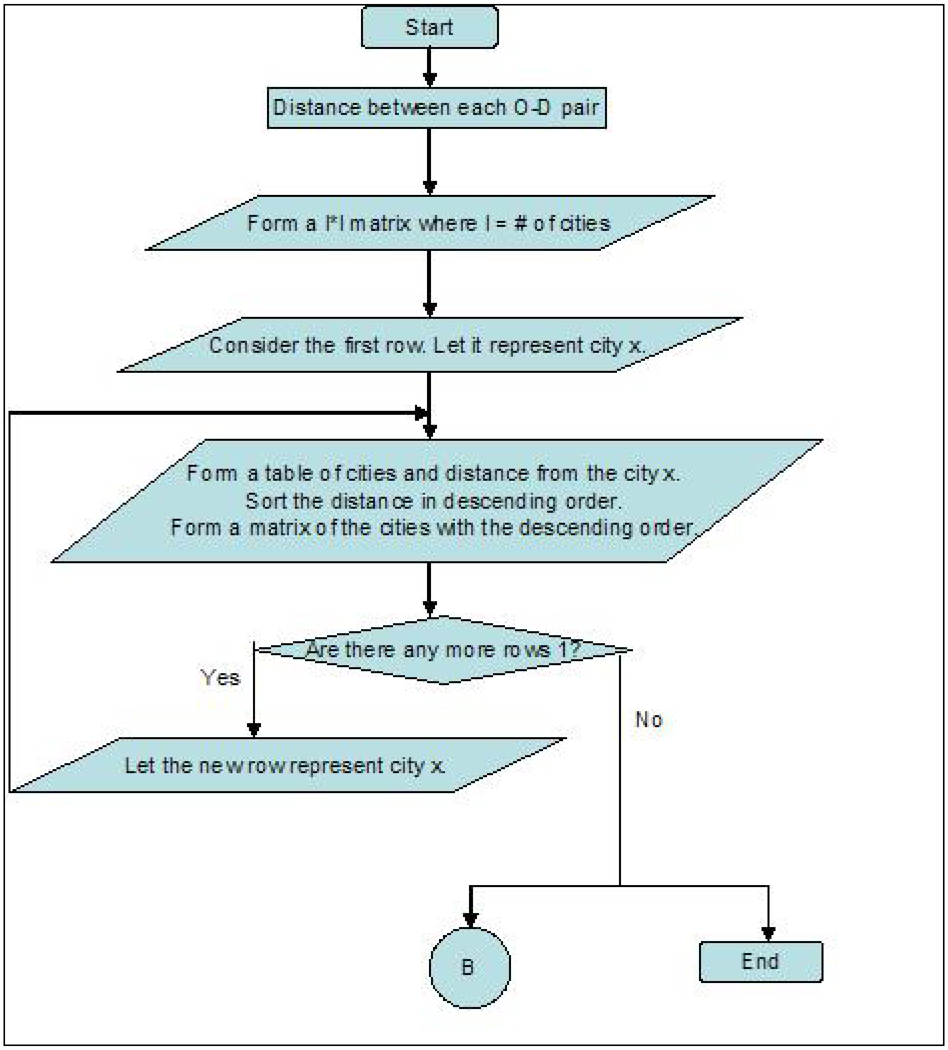


Figure 3.6 Flowchart of algorithm 1

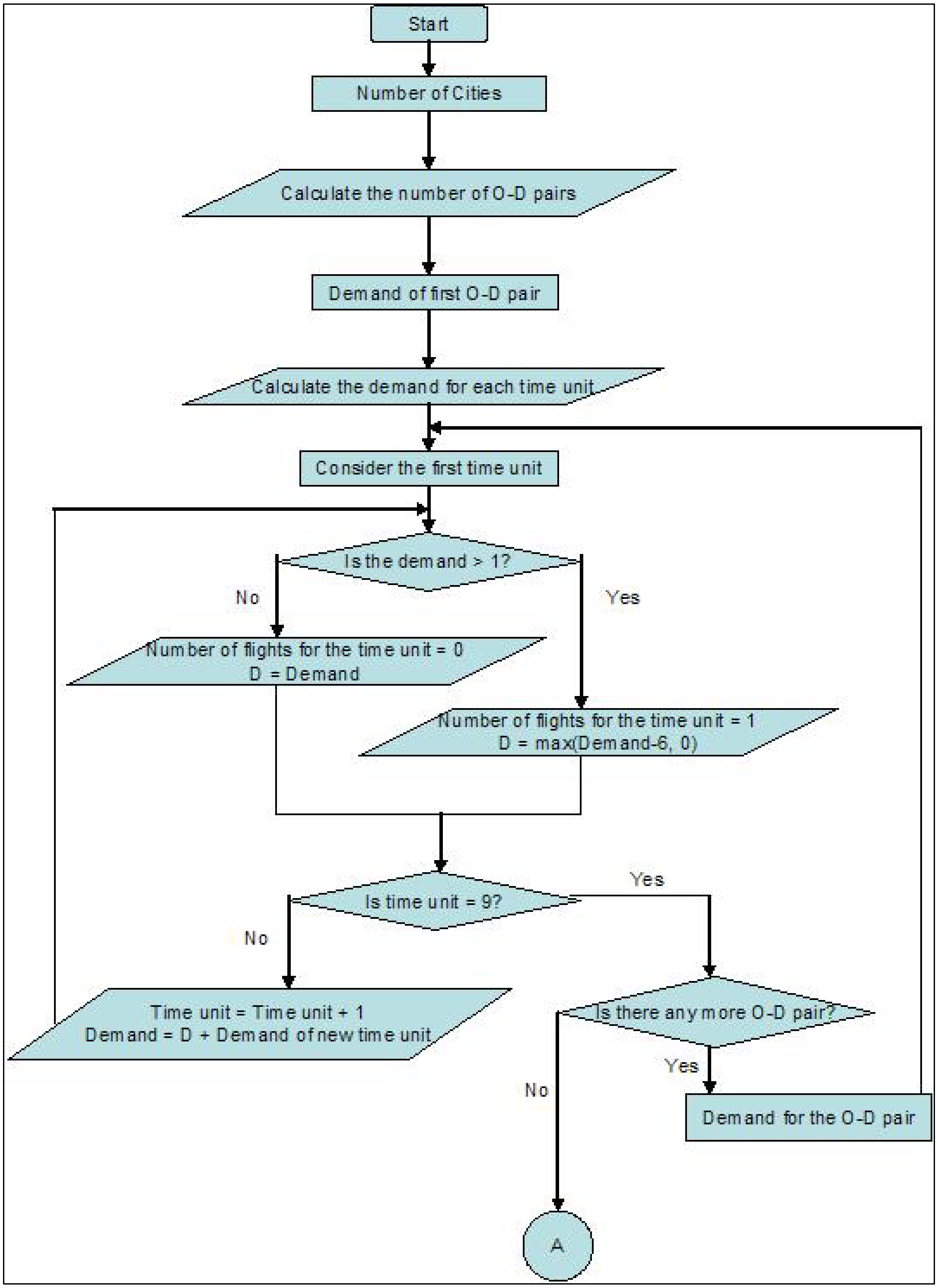


Figure 3.7 Flowchart of algorithm 2 Part(a)

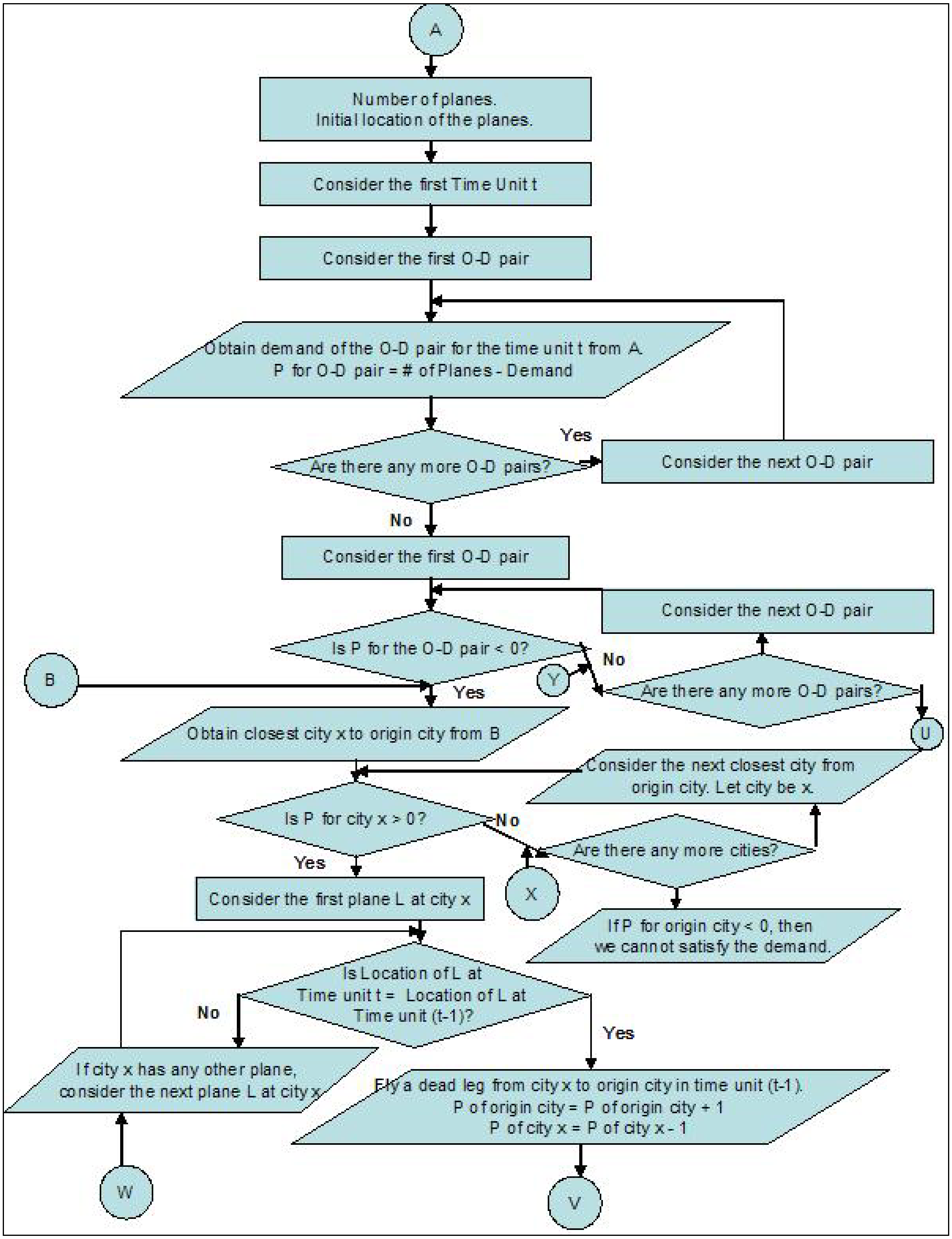


Figure 3.8 Flowchart of algorithm 2 Part(b)

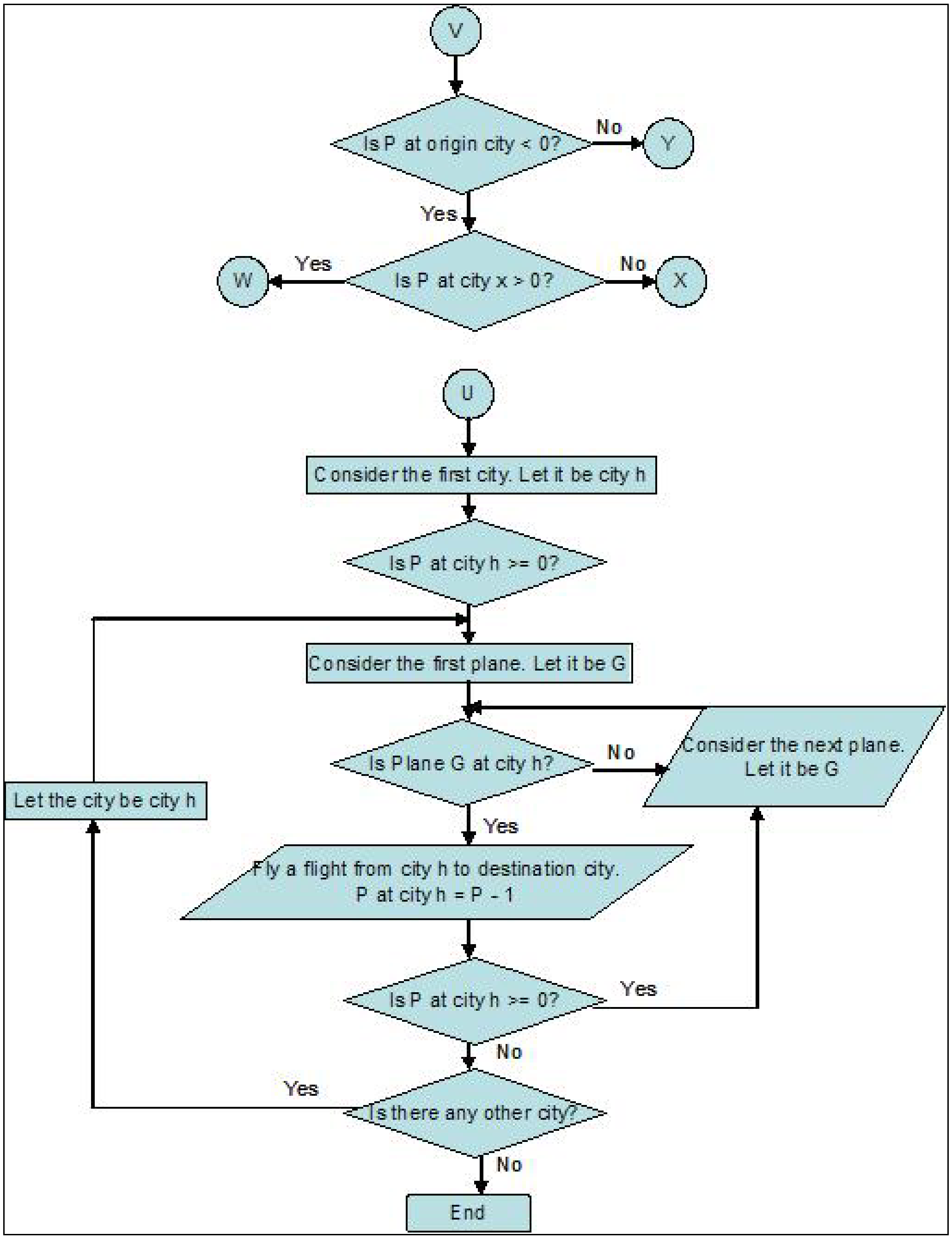


Figure 3.9 Flowchart of algorithm 2 Part(c)

Step 2:

Each row of matrix G will be then arranged in descending order. This will arrange the cities from farthest to the closest for the city represented by that row. Matrix I shows that, for city 1, the closest city is city 2, then 3 and finally 4. Similarly, for city 2, the closest city is city 4, then 1 and then 3.

$$G = \begin{bmatrix} 76 & 65 & 23 & 0 \\ 78 & 23 & 22 & 0 \\ 78 & 65 & 12 & 0 \\ 76 & 22 & 12 & 0 \end{bmatrix}$$

$$I = \begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 1 & 4 & 2 \\ 2 & 1 & 4 & 3 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

3.6.2 Algorithm 2 – Develop a schedule for the network

The algorithm will first find the flights for the network for any number of cities. The number of O-D pairs is found out by the equation given below.

$l(l-1)$, where l = number of cities in the network.

Step 3:

The algorithm will be fed with daily demand for each O-D pair. The demand for each O-D pair will be split into demand for each time unit according to the probability factors from earlier referenced p.d.f (Table 3.4).

Time Unit	Probability of Arrival
1	0.006
2	0.144
3	0.127
4	0.11
5	0.108
6	0.117
7	0.143
8	0.142
9	0.093

Table 3.4 Probability of passenger arrival per unit time

Step 4:

If the demand for any O-D pair for any time unit is greater than one, assign a flight for the O-D pair in that time unit.

If the demand for any time unit is greater than six, assign a flight for the O-D pair in that time unit. This flight will carry only six passengers, while the remaining passengers are carried to the next time unit.

