Minimizing Airline Passenger Delay through Integrated Flight Scheduling and Aircraft Routing

by

Sepehr Sarmadi

B.S. in Civil Engineering
Sharif University of Technology, 1999

Submitted to the Department of Civil and Environmental Engineering
and the Alfred P. Sloan School of Management
in partial fulfillment of the requirements for the degrees of

Master of Science in Transportation
and
Master of Science in Operations Research
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2004

Author ....
Department of Civil and Environmental Engineering
and Operations Research Center
May 14, 2004

Certified by ....
Cynthia Barnhart
Professor of Civil and Environmental Engineering and Engineering Systems
Thesis Supervisor

Accepted by ....
John-Paul Clarke
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by ....
Heidi Nepf
Chairman, Departmental Committee on Graduate Studies

Accepted by ....
James B. Orlin
Edward Pennell Brooks Professor of Operations Research
Codirector, Operations Research Center

Minimizing Airline Passenger Delay through Integrated Flight Scheduling and Aircraft Routing

by

Sepehr Sarmadi

Submitted to the Department of Civil and Environmental Engineering
and the Operations Research Center
on May 14, 2004
in partial fulfillment of the requirements for the Degrees of
Master of Science in Transportation
and
Master of Science in Operations Research

ABSTRACT

Statistics show that airline flight delays and cancellations have increased continuously over the period from 1995 to 2000. During the same period, customer dissatisfaction and complaints have followed a similar, even more dramatic trend. In 2001, as a consequence of the September 11th terrorist attacks and the resulting airline schedule reductions, delay levels decreased, but only temporarily. With growing passenger demands and stagnant capacity passenger delays and disruptions are again on the rise.

Approaches to mitigate schedule disruptions include: 1) re-optimizing the schedule during operations after a disruption occurs. For example, an airline operations controller might decide to cancel or postpone some flight legs or to re-route some aircraft to recover the rest of the schedule; and 2) building robustness into the schedule in the planning stage. By robustness we mean the ability to absorb flight delays so these effects are minimized on passengers and crews. In many cases, trying to reduce delays in the planning stage can be less costly for the airlines, especially if the actions suggested to modify the schedule are not expensive. Pushing back a flight's departure time only ten minutes might cost the airline little but can potentially reduce the number of passenger misconnections given the stochastic nature of airline operations. Canceling a flight during operations for example, can be however very costly.

The primary goal of this research is to propose planning models to re-route aircraft and re-time flight departures, either separately or simultaneously, in order to distribute slack time in the network optimally and reduce passenger delays. Using data from a major U.S. airline we observe that with our model, we can reduce flight and passenger delay levels.

Thesis Supervisor: Cynthia Barnhart
Title: Professor of Civil and Environmental Engineering and Engineering Systems

Thesis Supervisor: John-Paul Clarke
Title: Associate Professor of Aeronautics and Astronautics Engineering

Acknowledgments

I would first like to express my greatest gratitude to my advisor, Professor Cynthia Barnhart, for her constant mentorship, guidance and encouragement during my three years at M.I.T. Thank you Cindy for giving me the chance to work with you, for your patience and understanding during hard times and most importantly for your unforgettable friendship.

I would also like to thank Professor John-Paul Clarke for reading this thesis and for his valuable suggestions and ideas. J.P. is always an endless source of research ideas.

I am deeply indebted to Stephane Bratu for his help in understanding many aspects of the airline industry and for his supervision on my research from the very first day I started working with Cindy. Without him, my MIT experience would definitely be much harder.

I would like to thank my Java class TA, Jed Northridge, and Professor Steven Lerman for their Java programming tips and ideas. Without them, I would never be able to finish this thesis.

Many thanks to my dear friends at Center for Transportation and Logistics and the International Center for Air Transportation; Tom, Emmanuel, Kendell, Richard, Diana, Biova, Ryan and Michelle; whose friendship made my stay more enjoyable.

I am also grateful to Mehdi, Ali, Adel and Mardavij for being such wonderful friends. I had a lot of fun with you. I will never forget the memories we had together these years.

I truly thank my only sister Sarvin for her continuous support and encouragement.

Most importantly, I want to thank my dearest father and mother for their love, sacrifices and blessing. Without you, I wouldn't be where I am today. Thank you for everything.

Finally, I would like to thank Jill-Alicia for her love and companionship.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Airlines typically plan their schedule assuming that everything will be operated as planned. If all flights in an airline's schedule are operated as planned, all passengers would get to their destinations on time and there would be no delay. However, plans are rarely implemented as expected because airlines operate in a stochastic environment. For various reasons, actual flight departure and arrival times are different from their scheduled values. According to the Bureau of Transportation Statistics (BTS) [16], 23.1% of all flights arrived *late* on average in the period from 1995 to 2002 (Figure 1.1). A flight is classified as late using the FAA criterion when its arrival delay is larger than 15 minutes.

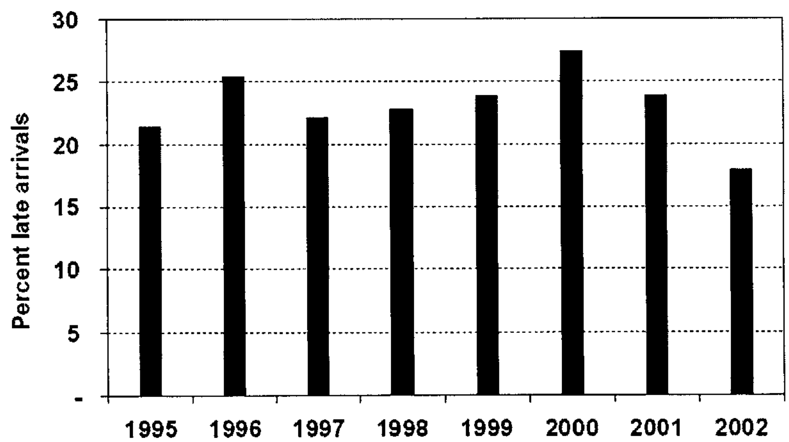


Figure 1.1: Average annual percentage of late arriving flights in the U.S.

There are several causes of flight leg delay. These include aircraft mechanical problems or unscheduled maintenance, crew unavailability, longer than expected security check times and inclement weather. Among these factors, inclement weather is the most significant in that it is reported to be the cause of 70% of the delays in the U.S. National Airspace System (NAS). U.S. airlines usually schedule their flights under the assumption that the airport will operate close to its maximum throughput capacity. Maximum throughput capacity corresponds to the conditions when visibility is excellent, cloud ceiling is high and wind direction and speed are favorable. However, when bad weather prevails, airport capacity drops. During such periods of low capacity, landing and take off queues form and delays occur.

The above factors are relatively constant from one year to another. The probability that an aircraft will have mechanical problems is essentially constant. The number of days during a year when an airport is affected by a thunderstorm will not increase dramatically in 2020 compared to 2002. But a factor that might pose some serious risks to the future growth of global air transportation is *congestion.*

Congestion

De Neufville and Odoni (2003) [