Thus, if there is any demand greater than one for any time unit, there will be a flight for that O-D pair in that time unit, otherwise there will be no flights for that O-D pair in that time unit. This will give the time units in which there are flights for that O-D pair. Each O-D pair will have a matrix with nine rows (corresponding to nine time units) and one column with values of either 0 or 1. An example is shown below.

$$Q = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Step 5:

Step 4 is repeated for all the O-D pairs to obtain the flight information for all O-D pairs in each time unit. The final matrix will have nine rows (corresponding to nine time units) and columns equal to the number of O-D pairs. Matrix J shows the flight information for a network of six cities. A network of six cities will have 30 O-D pairs.

$$J = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

The number of cities, their daily demand and the distance is found. The next step is to determine the number of planes for the network. The information required is the number of planes and their location at the start of the day.

The information required to schedule the flights is the flight information for each O-D pair in every time unit, the distance from each city to every other city in the network, and the initial location of the planes.

Step 6:

The schedule is solved for each time unit. Each time unit schedule is further solved for each O-D pair. Consider the first time unit. The demand for all O-D pairs, and the number of planes located at each origin are known. The equation below gives the status of each city.

$$P_i = \sum_x A_{ix} - \sum_j D_{ij}$$

where, x = the aircraft number

i = origin city

j = destination city

A_{ix} assumes the value of 1 or 0 depending on whether the aircraft is available at the origin city or not.

D_{ij} assumes the value 1 or 0 depending or whether there is a demand from city i to city j.

P_i provides with the information about the state of city i.

P_i can be found for all the cities. The value of P_i determines the state of the system at city i for a specific time unit. If

$P_i = 0$, then city i has the exact number of aircraft required to satisfy the demand,

$P_i > 0$, then city i has surplus planes, and

$P_i < 0$, then city i has more demand than the number of aircraft.

Step 7:

If the value of P_i for any O-D pair is greater than or equal to zero, then the origin city is self sufficient in satisfying the demand for that time unit. If P_i is less than zero, the origin city will require planes from other cities. This will require dead legs to be flown from other cities to the origin city in the previous time unit. The method below is used to determine the city from which the dead leg is flown.

Method to determine the dead leg:

- This method is used if the value of P_i is less than zero for any city. The previously determined matrix I from algorithm-1 provides the information of cities closest to any city. This matrix will determine the city from which the dead leg needs to be flown.
- From matrix I, the row that represents the origin city is found. This will provide the list of cities in the order of their distance from the origin city.
- The city closest to the origin city is selected and the value of P is found for that city. If the value of P is greater than zero, we need to check the activities of all the planes located at that city. If a plane has not flown in a time unit, it is assumed to have performed no activity for that time unit.
- For assigning a dead leg, it is necessary to know the activity of the plane in the previous time unit. If the plane has no activity in the previous time unit, then that plane can be assigned a dead leg.
- If the value of P is less than or equal to zero or if all the planes have an activity in the previous time unit, then those planes cannot be assigned a dead leg.

Step 8:

If the value of P for the origin city is still less than zero, then the next closest city is considered and Step 11 is repeated.

Step 9:

A matrix, Z, will give the position and the activity of each plane for all 9 time units. The first row represents the dummy row that will handle any initial adjustments in the positions of the planes, while the remaining rows of this matrix represent the time units and the columns represent the plane. As the initial position of the planes is known, the first and the second row are populated. All the other rows are populated with a high number e.g. 99. If there are 3 cities and 5 planes and the planes are initially located at 1, 2, 2, 3, 1 respectively, then the matrix Z will be

$$Z = \begin{bmatrix} 10 & 20 & 20 & 30 & 10 \\ 10 & 20 & 20 & 30 & 10 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

The location of the planes is represented by the extreme left hand side digit of the number. If this digit is followed by a zero, it represents that no activity has been assigned to this plane in that time unit. If a dead leg is flown by plane 5 from city 1 to 2 in the dummy time unit, then we assign the number 120 to column 5 and row 1. Any number in the matrix that is greater than 99 represents a dead leg flown and the flight can be read as explained below.

120 = a dead leg from city 1 to city 2.

310 = a dead leg from city 3 to city 1.

Step 10:

Steps 7 to 9 are repeated for all O-D pairs. At the end of step 9, we will have flown dead legs wherever necessary. Also, the values of P are changed due to the dead legs flown.

Step 11:

The summation of new P_i determines the status of the entire network for any time unit. If the summation is greater than or equal to zero, the network is stable i.e. the number of planes available will satisfy the demand for that time unit. However, if the summation is less than zero, then the available planes are insufficient to satisfy the demand.

Step 12:

This step will allocate the flights that will satisfy the demand. Every O-D pair needs to repeat this step.

Consider an O-D pair. If the value for this O-D pair in the specific time unit is 1 in the matrix J, then one plane will fly from the origin to destination.

If plane 3 is flown from city 2 to city 1 in time unit 4, then the 3rd column of the 5th row is populated by the number 21. The number tells that the plane is flown from city 2 to city 1.

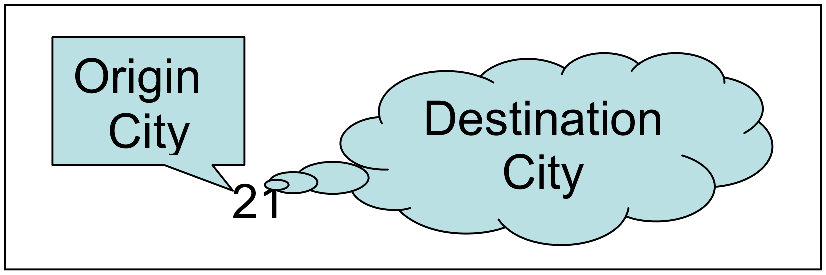


Figure 3.10 Explanation of number populated in the schedule matrix

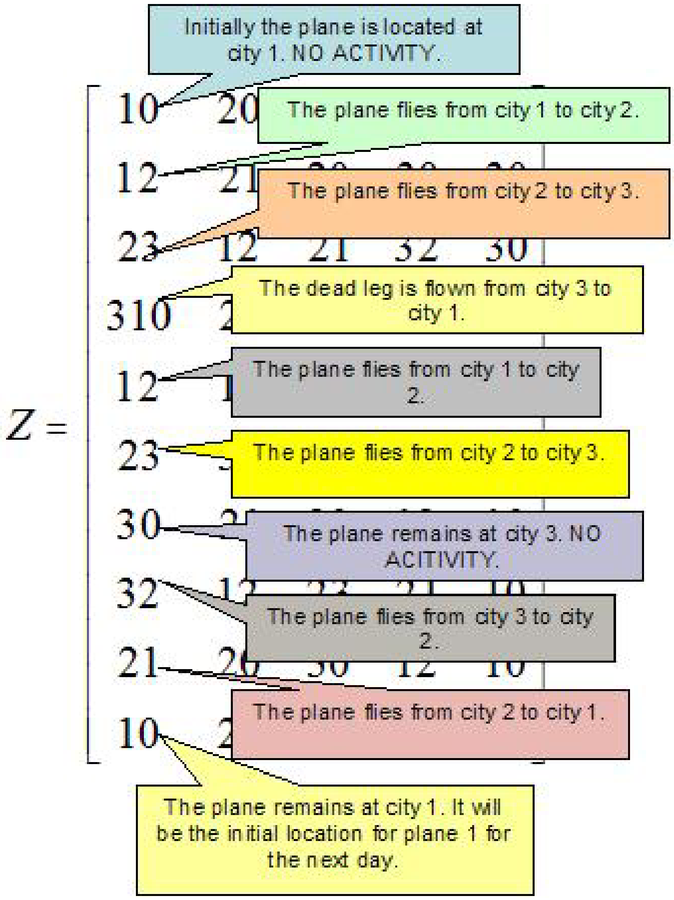
The step is repeated for all the O-D pairs. After the completion of all O-D pairs, we obtain the flights flown by each plane in the specific time unit. The destination city of each plane becomes the origin city for the same plane in the next time unit. If the plane has no activity in the time unit, it will have the same location for the next time unit.

Step 13:

Steps 6 to 12 are repeated for all the time units. At the end of the 9th time unit, matrix Z is populated with all the flight information of the planes throughout the day. Matrix Z, below, gives a final schedule for a network of 3 cities and 5 planes.

$$Z = \begin{bmatrix} 10 & 20 & 20 & 30 & 30 \\ 12 & 21 & 20 & 30 & 30 \\ 23 & 12 & 21 & 32 & 30 \\ 310 & 21 & 12 & 20 & 30 \\ 12 & 13 & 21 & 20 & 31 \\ 23 & 32 & 12 & 21 & 10 \\ 30 & 21 & 20 & 12 & 10 \\ 32 & 12 & 23 & 21 & 10 \\ 21 & 20 & 30 & 12 & 10 \\ 10 & 20 & 30 & 20 & 10 \end{bmatrix}$$

The above is explained as shown in Figure 3.8.



Initially the plane is located at city 1. NO ACTIVITY.

The plane flies from city 1 to city 2.

The plane flies from city 2 to city 3.

The dead leg is flown from city 3 to city 1.

The plane flies from city 1 to city 2.

The plane flies from city 2 to city 3.

The plane remains at city 3. NO ACITIVITY.

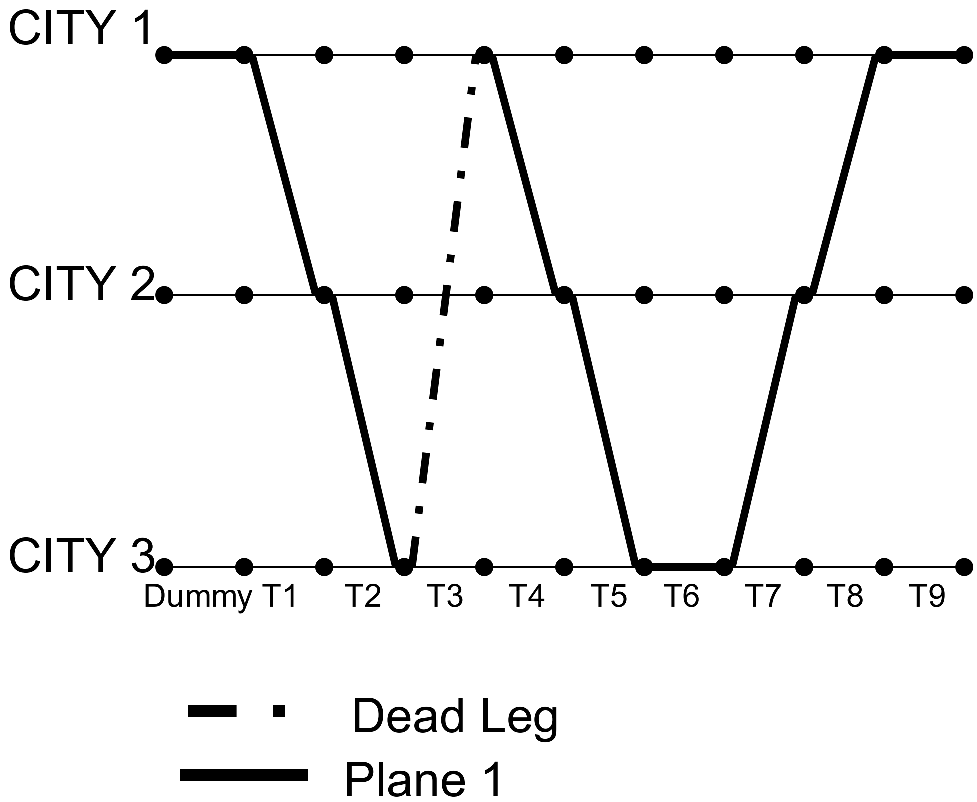
The plane flies from city 3 to city 2.

The plane flies from city 2 to city 1.

The plane remains at city 1. It will be the initial location for plane 1 for the next day.

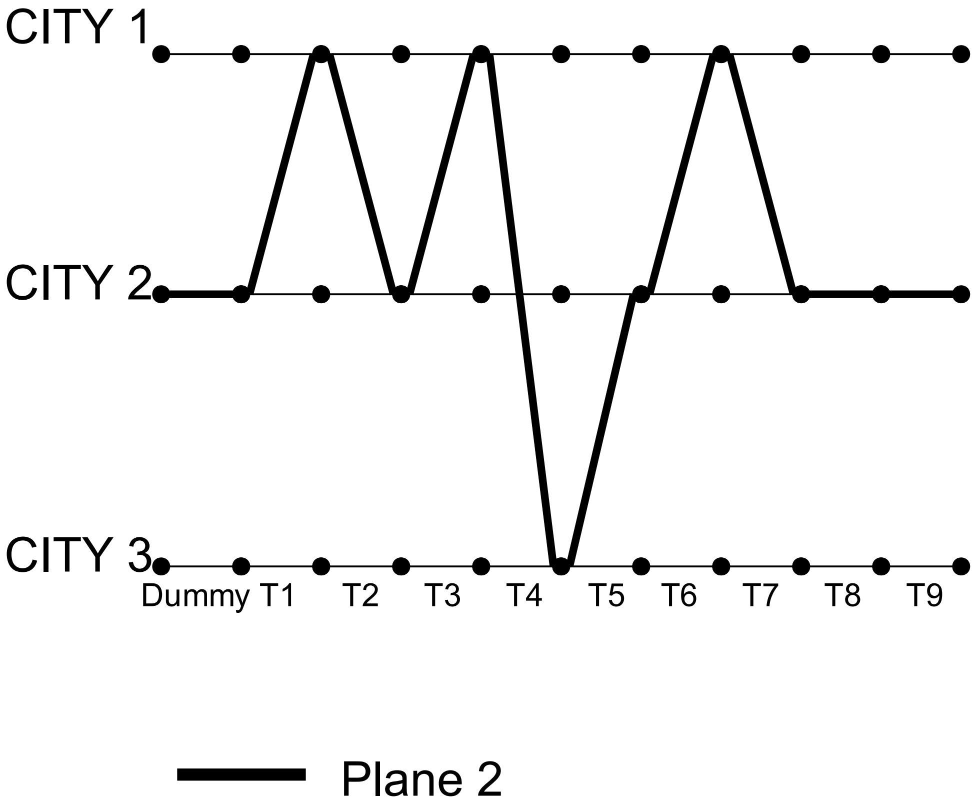
Figure 3.11 Description of a daily schedule for a single aircraft

Figures 3.12, 3.13, 3.14, 3.15, 3.16 show the route of each aircraft in the network.



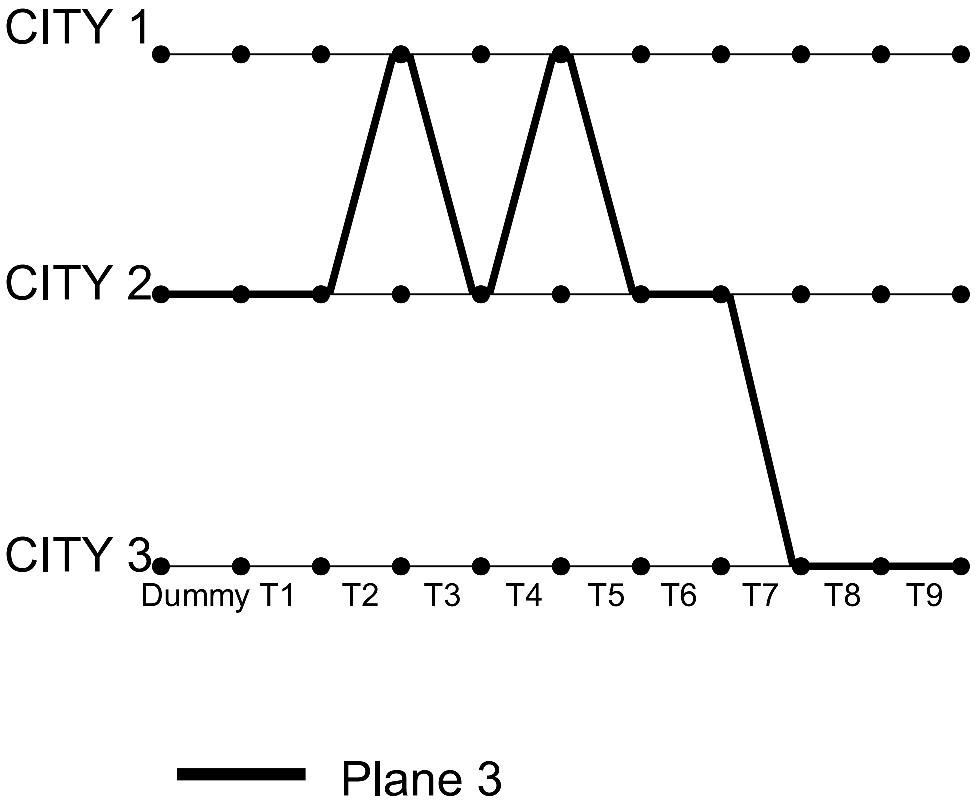
Dumm	T1	T2	T3	T4	T5	T6	T7	T8	T9
10	12	23	31	12	23	30	32	21	10

Figure 3.12 Route of Plane 1 in the network



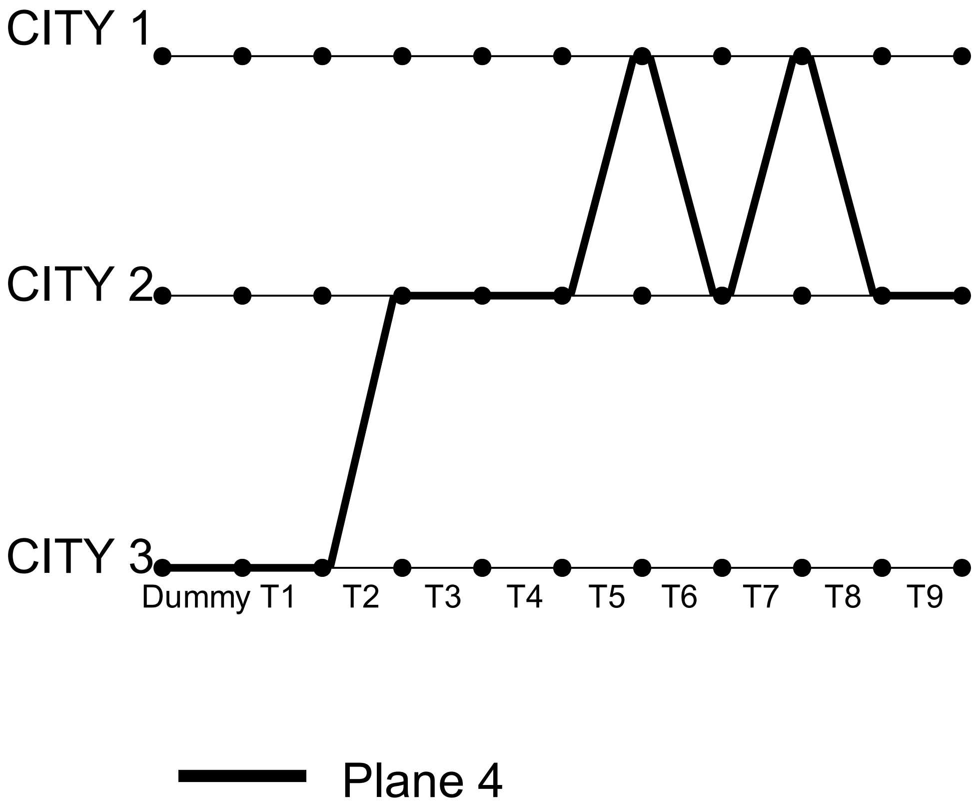
Dumm	T1	T2	T3	T4	T5	T6	T7	T8	T9
20	21	12	21	13	32	21	12	20	20

Figure 3.13 Route of Plane 2 in the network



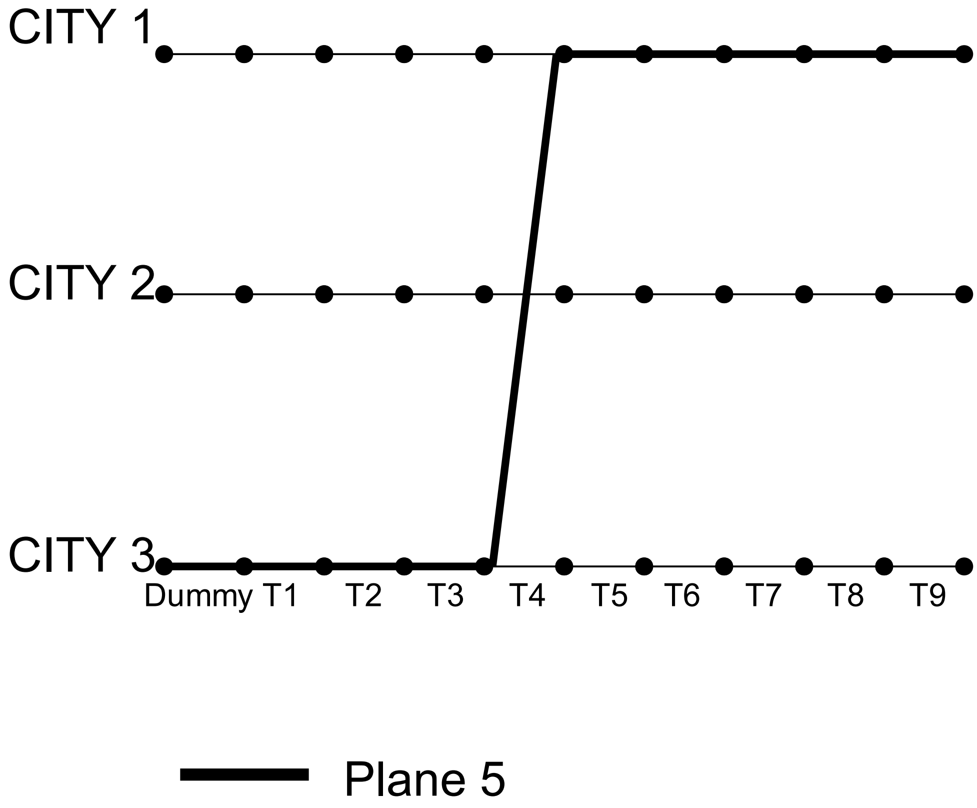
Dumm	T1	T2	T3	T4	T5	T6	T7	T8	T9
20	20	21	12	21	12	20	23	30	30

Figure 3.14 Route of Plane 3 in the network



Dumm	T1	T2	T3	T4	T5	T6	T7	T8	T9
30	30	32	20	20	21	12	21	12	20

Figure 3.15 Route of Plane 4 in the network



Dumm	T1	T2	T3	T4	T5	T6	T7	T8	T9
30	30	30	30	31	10	10	10	10	10

Figure 3.16 Route of Plane 5 in the network

Chapter 4

Results and Analysis

4.1 Matlab Program

The algorithm described in chapter three was coded in matlab version 6.1.0.450 release 12. The program is capable of solving a network of 99 cities. The features of the program are given below.

1. The program will run the schedule for a period of 7 days.
2. Inputs required are the number of cities and the number of planes. Different demands and demand patterns are assigned by the program to different O-D pairs.
3. The output of the program provides
    a. daily utilization of the planes and the entire fleet,
    b. the number of dead legs,
    c. the number of flights not scheduled,
    d. number of dissatisfied customers,
    e. average waiting time for the network,
    f. number of flights scheduled,
    g. number of customers served, and
    h. number of customers lost.

4.2 Experiments

A list of scenarios was run each with a different number of cities and varying fleet size. The experiments were conducted for a network of 3, 6, 8, 10, 12, 15,18, and 20 cities with the fleet size changing as a percent of the O-D pairs. The fleet size was 20%, 50%, 60%, 70%, 75%, 80%, 85%, 90%, 95%, or 100% of the O-D pairs. For example, for a network of 6 cities, 30 O-D pairs, the fleet size is a percent of 30. Table 4.1 shows the fleet size for different numbers of cities.

Table 4.1 Fleet size for different number of cities

Fleet Size (% of O-D pairs) **Number of Cities**	**20%**	**50%**	**60%**	**70%**	**75%**	**80%**	**85%**	**90%**	**95%**	**100%**
3	1	3	4	-	5	-	-	-	6	-
6	6	15	18	21	23	24	26	27	29	30
8	11	28	34	39	42	45	48	50	53	56
10	18	45	54	63	68	72	77	81	86	90
12	26	66	79	92	99	106	112	119	125	132
15	42	105	126	147	158	168	179	189	200	210
18	61	153	184	214	230	245	260	275	291	306
20	76	190	228	266	285	304	323	342	361	380

The Table 4.2 shows the actual experimental experiments examined.

Table 4.2 List of Experiments

Experiment	Cities	Planes
1	3	1
2	3	3
3	3	4
4	3	5
5	3	6

Experiment	Cities	Planes
6	6	6
7	6	15
8	6	18
9	6	21
10	6	23
11	6	24
12	6	26
13	6	27
14	6	29
15	6	30

Experiment	Cities	Planes
16	8	11
17	8	28
18	8	34
19	8	39
20	8	42
21	8	45
22	8	48
23	8	50
24	8	53
25	8	56

Experiment	Cities	Planes
26	10	18
27	10	45
28	10	54
29	10	63
30	10	68
31	10	72
32	10	77
33	10	81
34	10	86
35	10	90

Experiment	Cities	Planes
36	12	26
37	12	66
38	12	79
39	12	92
40	12	99
41	12	106
42	12	112
43	12	119
44	12	125
45	12	132

Experiment	Cities	Planes
46	15	42
47	15	105
48	15	126
49	15	147
50	15	158
51	15	168
52	15	179
53	15	189
54	15	200
55	15	210

Experiment	Cities	Planes
56	18	61
57	18	153
58	18	184
59	18	214
60	18	230
61	18	245
62	18	260
63	18	275
64	18	291
65	18	306

Experiment	Cities	Planes
66	20	76
67	20	190
68	20	228
69	20	266
70	20	285
71	20	304
72	20	323
73	20	342
74	20	361
75	20	380

4.3 Output

The specific output from the algorithm is described below.

- Fleet Utilization: The fleet utilization is calculated as the ratio of the actual flying time of the fleet to the total time of the network. The total time in this case is 18 hours per day for 7 days i.e. 7560 min.

$$\text{Fleet Utilization} = \frac{\sum_{i=1}^{7} \sum_{j=1}^{G} \sum_{k=1}^{9} T_{ijk}}{7560}$$

where, i = days

j = planes

k= time units (segments into which the day is divided)

T_{ijk} = block time of plane j in time unit k on day i.

G = total number of planes flown

- Dead Legs: The number of flights flown empty.
- Flights Not Scheduled: The number of flights which could not be flown due to unavailability of planes. The unavailability arises due to a shortage of planes in a larger network size.
- Number of Dissatisfied Customers: The number of customers that could not be flown in the time unit that they requested to fly.
- Average Waiting Time of customers (including satisfied customers): Assuming that the dissatisfied customers wait for one hour.

$$Av.Wait\ Time(hours) = \frac{\sum_{i=1}^{7}\sum_{k=1}^{9} D_{ik} * 1}{\sum_{i=1}^{7}\sum_{k=1}^{9} P_{ik}}$$

where, i = days

k= time units (segments into which the day is divided)

D_{ik} = number of dissatisfied customers

P_{ik} = total number of customers

- Scheduled Flights: The flights that were flown with passengers over the network excluding the dead legs.
- Customers Served: The number of passengers that were flown in the network.
- Passenger Load Factor:

$$Passenger\ Load\ Factor = \frac{S}{4F}$$

where, S = customers served

4F = total number of seats available (seats per flight * scheduled flights)

- Flights not scheduled: The number of flights that were flown to satisfy the passengers that could not fly by their scheduled flights due to overbooking.
- Failures: The number of passengers who could not be flown in a particular day.

4.4 Results

Appendix A contains the results from the experiments explained above.

4.4.1 Analysis

4.4.1.1 Fleet Utilization

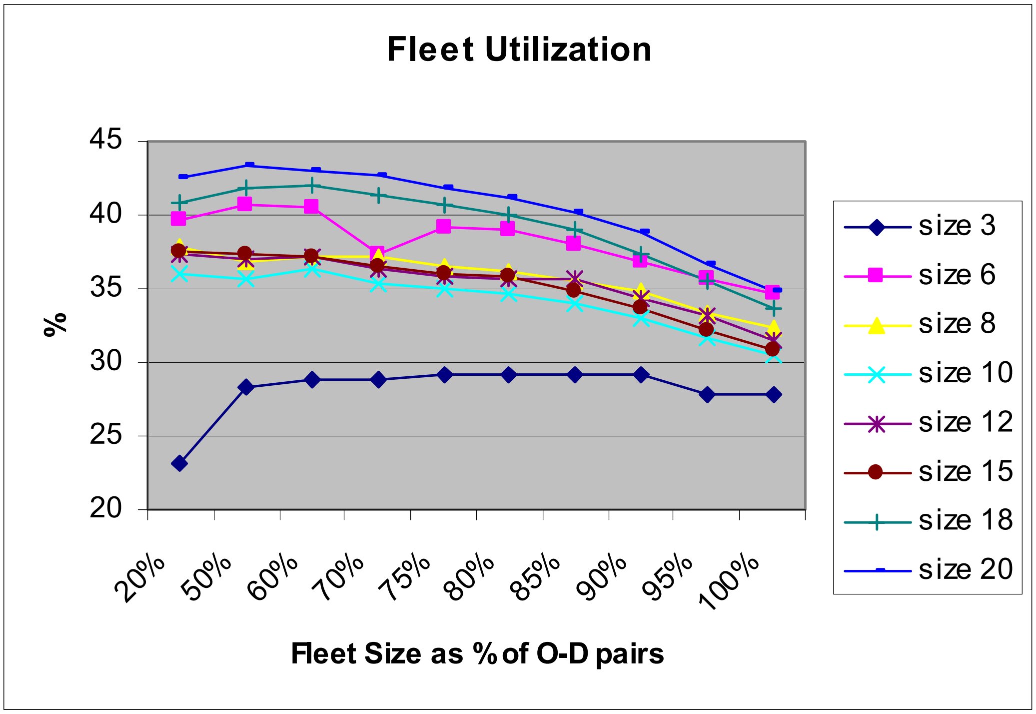


Figure 4.1 Fleet Utilization

Figure 4.1 gives the behavior of fleet utilization for the various experiments performed. It is observed that the utilization increases slightly as the fleet size increases from 20% to 50%. However, the fleet utilization decreases as we increase the size above 85 – 90%.

The low utilization for small fleet size (20%) can be due to the fact that the fleet has to stay at one particular city to satisfy the requests in the next time unit. The probability of the fleet flying a dead leg is very low in such a case. The increase in the fleet size from 50% to 100% results in a decrease in the utilization.

4.4.1.2 Dead Legs

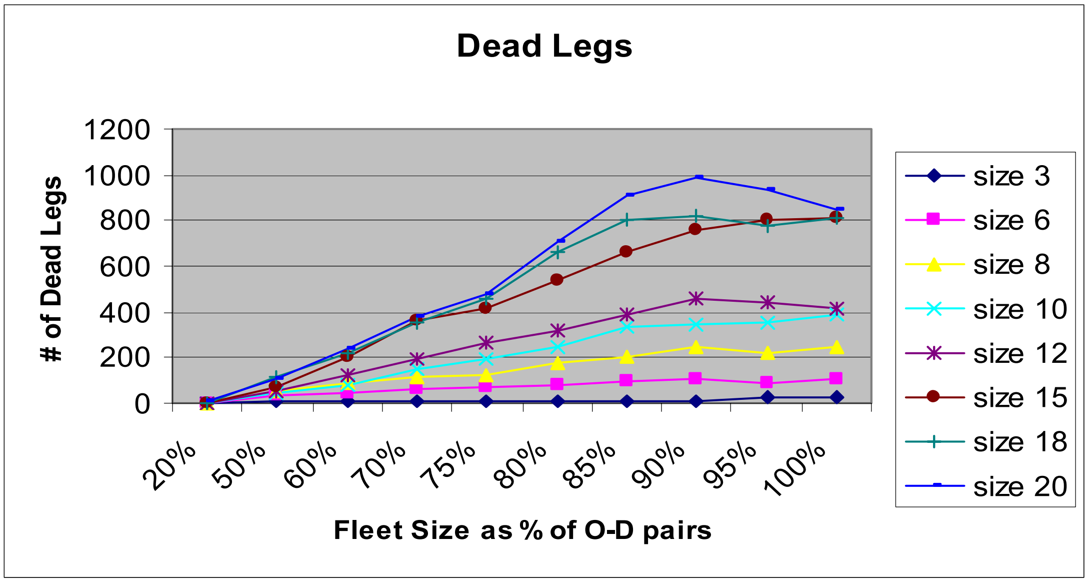


Figure 4.2 Dead Legs

Figure 4.2 shows the change in the number of dead legs with an increase in the fleet size. The different lines resemble the different network size. The number of dead legs increases as the fleet size increases until the fleet is approximately 90% of the O-D pair. A further increase in fleet size reduces the number of dead legs.

4.4.1.3 Scheduled Flights

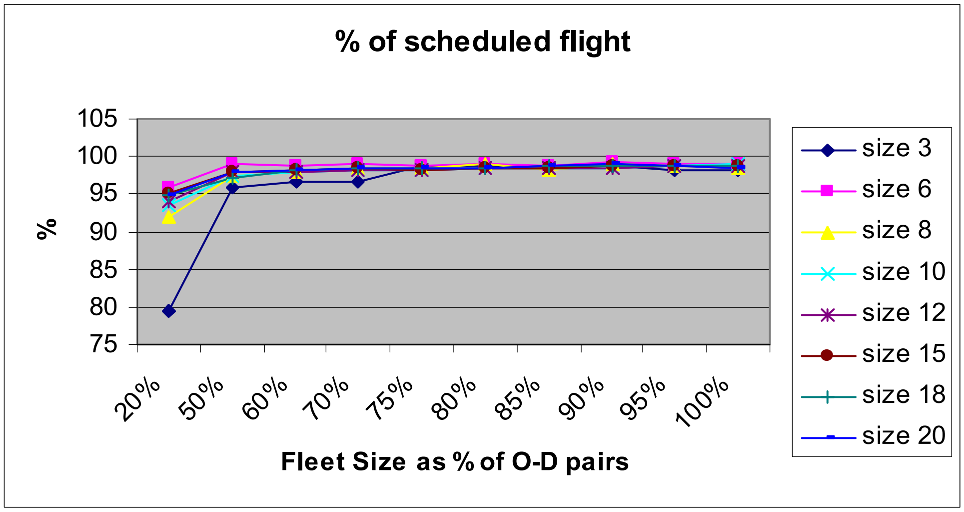


Figure 4.3 Scheduled Flights

The number of flights scheduled increases with the increase in the fleet size. The percentage jumps as the fleet size increases from 20% to 50% of the O-D pair, and it shows a slight increase as the fleet size is further increased to 100%.

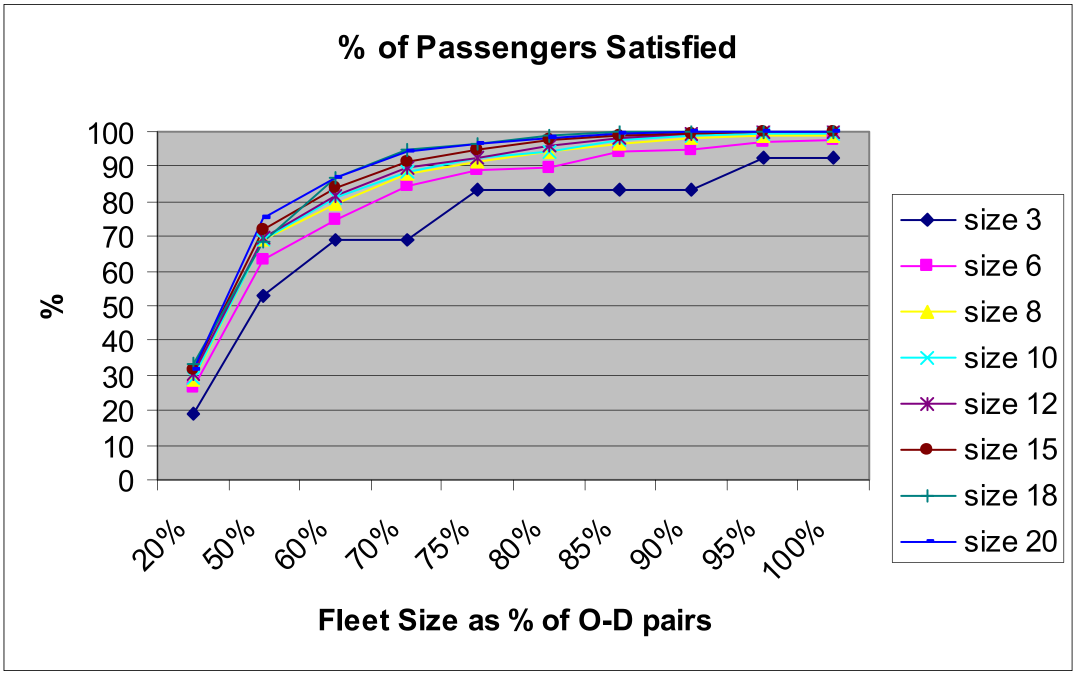


Figure 4.4 Passengers Satisfied

The percent of passengers satisfied is directly proportional to the percent of scheduled flights. It can be observed that for most fleet sizes, the percent of passengers satisfied crosses 90% when the fleet size reaches 95% of the O-D pairs. The percentage is also very close to 100% for any fleet size above 90% of O-D pair.

It is very important to observe that the rate of increase of passengers satisfied drop as we increase the fleet size. Thus, it is important to understand the cost of increase in an fleet size as compared to the cost of losing a passenger. The increase in the number of dead legs increases the cost substantially. A cost estimate of an additional plane in the fleet and the cost of flying dead legs can help in this decision.

4.4.1.4 Passenger Load Factor

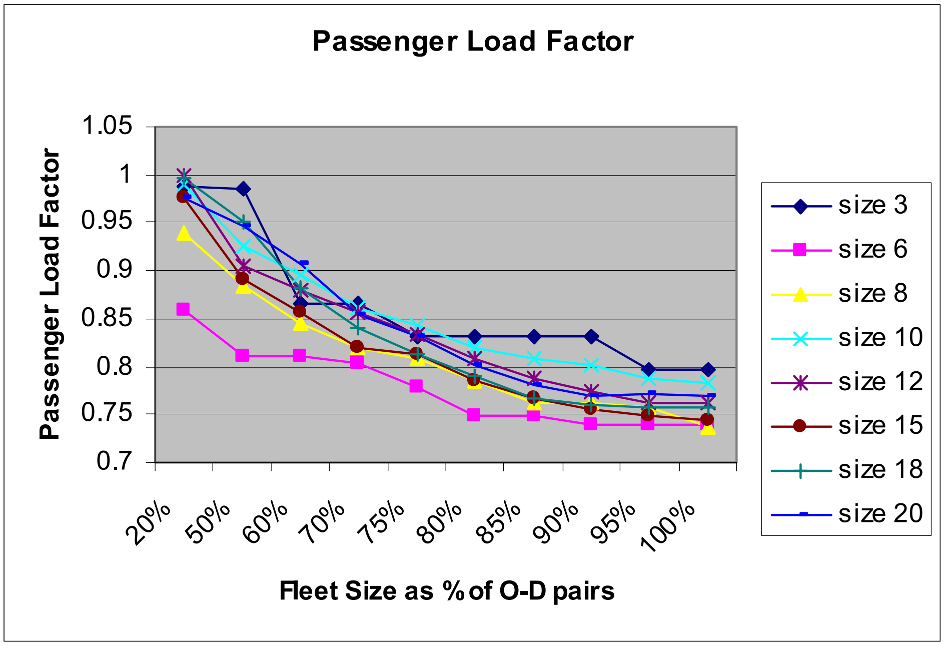


Figure 4.5 Passenger Load Factor

The passenger load factor is the fraction of available seats that are actually occupied by the passengers in a flight.

Observe that the passenger load factor decreases as the fleet size increases. This shows that with an increase in fleet size, the flights are not completely booked. The decrease in the passenger load factor is the result of satisfying more passengers in the requested time unit, yielding a decrease in dissatisfied customers.

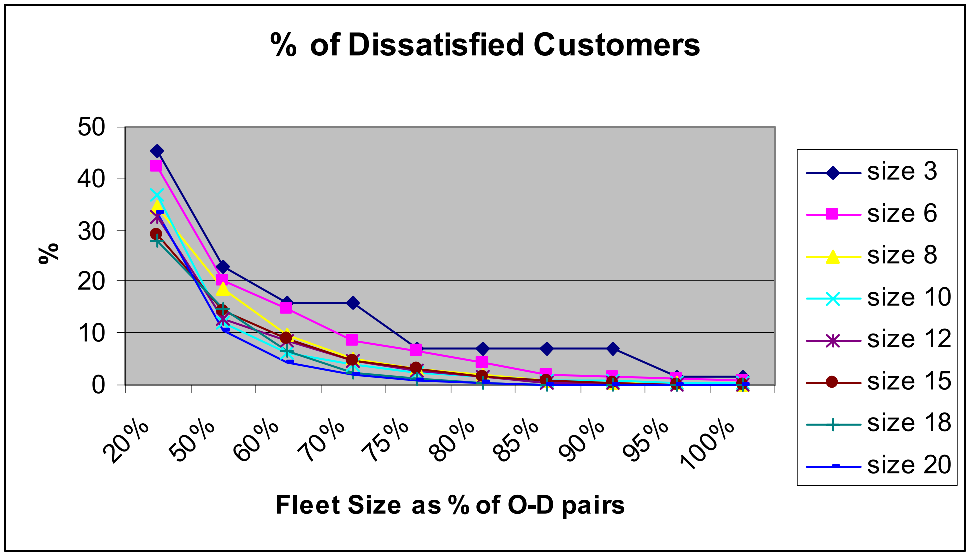


Figure 4.6 Dissatisfied Customers

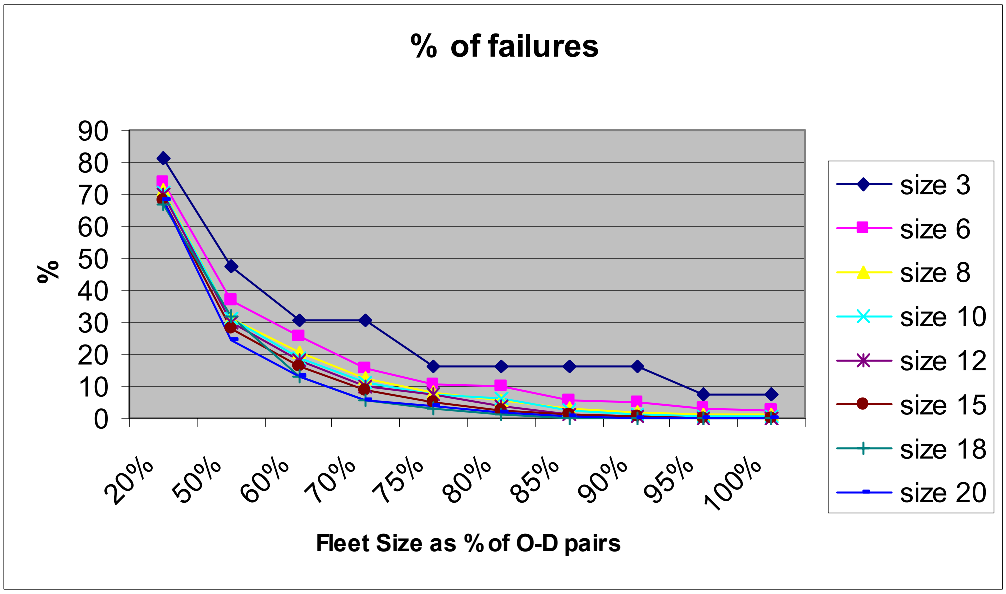


Figure 4.7 Percentage of Failures

An increase in the fleet size also reveals substantial decrease in the number of dissatisfied customers and a decrease in customers lost. Figures 4.6 and 4.7 illustrate these trends.

4.4.2 Cost Analysis

The cost of operation includes various factors like maintenance, fuel, and other activities. Some of these costs are taken into consideration in this cost analysis.

4.4.2.1 Cost of new aircraft

The purchase price of a new aircraft is $ 1,170,000. For the purpose of analysis, this cost is divided into cost per week with the assumption that the aircraft has a life of 5 years and no salvage value.
(Note: This analysis does not consider inflation or the time value of money)

$$Cost\ of\ aircraft\ per\ week = \frac{1170000}{Life\ of\ \ Aircraft(in\ years)*52}$$

4.4.2.2 Pilot Salary

The annual salary of a pilot is $ 60000. Thus, the weekly salary of the pilot is

$$Pilot\ salary\ per\ week = \frac{60000}{52}$$

4.4.2.3 Cost of fuel

$$Fuel\ Expense = Fuel\ Consumption(Gallons)\ per\ hour * Jet\ fuel\ \cos t\ per\ gallon$$

Fuel Consumption can be calculated from Figure 4.8., and jet fuel cost per gallon = $ 2.71.

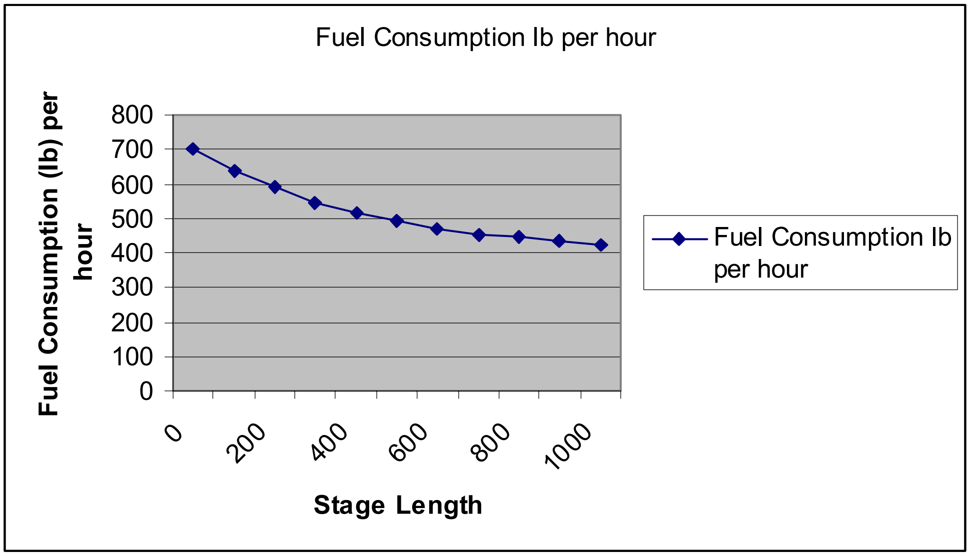


Figure 4.8 Fuel Consumption (lb) per hour

4.4.2.4 Maintenance Cost

Maintenance Hours per Flight Hour = 0.84

Maintenance Labor Expense per Hour = 64.39

Maintenance cost per hour = Maintenance Hours per Flight Hour * Maintenance Labor Expense per Hour = 54.09

4.4.2.5 Miscellaneous Cost

Miscellaneous expenses include service during the flight.

Miscellaneous Trip Expenses = 65

Schedule Parts Expense = 50

4.4.2.6 Cost

Total Variable Expense per hour = Fuel Cost per hour + Maintenance Cost + Miscellaneous Cost

Total Fixed Expense per hour = Weekly Cost of purchasing aircraft + Weekly salary expense

4.4.2.7 Cost of Dead leg

The cost of a dead leg = total variable expense * time of dead leg.

4.4.2.8 Cost of losing a passenger

For the purpose of an example, the cost of losing a passenger is considered to be $100 per customer.

Figures 4.9 and 4.10 show the trend in variable and fixed cost. Figure 4.9 shows that the variable cost increases rapidly with initial increase in fleet size and gradually flattens out as the fleet size increases above 90%. This is because utilization and miles flown reach a steady figure as the fleet size increases.

Figure 4.10 shows that the aircraft cost and the salary increase steadily with an increase in fleet size. This is primarily because the increase in fleet size increases the aircraft expense and also the pilots required.

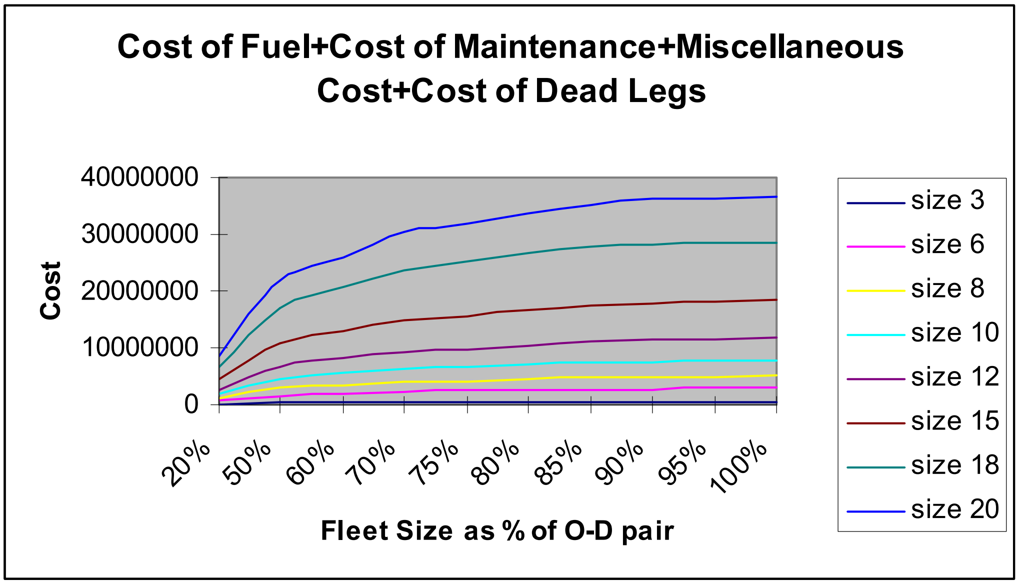


Figure 4.9 Fuel Cost, Maintenance Cost, Miscellaneous Cost, Dead leg Cost

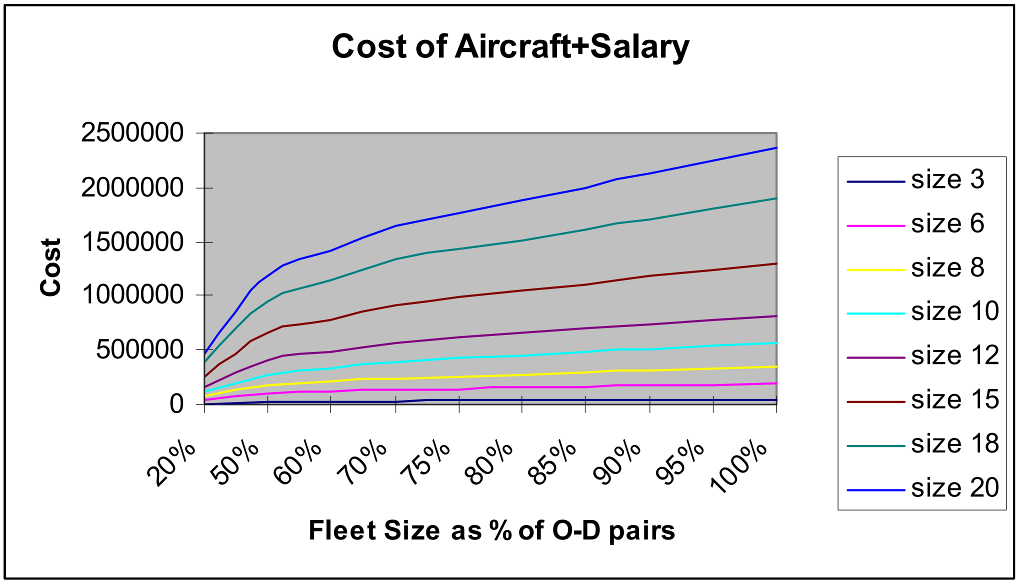


Figure 4.10 Aircraft Cost, Salary

The number of customers lost dips as we increase fleet size. Thus, we observe a negative exponential curve in the cost with increase in fleet size as seen in figure 4.11.

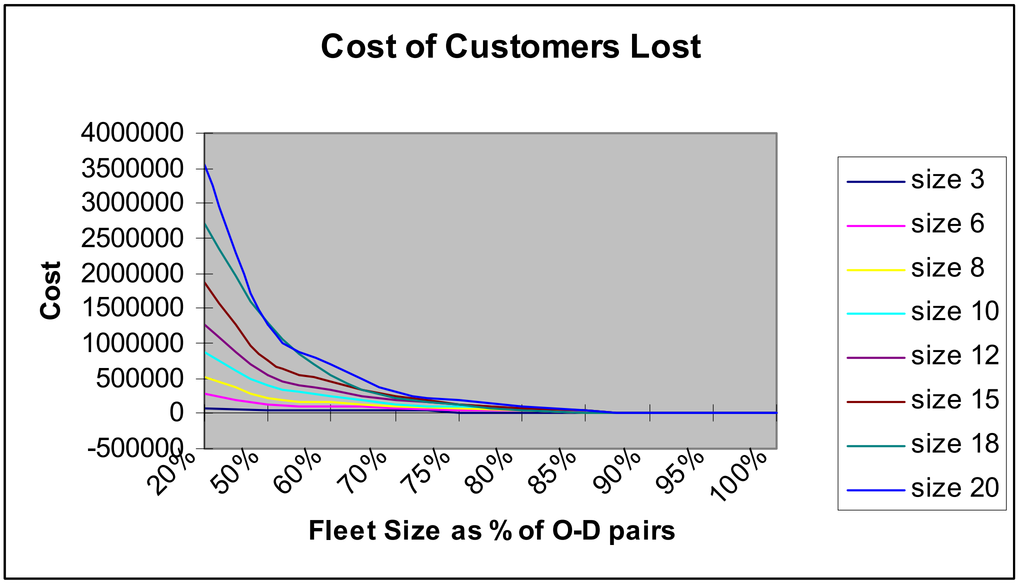


Figure 4.11 Cost of customers lost

Total Cost = (Total variable expense per hour + Total fixed expense per hour) * Total hours flown by an aircraft * Number of aircrafts + Total cost of dead legs + Total Cost of customers that could not be served.

Dividing the total cost by the total number of customers served yields the average cost per seat. This is the required average revenue per seat for break even.

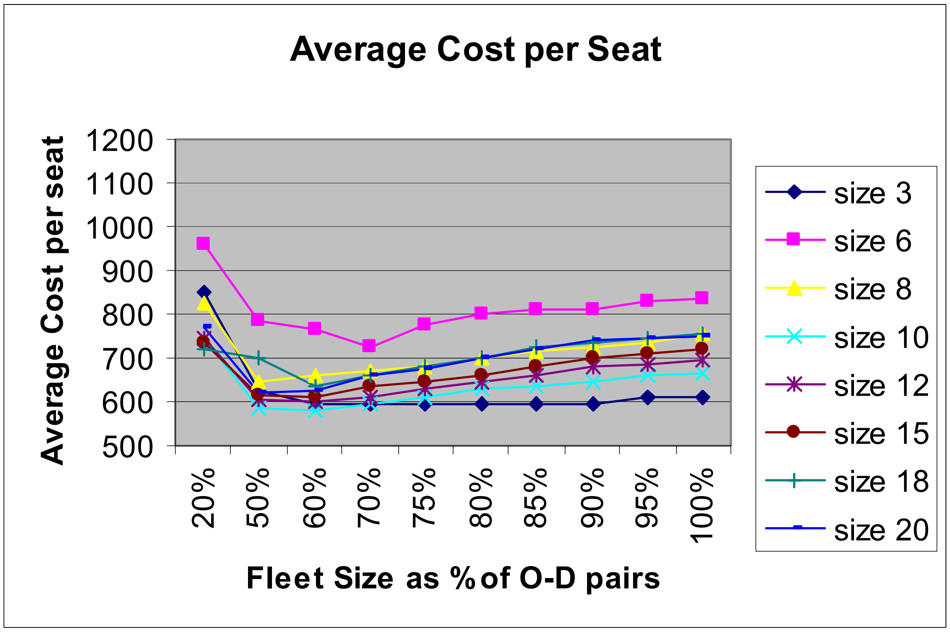


Figure 4.12 Average Cost per seat

Figure 4.12 shows the trend in cost per seat the fleet size increases, for different network sizes.

There is a drop in the average cost per seat for an increase in fleet size to a certain point. However, as the fleet size increases further, the cost starts increasing again.

		Number of cities							
		3	6	8	10	12	15	18	20
Fleet size as % of O-D pairs	20%	1030.124	1239.709	1059.511	945.8711	948.0045	943.3603	930.7705	993.2818
	50%	828.3884	1075.499	**879.117**	**795.3955**	**826.6426**	**839.3287**	961.2937	**856.7734**
	60%	**807.4545**	1060.806	910.7221	800.4619	828.007	843.998	**886.1711**	874.2403
	70%	**807.4545**	**1004.515**	928.5701	822.798	845.8232	878.2122	920.0454	921.0781
	75%	811.2855	1078.499	941.1819	843.9341	868.7175	893.5895	952.273	946.9334
	80%	811.2855	1111.837	967.4817	866.9192	891.872	916.8493	979.1738	978.6222
	85%	811.2855	1123.607	989.9927	877.8199	915.3914	942.0463	1008.535	1006.39
	90%	811.2855	1124.285	1000.67	887.9608	935.6071	961.6182	1023.604	1028.007
	95%	833.0996	1147	1013.055	903.8172	947.0451	973.3606	1034.557	1033.803
	100%	833.0996	1153.722	1038.524	913.7886	957.7321	983.858	1039.977	1039.789

Table 4.3 Average Cost / Seat

Table 4.3 shows that the minimum cost per seat is obtained when the fleet size ranges from approximately 55% to 70% of the O-D pairs. It is clearly indicated that a larger fleet size requires more cost per seat to break even.

For a very small fleet size, the cost per seat is high. The dominant reason is the number of customers lost for this small fleet size. Even though the passenger load factor is high and the fixed cost is low, the percentage of customers lost is very high.

The possible cause of the rise in the cost as the fleet size increases beyond 70% of O-D pairs is the rate at which the dead legs change. Also, the passenger load factor decreases indicating that flights are not flying full.

4.5 Sensitivity Analysis on Cost of Losing a Customer

4.5.1 Average cost per seat

Average cost per seat is calculated for various penalty fees. The change in average cost per seat is observed for penalty fees ranging from \$100 to \$ 1200 with an increment of \$100.

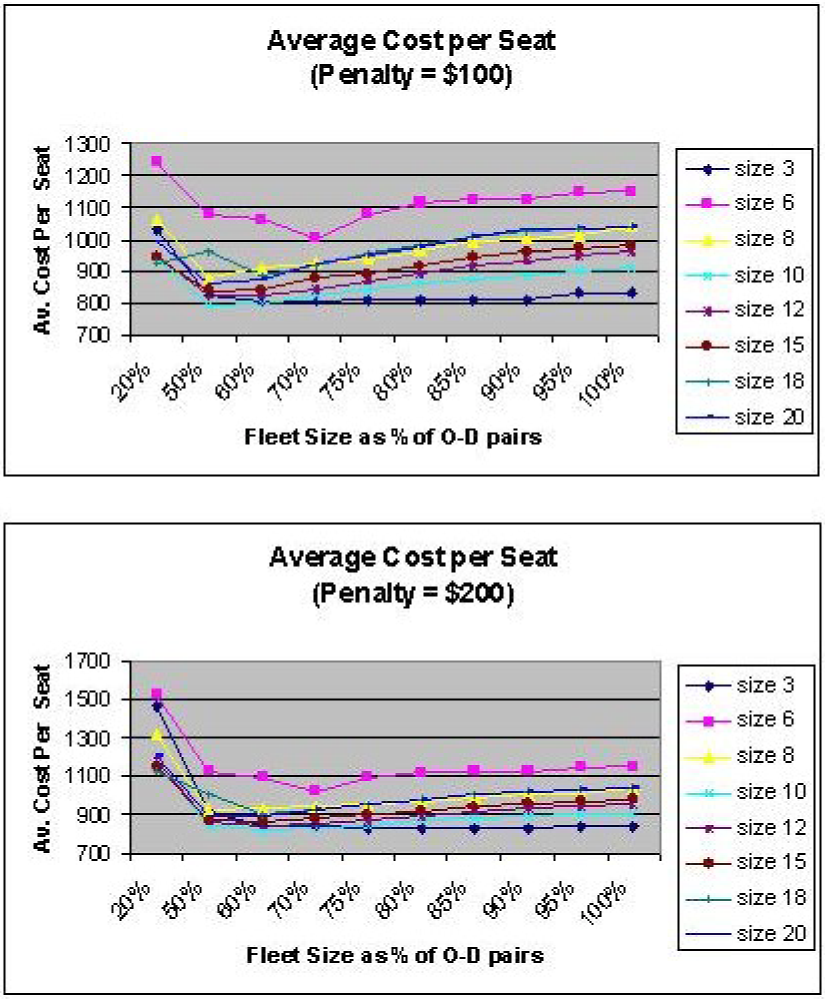
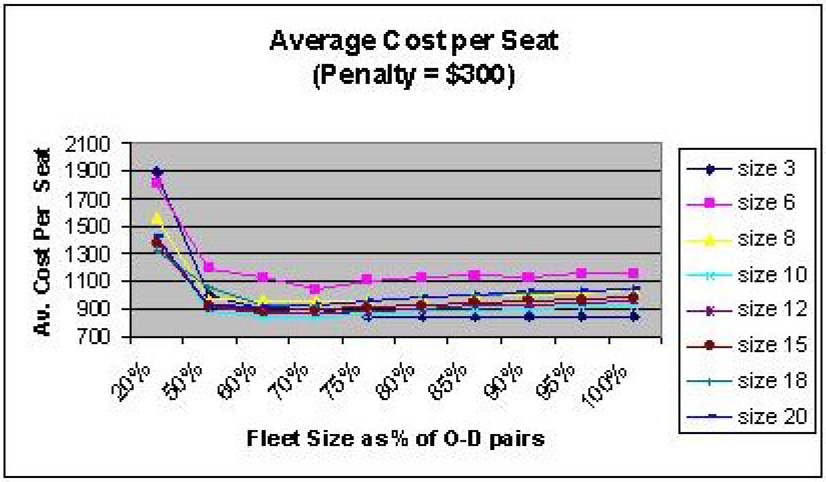


Figure 4.13 Average Cost per Seat (Penalty of $100 and $ 200)

Figure 4.13 shows that the average cost per seat varies in a specific pattern. The cost per seat decreases at a very fast rate as the fleet size increases from 20% to 50%. Further increase in fleet size shows a slow growth in the average cost. Primary factors of this increase can be the increase in the fleet size and decrease in the passenger load factor.

A hypothesis stating that the pattern of the change in the average cost per seat should remain same can be observed for higher values of penalty fees.



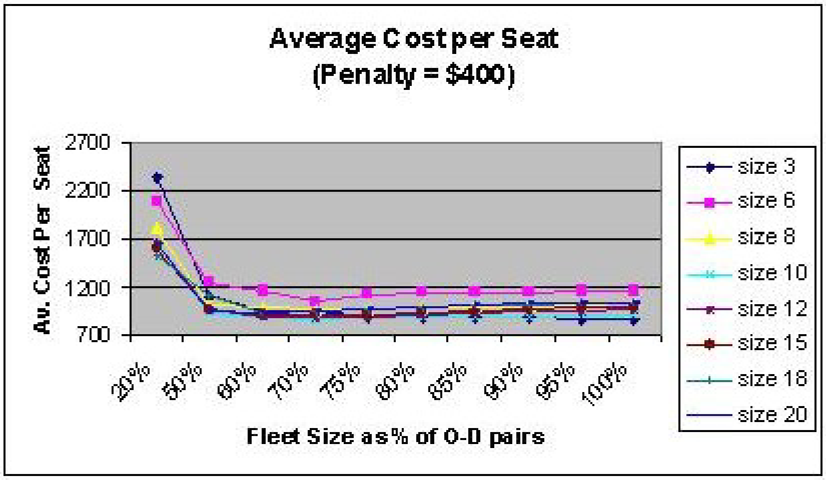
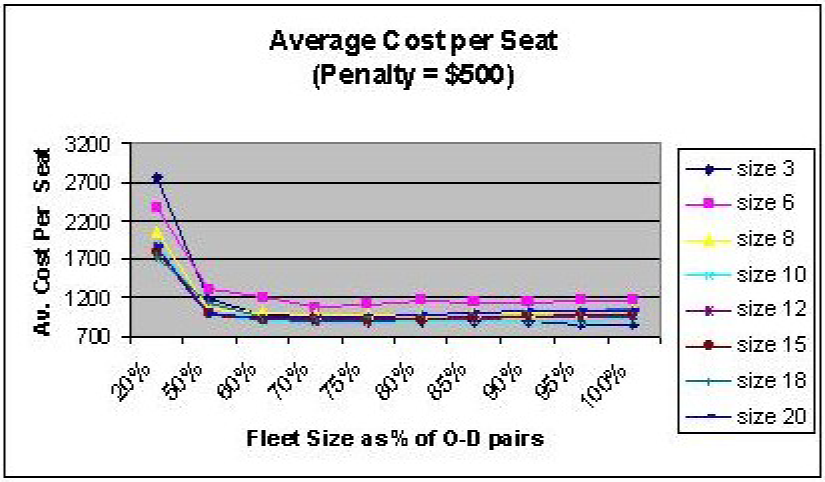


Figure 4.14 Average Cost per Seat (Penalty of $300 and $ 400)



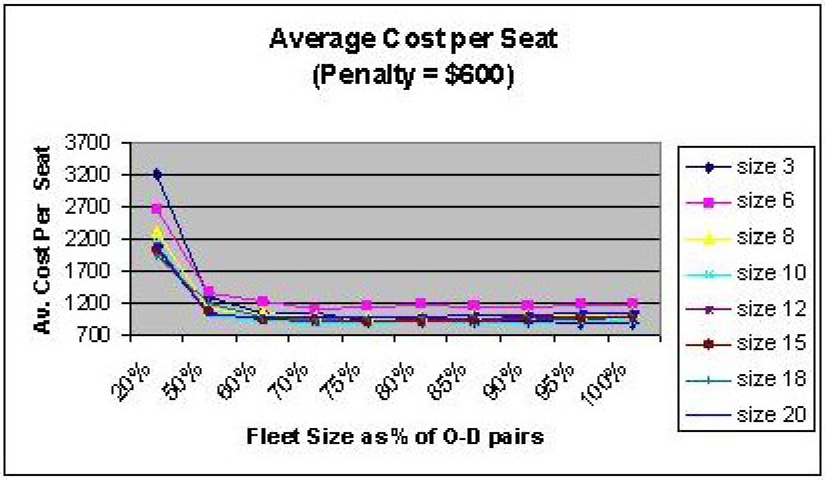
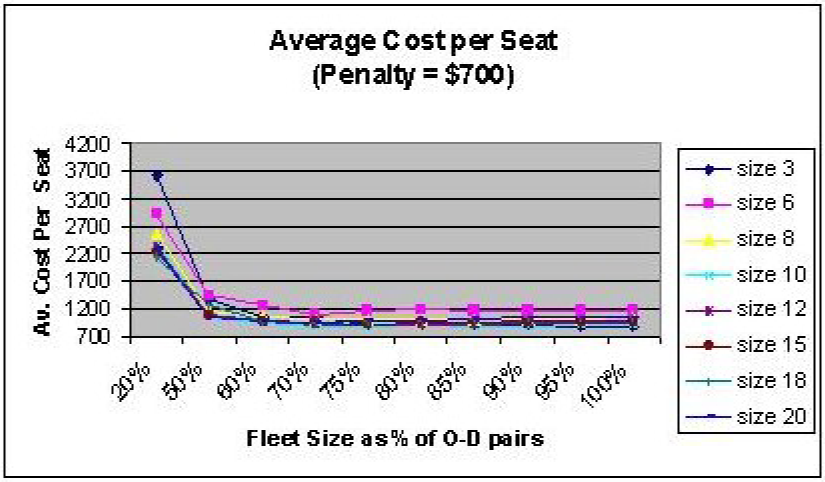


Figure 4.15 Average Cost per Seat (Penalty of $500 and $ 600)



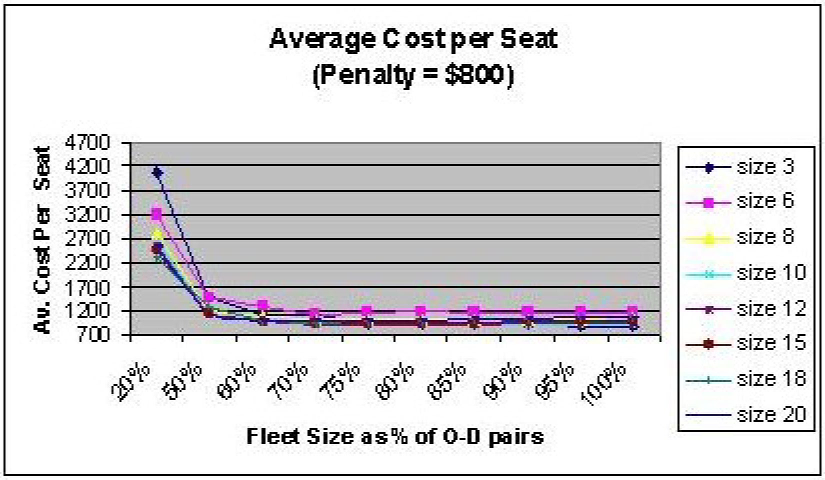


Figure 4.16 Average Cost per Seat (Penalty of $700 and $ 800)

Figures 4.14 – 4.16 support the hypothesis of displaying a similar pattern in change of average cost per seat.

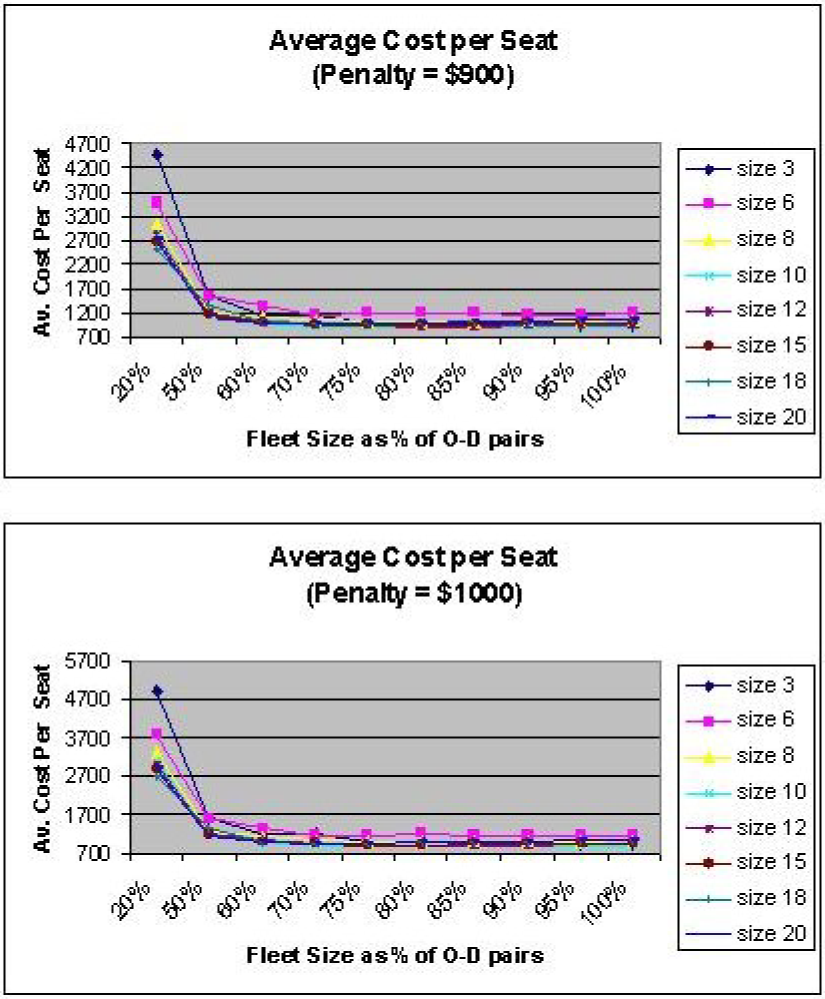


Figure 4.17 Average Cost per Seat (Penalty of $900 and $ 1000)

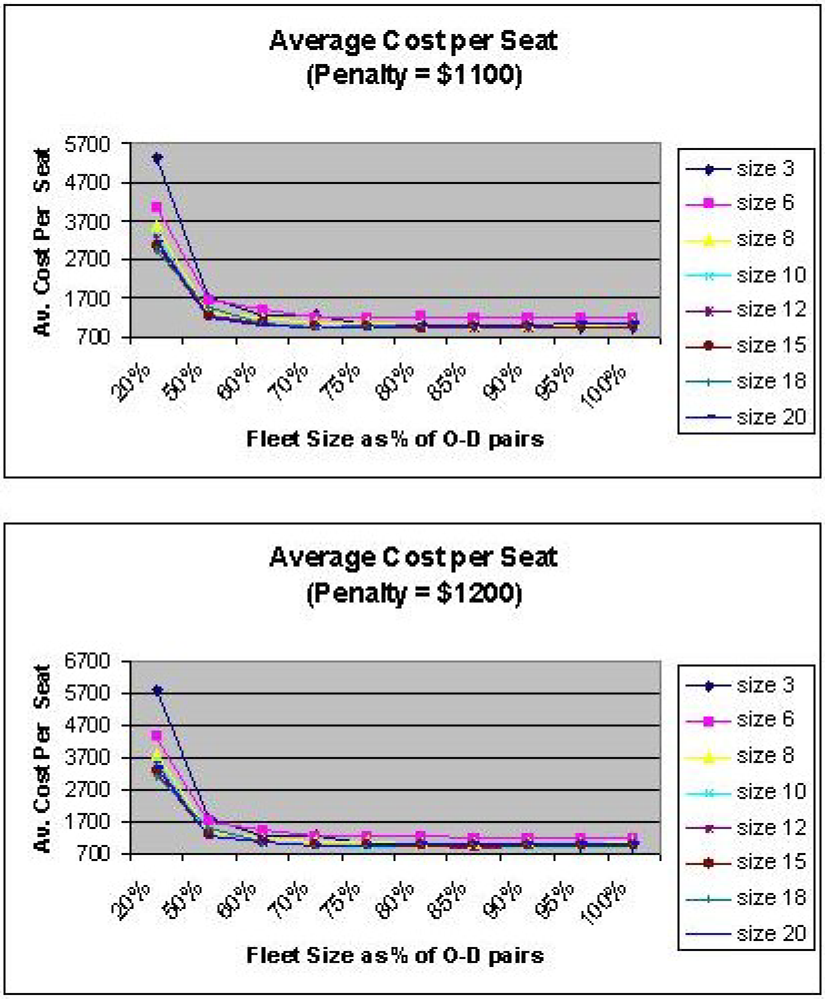


Figure 4.18 Average Cost per Seat (Penalty of $1100 and $ 1200)

Figures 4.17 – 4.18 also show that the cost remains steady with increase in fleet size from 50% to 100%.

Thus, the cost is observed to behave similarly with an assumption that each flight will fly equal miles. The total miles flown by different network sizes and different fleet sizes is different.

4.5.2 Average cost per seat per mile

Average cost per seat per mile is calculated by considering the entire fleet size.

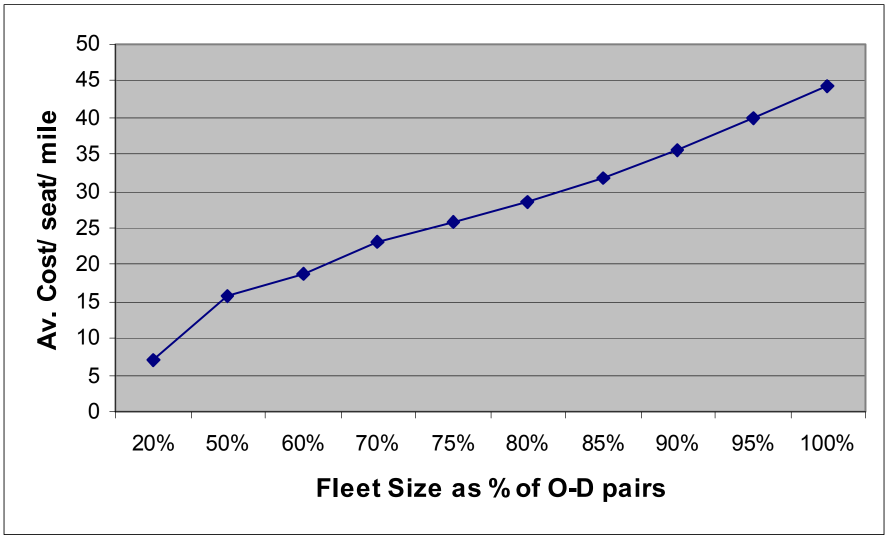


Figure 4.19 Average Cost per Seat per mile

Figures 4.19 illustrates that the average cost per seat per mile increases with increase in fleet size.

4.6 Fleet Size

The factors that help to determine a good fleet size can be divided into positive factors and negative factors. The positive factors are the factors that should be maximized for better performance of the network e.g. customer satisfaction, fleet utilization, passenger load factor. The negative factors are the factors that should be minimized for better performance of the network e.g. cost, failures, dead legs, wait time.

To find the fleet size, the following steps were carried out.

For positive and negative factors:
Step 1: The maximum value of each factor is determined. This value is set as 100%.
Step 2: By considering the maximum as 100%, the percentage value for each fleet size for each factor is determined.

Fleet Size (% of O-D Pairs)	Fleet Utilization	Customer Satisfaction	Passenger Load Factor	Average
20	97.53	29.19	100.00	75.57241
50	99.60	68.37	93.66	87.21084
60	100.00	81.51	90.17	90.56004
70	97.72	88.85	86.93	91.16423
75	97.32	93.57	84.84	91.90839
80	96.39	95.43	82.15	91.3237
85	94.74	97.47	80.41	90.87326
90	92.00	98.21	79.50	89.90563
95	87.99	99.88	78.97	88.94487
100	84.73	100.00	78.52	87.75019

Table 4.4 Percentage of positive factors as compared to their maximum value

Fleet Size (% of O-D Pairs)	Cost/seat	Dead legs	Failures	Cost/seat/mile	Wait time	Average
20	100	0.373333	100	15.85566	100	63.2458
50	87.29156	12.98667	44.65067	35.65693	95.41045	55.19926
60	86.66642	26.90667	26.4954	42.31957	88.0597	54.08955
70	88.10803	43.30667	16.5349	51.92626	92.35075	58.44532
75	90.67788	53.78667	9.427376	58.21399	81.34328	58.68984
80	92.99694	72.90667	6.496158	64.60016	80.97015	63.59402
85	94.86363	90.90667	4.098861	71.93454	76.08209	67.57716
90	96.07454	100	3.555381	80.09963	79.4403	71.83397
95	97.4675	97.22667	1.335293	89.98598	99.51493	77.10607
100	98.39144	97.41333	0.985274	100	98.95522	79.14905

Table 4.5 Percentage of negative factors as compared to their maximum value

The difference between the positive and negative values will determine the fleet size performance. A higher positive difference between the two values will illustrate better performance.

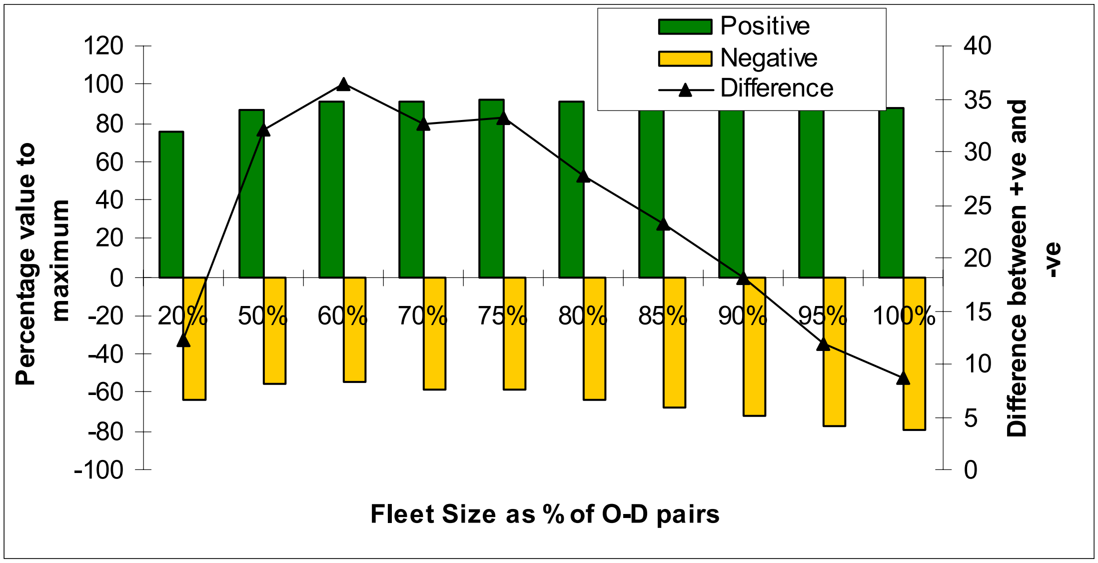


Figure 4.20 Comparison between different fleet sizes

Figure 4.20 illustrates that difference between the positive and negative factors. The fleet size of 60% - 70% of O-D pairs seems to give the best value under the assumed conditions.

Chapter 5

Conclusion

5.1 Summary

The goal of the SATS program is to increase the role of small, general aviation airports in order to effectively triple the current level of aviation system throughput in the future. This thesis provides a basic understanding of aircraft scheduling keeping the quality and quantity as priorities.

The significant contribution of this research was the understanding of how various factors affect the network behavior and affects the schedule and cost. The understanding of the system was provided by two ways. First, the system was tested for various network sizes. The networks used for analysis are built after a thorough understanding of the population and other variables that will randomize the cities to form similar networks. This helped in comparing the correct behavior of a small network to a larger network. The cities within a network have different populations and different daily demands. These variables helped in choosing cities that will represent different types of demand constraints. Second, the system changed the fleet size that served each network. Fleet sizes varied by the percentage of the O-D pairs in the network. The software used to test the system was Matlab.

The program and schedules were run on a daily basis. Once the daily schedules worked correctly, the program was coded to work for more than one day. The continuity of the schedule was the prime challenge in a model that ran for more than one day. Finally, a full schedule was run for a time span of 7 days with different network sizes. Various parameters were changed during the run. Some of these parameters included the actual fleet size, the starting points of the flights and cost of losing a customer.

Cost factors added to the model helped in reviewing the actual feasibility of different networks and fleet sizes. Interesting results were obtained with cost analysis. It was clear that more planes did not seem to improve the performance beyond a certain level. There was some initial improvement in the performance as the fleet size increased from 20% to 50%. However, a fleet size beyond 80% of the O-D pairs did not change the performance, but the cost increased with the increase in the fleet size. It was found that the network gave the best performance when the network size was between 70% - 75% of the O-D pairs.

The analysis performed can be a good reference to understand the actual network behavior. The schedule can be used to find the right fleet size for a particular network.

5.2 Future Research

The model built has limitations on variables. The robustness and reliability of the model can be verified by adding more constraints and more complex networks. The accuracy and precision of the results are limited and are bounded by the assumptions made before actual model was built.

The model can be further enhanced by the use of more complex and wider variety of network properties. Properties like weather, mechanical problems can actually affect the results of the current model. A more robust model incorporating such properties can be compared with this model for verifying the results and observing the sensitivity of such properties.

The model uses one aircraft type for all the networks. This limits the performance of the model. Introducing more than one type of aircraft can give interesting results. New and better aircrafts can definitely cut down cost per mile and also improve average satisfaction. The decision of choosing between the different aircraft types would be challenge for such a model. Different speeds and different service schedules can add to the complexity of the model. Various facts and right assumptions can simplify this task.

The thesis work shows the behavior of networks that are within the radius of 500 nm. This limitation has influenced the results obtained. A strong analysis can be made by expanding the network size in terms of number of cities and in terms of distance as well. A network that has cities in different time zones can prove to be a very interesting analysis. Scheduling flights across time zones will replicate a practical network that flows across U.S.A.

Time zones are as important constraints as weather and mechanical problems. Aircraft require constant mechanical service and check-ups. Also, different types of aircraft have different schedules for servicing. Along with these schedules, the normal time table can be disrupted by weather. The model should be able to minimize the affect of weather delays and should try to adjust the schedule so as to maximize satisfaction.

SATS have an interesting problem of scheduling flights in a network that has dynamic demand. Scheduling flights would also require crew scheduling for the network. Getting the crew to schedule to such a dynamic network problem is interesting. The model can consider constraints like getting the crew back to their base and scheduling work without exceeding the hours worked per day by each crew member.

The model presented in this thesis gives a heuristic approach to solve the scheduling problem. A non heuristic approach which will have all the variables discussed and considered in the model can give a robust model. The time to solve such a problem can also be improved by using more sophisticated tools.

Finally, this thesis provides a basis for a problem that can be represented in several different ways and can be solved by various techniques. The study done for this thesis will be a strong platform for someone who is challenged by this type of a problem. This thesis touches various aspects of scheduling and provides an understanding of a SATS network. Using this study as a base, a sophisticated and robust model can be built to solve this dynamic problem.

References:

Abara J., "Applying Integer Linear Programming to the Fleet Assignment Problem," Interfaces, Vol. 19, No. 4, Jul – Aug. 1989.

Berge M. and Hopperstad C. A., "Demand Driven Dispatch: A Method for Dynamic Aircraft Capacity Assignment, Models and Algorithms," Operations Research, Vol. 41, No. 1, Jan-Feb. 1993.

Clarke L., Johnson E., Nemhauser G. and Zhongxi Zhu Feo, "The Aircraft Rotation Problem," Annals of Operations Research 69, 33-46 (1997).

Elce I., "The Development and Implementation of Air Canada's Long Range Planning Model," AGIFORS X, 1970.

Etschmaier M. M. 1984. "Aircraft Scheduling: The State of Art," XXIV AGIFORS Symposium, Sept. 1984.

Farrell C. M.. "A Modeling and Simulation Approach to the Small Aircraft Transportation System (SATS): Assessing Midair Conflict Risk Under the Free-Flight Paradigm," Virginia Tech, 2001.

Gagnon G., "A Model for Flowing Passengers over Airline Networks," Transportation Science, Vol. 1, No. 3, 1967. 232-248.

Gopalan R. and Talluri R. T., “The Aircraft Maintenance Routing Problem,” Operations Research (1998).

Gunn W. A. 1964. “Airline Systems Simulation,” Operations Research, Vol. 12, 1964, pp. 206 – 229.

Herniter M. E., “Programming in Matlab,” Brooks/Cole Publications, 2001.

Horn Mark, “Fleet Scheduling and Dispatching for Demand-Responsive Passenger Services,” Transportation Science Part-C Vol.10. 2002. 35-63.

http://sats.nasa.gov

Kabbani N. M. and Patty B. W. “Aircraft Routing at American Airlines,” Proceedings of 32nd Annual Symposium of AGIFORS, Budapest, Hungary, 1992.

Keskinocak P. and Tayur S., “Scheduling of Time-Shared Jet Aircraft,” Transportation Science, Vol. 32, No. 3, Aug. 1998.

Kikuchi S. and Donnelly R. A., “Scheduling Demand-Responsive Transportation Vehicles using Fuzzy-Set Theory,” Journal of Transportation Engineering, Vol. 118, No. 3. 1992.

Kikuchi S. and Jong-Ho Rhee, “Scheduling Method for Demand Resposive Transportation System,” Journal of Transportation Engineering, Vol. 115, No. 6, 1989.

Loughran B. P., “An Airline Schedule Construction Model,” AGIFORS.1970.

Mathaisel D. F. X. and De Lamotte H. D., "Fleet Assignment with Variable Demand: A Goal Programming Approach," Paper submitted to AGIFORS 1983.

Miller J. C. 1972. "A Time of Day Model for Aircraft Scheduling," Transportation Science, Vol. 6, No. 3, 1972, pp. 221 – 246.

Miller J. C., "An Aircraft Routing Model for the Airline Firm," American Economist 12, 24-32 (1968).

Newell G., "Dispatching Policies for a Transportation Route," Transportation Science, Vol. 5, No. 1, 1971.

Ronen D., "Scheduling Charter Aircraft," Journal of the Operations Research Society, Vol. 51, No. 3, 258-262. 2002.

Ross S. M., "Introduction to Probability Models," Seventh Edition, Harcourt/ Academic Press, 2000.

Scott P., "Taxi! Taxi!," Popular Science, 12 Dec. 2002.

Simpson R. W., "Computerized Schedule Construction for an Airline Transportation System," Department of Aeronautics and Astronautics. Flight Transportation Laboratory. 1966.

Soudarovich J., "Routing Selection and Aircraft Allocation," AGIFORS XI, 1971.

Swan W., "FA-7: 1977 Version of the Linear Programming Fleet Assignment Model," MIT Flight Transportation Laboratory, Spring 1977.

Teodorovic D. B., “ Airline Operations Research,” New York : Gordon and Breach Science Publishers, 1988.

Teodorovic D. B., “Transportation Networks: A Quantitative Approach,” New York : Gordon and Breach Science Publishers, 1986.

Yan S. and Chung-Rey Wang, “The Planning of Aircraft Routes and Flight Frequencies in an Airline Network Operations,” Journal of Advanced Transportation, Vol. 35, No. 1, 2000. 33-46.

APPENDIX A: Results Obtained by various Program Runs

Number of Cities	Planes	Fleet Utilization	Dead Legs	Flights not scheduled	# of dissatisfied cust	Av. Wait Time	Scheduled flights	Satisfied passengers	Passanger Load Factor	non scheduled flights	Failures
3	1	23.1	0	185	79	1.43	44	174	0.9886364	9	447
	3	28.29	7	142	111	1.32	124	489	0.9858871	5	324
	4	28.86	12	114	102	1.24	185	641	0.8662162	6	251
	4	28.86	12	114	102	1.24	185	641	0.8662162	6	251
	5	29.1	13	72	53	0.97	233	775	0.8315451	3	166
	5	29.1	13	72	53	0.97	233	775	0.8315451	3	166
	5	29.1	13	72	53	0.97	233	775	0.8315451	3	166
	5	29.1	13	72	53	0.97	233	775	0.8315451	3	166
	6	27.8	28	27	14	1.55	270	861	0.7972222	5	67
	6	27.8	28	27	14	1.55	270	861	0.7972222	5	67
6	6	39.66	4	799	424	0.44	291	1000	0.8591065	12	1576
	15	40.72	35	620	487	0.53	740	2402	0.8114865	8	1218
	18	40.42	42	532	422	0.48	875	2836	0.8102857	10	995
	21	37.28	64	405	280	0.51	1000	3219	0.80475	10	745
	23	39.1	73	312	219	0.46	1090	3395	0.7786697	14	587
	24	38.97	83	282	150	0.45	1144	3424	0.7482517	11	517
	26	38.04	95	220	64	0.4	1198	3581	0.7472871	14	374
	27	36.84	105	191	57	0.47	1222	3615	0.7395663	10	322
	29	35.72	91	136	49	0.46	1249	3693	0.7391914	11	220
	30	34.72	107	107	25	0.39	1254	3709	0.7394338	14	98
8	11	37.87	1	1487	718	0.32	554	2082	0.9395307	44	2945
	28	36.85	54	1149	926	0.22	1418	5017	0.8845205	36	2271
	34	37.25	92	864	561	0.19	1717	5798	0.844205	33	1557
	39	37.09	119	615	328	0.24	1953	6412	0.8207885	32	1095
	42	36.54	121	491	213	0.24	2068	6688	0.8085106	32	832
	45	36.15	173	336	134	0.23	2198	6895	0.7842357	21	537

	48	35.52	206	192	61	0.19	2315	7067	0.7631749	41	319
	50	34.79	249	112	23	0.18	2350	7156	0.7612766	24	194
	53	33.37	217	94	20	0.16	2380	7210	0.7573529	29	169
	56	32.29	245	46	14	0.23	2453	7227	0.7365471	38	75
10	18	36	0	2430	1374	0.14	940	3724	0.9904255	60	4924
	45	35.64	48	1750	1048	0.12	2348	8685	0.9247232	63	3205
	54	36.29	79	1301	630	0.1	2852	10222	0.8960379	57	2224
	63	35.34	147	901	422	0.12	3254	11199	0.8604026	62	1517
	68	35.06	194	667	263	0.14	3460	11651	0.8418353	60	1091
	72	34.6	244	517	189	0.13	3612	11858	0.8207364	59	854
	77	33.97	332	291	113	0.13	3808	12298	0.8073792	60	465
	81	32.98	348	183	103	0.13	3887	12450	0.8007461	61	327
	86	31.67	354	110	37	0.14	3977	12527	0.7874654	46	174
	90	30.46	387	46	30	0.12	4002	12542	0.7834833	45	74
12	26	37.31	2	3515	1777	0.1	1366	5461	0.999451	82	6857
	66	37	54	2457	1593	0.08	3473	12585	0.9059171	70	4417
	79	37.22	123	1878	1238	0.09	4184	14738	0.8806166	82	3216
	92	36.31	193	1337	753	0.1	4735	16233	0.857075	84	2143
	99	35.78	264	1025	485	0.09	5023	16742	0.833267	86	1713
	106	35.62	316	669	267	0.09	5359	17328	0.8083598	75	1057
	112	35.59	391	317	74	0.08	5652	17789	0.7868454	85	453
	119	34.41	462	144	36	0.1	5797	17962	0.7746248	81	219
	125	33.1	440	70	17	0.09	5910	18009	0.761802	79	97
	132	31.54	411	25	7	0.08	5913	18020	0.7618806	80	36
15	42	37.55	2	5505	2537	0.09	2230	8716	0.97713	107	10255
	105	37.34	70	3761	2861	0.11	5536	19727	0.8908508	109	6444
	126	37.12	202	2947	2046	0.1	6678	22907	0.8575547	115	4941
	147	36.5	360	2031	1132	0.09	7625	25037	0.8208852	107	3235
	158	36.04	412	1515	802	0.09	8020	26040	0.8117207	138	2261
	168	35.77	536	922	368	0.1	8515	26758	0.7856136	125	1338
	179	34.81	665	438	181	0.1	8824	27100	0.7677924	125	655

	189	33.73	759	200	72	0.1	9036	27283	0.7548417	109	297
	200	32.09	801	66	40	0.1	9137	27322	0.7475648	115	88
	210	30.77	815	24	2	0.11	9195	27360	0.7438825	119	30
18	61	40.77	0	7916	3758	0.08	3390	13532	0.9979351	177	14511
	153	41.89	113	5448	4097	0.09	7297	27724	0.9498424	202	9120
	184	42.02	219	4119	2270	0.08	9953	35147	0.8828243	180	6149
	214	41.28	352	2382	842	0.1	11401	38377	0.8415271	181	3151
	230	40.68	462	1644	400	0.1	12068	39277	0.8136601	177	2077
	245	39.97	659	782	141	0.1	12688	40053	0.7891906	184	910
	260	39.08	799	243	18	0.09	13185	40474	0.7674251	179	263
	275	37.36	822	8	0	0.096	13367	40580	0.7589586	180	9
	291	35.44	778	0	0	0.08	13379	40587	0.7584087	184	0
	306	33.63	814	0	0	0.09	13375	40587	0.7586355	167	0
20	76	42.5	5	9897	5495	0.08	4201	16421	0.9772078	222	18399
	190	43.29	106	6793	4125	0.087	10373	39258	0.9461583	208	10769
	228	43.04	240	4914	1942	0.08	12418	45033	0.9066073	208	6994
	266	42.66	377	3053	877	0.075	14331	48976	0.8543716	226	3933
	285	41.81	478	2037	433	0.09	15061	50074	0.8311865	219	2521
	304	41.13	710	1005	152	0.1	15894	50936	0.8011828	230	1187
	323	40.2	908	324	13	0.079	16524	51596	0.7806221	213	373
	342	38.84	992	15	0	0.083	16876	51923	0.769184	190	17
	361	36.74	937	0	0	0.087	16842	51949	0.7711228	231	0
	380	34.87	846	0	0	0.082	16892	51949	0.7688403	252	0

Appendix B: Cost Analysis

Number of Cities	Planes	Toal exp per hr for all planes	Dead Legs	Total hrs run (total time per plane * # of planes + time by dead leg)	Cost of aircraft	Total cost for 7 days	Passengers Lost	Passanger Load Factor	Satisfied passengers	Total Cost
3	1	445.16	0	29.11	1170000	4838.51	928	0.98864	174	5435.61
	3	1330.71	7	112.54	3510000	14515.54	921	0.98589	489	2219.36
	4	1773.59	12	155.05	4680000	19354.06	916	0.86622	641	1888.23
	4	1773.59	12	155.05	4680000	19354.06	916	0.86622	641	1888.23
	5	2216.62	13	193.73	5850000	24192.57	915	0.83155	775	1765.96
	5	2216.62	13	193.73	5850000	24192.57	915	0.83155	775	1765.96
	5	2216.62	13	193.73	5850000	24192.57	915	0.83155	775	1765.96
	5	2216.62	13	193.73	5850000	24192.57	915	0.83155	775	1765.96
	6	2662.33	28	232.57	7020000	29031.08	900	0.79722	861	1798.15
	6	2662.33	28	232.57	7020000	29031.08	900	0.79722	861	1798.15
6	6	2640.57	4	303.03	7020000	29031.08	3803	0.85911	1000	4632.2
	15	6596.56	35	797.61	1.80E+07	72577.71	3772	0.81149	2402	3791.03
	18	7917.52	42	950.33	2.10E+07	87093.25	3765	0.81029	2836	4011.39
	21	9257.27	64	1037.63	2.50E+07	101608.8	3743	0.80475	3219	4178.38
	23	10126.11	73	1191.52	2.70E+07	111285.8	3734	0.77867	3395	4686.52
	24	10567.33	83	1244.85	2.80E+07	116124.3	3724	0.74825	3424	4963.46
	26	11455.34	95	1322.19	3.00E+07	125801.4	3712	0.74729	3581	5301.29
	27	11905.83	105	1337.3	3.20E+07	130639.9	3702	0.73957	3615	5464.53
	29	12797.68	91	1378.01	3.40E+07	140316.9	3716	0.73919	3693	5819.56
	30	13248.15	107	1398.02	3.50E+07	145155.4	3700	0.73943	3709	6030.27
8	11	4847.06	1	525.68	1.30E+07	53223.65	7301	0.93953	2082	4756.11
	28	12346.7	54	1343.27	3.30E+07	135478.4	7248	0.88452	5017	4777.44

	34	14988.27	92	1669.39	4.00E+07	164509.5	7210	0.84421	5798	5587.41
	39	17194.33	119	1917.8	4.60E+07	188702.1	7183	0.82079	6412	6292.43
	42	18524.04	121	2030.5	4.90E+07	203217.6	7181	0.80851	6688	6728.05
	45	19852.55	173	2188.11	5.30E+07	217733.1	7129	0.78424	6895	7365.66
	48	21185.3	206	2313.05	5.60E+07	232248.7	7096	0.76317	7067	7970.98
	50	22079.18	249	2390.97	5.90E+07	241925.7	7053	0.76128	7156	8396.53
	53	23426.95	217	2402.05	6.20E+07	256441.2	7085	0.75735	7210	8823.04
	56	24771.5	245	2474.38	6.60E+07	270956.8	7057	0.73655	7227	9495.24
10	18	7941.85	0	816.48	2.10E+07	87093.25	12616	0.99043	3724	5152.38
	45	19859.57	48	2059.19	5.30E+07	217733.1	12568	0.92472	8685	6180.81
	54	23820.75	79	2532.37	6.30E+07	261279.8	12537	0.89604	10222	7153.32
	63	27809.18	147	2922.89	7.40E+07	304826.4	12469	0.8604	11199	8398.69
	68	30022.08	194	3159.14	8.00E+07	329019	12422	0.84184	11651	9234.83
	72	31798.21	244	3334.11	8.40E+07	348373	12372	0.82074	11858	10013.42
	77	34021.25	332	3561.37	9.00E+07	372565.6	12284	0.80738	12298	10881.35
	81	35813.11	348	3644.34	9.50E+07	391919.6	12268	0.80075	12450	11500
	86	38058.24	354	3714.96	1.00E+08	416112.2	12262	0.78747	12527	12298.48
	90	39861.69	387	3763.76	1.10E+08	435466.3	12229	0.78348	12542	12971.97
12	26	11461.14	2	1223.88	3.00E+07	125801.4	18054	0.99945	5461	5897.6
	66	29099.92	54	3120.12	7.70E+07	319341.9	18002	0.90592	12585	8670.37
	79	34826.41	123	3803.28	9.20E+07	382242.6	17933	0.88062	14738	10230
	92	40582.94	193	4363.46	1.10E+08	445143.3	17863	0.85707	16233	12036.59
	99	43686.81	264	4674.4	1.20E+08	479012.9	17792	0.83327	16742	13288.77
	106	46780.97	316	5010.21	1.20E+08	512882.5	17740	0.80836	17328	14579.59
	112	49429.97	391	5335.26	1.30E+08	541913.6	17665	0.78685	17789	15848.49
	119	52562.28	462	5529.04	1.40E+08	575783.2	17594	0.77462	17962	17191.21
	125	55262.56	440	5565.25	1.50E+08	604814.3	17616	0.7618	18009	18089.33
	132	58420.23	411	5574.53	1.50E+08	638683.9	17645	0.76188	18020	19087.08
15	42	18511.07	2	1988.75	4.90E+07	203217.6	27388	0.97713	8716	7389.29
	105	46284.41	70	4996.08	1.20E+08	508044	27320	0.89085	19727	13132.7

	126	55549.77	202	6054.77	1.50E+08	609652.8	27188	0.85755	22907	15896.4
	147	64835.93	360	7048.53	1.70E+08	711261.6	27030	0.82089	25037	19360.92
	158	69709.83	412	7504.44	1.80E+08	764485.2	26978	0.81172	26040	21154.99
	168	74135.71	536	8000.59	2.00E+08	812870.4	26854	0.78561	26758	23200.41
	179	79042.38	665	8383.05	2.10E+08	866094	26725	0.76779	27100	25468.9
	189	83520.57	759	8639.66	2.20E+08	914479.2	26631	0.75484	27283	27457.94
	200	88481.86	801	8727.48	2.30E+08	967702.8	26589	0.74756	27322	29272.39
	210	92990.71	815	8793.74	2.50E+08	1016088	26575	0.74388	27360	30896.47
18	61	26825.06	0	3133.58	7.10E+07	295149.4	40587	0.99794	13532	9232.98
	153	67230.13	113	8165.95	1.80E+08	740292.6	40474	0.94984	27724	21288.86
	184	80844.6	219	9917.12	2.20E+08	890286.6	40368	0.88282	35147	23985.08
	214	94074.21	352	11412.34	2.50E+08	1035442	40235	0.84153	38377	29050.66
	230	101150	462	12158.66	2.70E+08	1112858	40125	0.81366	39277	32362.11
	245	107799.9	659	12865.94	2.90E+08	1185436	39928	0.78919	40053	35654.28
	260	114470.7	799	13441.81	3.00E+08	1258014	39788	0.76743	40474	39030.96
	275	121219.4	822	13602.84	3.20E+08	1330591	39765	0.75896	40580	41646.71
	291	128443	778	13616.83	3.40E+08	1408008	39809	0.75841	40587	44107.81
	306	135233.1	814	13617.58	3.60E+08	1480585	39773	0.75864	40587	46389.29
20	76	33381.18	5	4073.8	8.90E+07	367727.1	51944	0.97721	16421	11467.02
	190	83407.05	106	10448.43	2.20E+08	919317.7	51843	0.94616	39258	23542.58
	228	100105.9	240	12556.53	2.70E+08	1103181	51709	0.90661	45033	29085.23
	266	116821.1	377	14599.53	3.10E+08	1287045	51572	0.85437	48976	35903.14
	285	125239.6	478	15396.37	3.30E+08	1378976	51471	0.83119	50074	39563.14
	304	133652.1	710	16322.44	3.60E+08	1470908	51239	0.80118	50936	43863.62
	323	142097.2	908	17087	3.80E+08	1562840	51041	0.78062	51596	48077.72
	342	150598.1	992	17530.53	4.00E+08	1654772	50957	0.76918	51923	51859.03
	361	159196.5	937	17461.16	4.20E+08	1746704	51012	0.77112	51949	54524.88
	380	167792.5	846	17372.56	4.40E+08	1838635	51103	0.76884	51949	57131.54

Appendix C: Matlab Program

Program 1:
Plots a US map and locates two cities

```
city1 = [-80 36.3];             % Latitude and Longitude of city1 (Blacksburg)
city2 = [-120 36.0];        % Latitude and Longitude of city2 (West Coast)

[pos,distance] = great_circle(city1,city2);

% Load the US Map to plot
load usalo

plot(uslon,uslat,'b')
hold on
plot(gtlakelon,gtlakelat)
plot(statelon,statelat)
plot(city1(1),city1(2),'r+',city2(1),city2(2),'ro')
grid

title(['Distance = ',num2str(distance,'%15.0f'),' miles']);
```

Program 2: Great Circle
Calculates the distance between cities

```
% Function great_circle to estimate great circle distance between two points
% on the surface of the earth

function [pos,distance]=great_circle(city1,city2);
%WRLDTRV2 Calculate and plot great circle distances.
%   WRLDTRV2 is used by the demo WRLDTRV.

% East longitude is POSITIVE
% North latitude is POSITIVE
   phi1=city1(1)*pi/180;
   tht1=(city1(2)+135)*pi/180+pi/4;
   [xp1,yp1,zp1]=sph2cart(tht1,phi1,1);
```

```
phi2=city2(1)*pi/180;
tht2=(city2(2)+135)*pi/180+pi/4;
[xp2,yp2,zp2]=sph2cart(tht2,phi2,1);

out=cross([xp2 yp2 zp2],[xp1 yp1 zp1]);
[tht3,phi3,r]=cart2sph(out(1),out(2),out(3));

% Following calculation uses Napier's Spherical
% Trigonometry Cosine Rule for Sides
% Ref: VNR Encyclopedia of Mathematics
angularDistCos=sin(phi1)*sin(phi2)+cos(phi1)*cos(phi2)*cos(tht1-tht2);
angularDist=acos(angularDistCos)*180/pi;
% Earth radius in miles: 3963
earthRadius=3963;
distance=(angularDist*pi/180)*earthRadius;
 pos=[xp1 yp1 zp1];
```

Program 3:
Developing a matrix when a network of cities is known.

```
clear all;

fprintf('\nNumber of cities? ');
G = input('');

for i = 1:G
   for j = (i+1):G
      fprintf('\ndistance between city %g and city %g ? ',i,j);
      entry = input('');
      F(i,i) = 0;
      F(i,j) = entry;
      F(j,i) = entry;
   end
end

for i = 1:G

   H(i,:) = F(i,:);

   for j = 2:G
      for k = G:-1:j
         W(i,k) = k;
```

```
                    W(i,k-1) = k-1;

                    if H(i,k) >= H(i,k-1)
                        temp = H(i,k-1);

                        H(i,k-1) = H(i,k);

                        H(i,k) = temp;

                    end

                end
            end
        end

        for i = 1:G
            t = 1;
            for j = 1:G
                for k = 1:G
                    if F(i,j) == H(i,k)
                        W(i,k) = j;
                        t = t + 1;
                    end
                end
            end
        end

        W
```

Program 4:
Program that schedules the flight for a period of 7 days

```
%%%%%%%%%%%%%%%%%%%%%%%START OF PROGRAM%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%-----------------INITIAL VALUES-------------------------%
clear all;

%---------------START OF THE PROGRAM THAT WILL GIVE THE FLIGHTS FOR THE DAY-------------------%
clear all;
```

```
C = input('\nNumber of cities? ');
v = 1;
for i = 1:C
    for j = 1:C
        if i ~= j
        fprintf('\ndemand from city %g to city %g ? ',i,j);
        entry = input('');

        m = [ entry*0.006 entry*0.144 entry*0.127 entry*0.11 entry*0.108 entry*0.117
entry*0.143 entry*0.142 entry*0.093];
        k = 0;
        G = 0;
        F = 0;
        T = 0;
        T(1,1) = 0;
        for k = 2:10
            X = m(k-1);
            F = G + m(k-1);

            if F >= 1
                T(1,k) = 1;
                if F > 6
                    G = F - 6;
                else
                    G = 0;
                end
            end
            if F < 1
                T(1,k) = 0;
                G = G + m(k-1);
            end
        end

        T;
                B(v,:) = T(1,:) ;
               v = v+1;
           end
       end
   end

B;
L = B';

B = L
```

```
%-----------------------------------------END---------------------------------------------------%

%---------------------WORKING OF CODE--------------------------------%



        fprintf('\nNumber of planes? ');
        G = input('');

        for i = 1:G
                fprintf('\nLocation of Plane %g (1, 2, 3)? ',i);
                entry = input('');
                F(1,i) = entry*10;
                F(2,i) = entry*10;
                F(3,i) = 99;
                F(4,i) = 99;
                F(5,i) = 99;
                F(6,i) = 99;
                F(7,i) = 99;
                F(8,i) = 99;
                F(9,i) = 99;
        end
        A = F;

   t = 2;

     D = [5 3 4 2 1; 4 5 1 3 2; 1 5 2 4 3; 2 1 5 3 4; 1 2 3 4 5];

    tic;
    while (t < 10)
        mm = t + 1;
        airport_a = 0;
        airport_b = 0;
        airport_c = 0;

        for i = 1:C
            v = 1;
            CC = i *10;
            airport(i) = 0;
        while (v < G+1)
            if A(t,v) == CC
                airport(i) = airport(i) + 1;
            end
            v = v + 1;
```

```
end
end
airport;

v = 1;
while (v < G+1)
    A(mm,v) = A(t,v);
    v = v + 1;
end

S = C;
v = 1;
for i = 1:S

    X(i) = 0;
   J = i*(S-1);
    while (v < (J+1) )
        if (B(t,v) == 1)
            X(i) = X(i) + 1;

        end
        v = v+1;
    end

end
B(t,:);
X

for i = 1:C
    P(i) = airport(i) - X(i);
end
P;

for i = 1:C

if P(i) < 0
    %DA = D(1,:);
    for k = 1:G
    if (P(i) < 0)
    for j = (C-1):-1:1
    Q = D(i,j);
        if (P(Q) > 0) & (P(i) < 0)
            QQ = Q*10;
                if (A(t,k) == A(t-1,k)) & (A(t,k) == QQ)
```

```
                            A(t-1,k) = Q*100 + i*10;
                            A(t,k) = i*10;
                            A(mm,k) = i*10;
                            P(i) = P(i) + 1;
                            P(Q) = P(Q) - 1;
                        end
                end
            end
            end
            end
        end
        end

        XXX = P

        for i = 1:C
        switch P(i)
        case -1

fprintf('=====================================================================
==================\n');
            fprintf('Could not fly one flight from Airport %f in time unit %f\n',i,(t-1));

fprintf('=====================================================================
==================\n');

        case -2

fprintf('=====================================================================
==================\n');
            fprintf('Could not fly two flights from Airport %f in time unit %f\n',i,(t-1));

fprintf('=====================================================================
==================\n');

        otherwise
        end
        end
    B
   v = 1;
    for i = 1:C
    for j = 1:C
        A;
        if i ~= j
```

```
      X(i);
      if X(i) > 0
          ii = i*10;
          for k = 1:G
              A(t,k);
              B(t,v);
              X(i);
           if A(t,k) == ii
              if (B(t,v) == 1) & (X(i) > 0)
                  A(t,k) = i*10 + j;
                  A(mm,k) = j*10;
                  X(i) = X(i) - 1;
                  B(t,v) = 2;
              else
                  A(mm,k) = A(t,k);
              end
          end
          end

      end
      v = v+1;
      end
  end
  end


     t = t + 1;
  end
  toc;
A
B
counter = 0;
for i = 1:G
    for j = 1:9
        if A(j,i) > 100
            counter = counter + 1;
        end
    end
end
```

```
fprintf('==================================================================================\n');
  fprintf('We have %g dead leg/s\n',counter);

fprintf('==================================================================================\n');

        %-----%
```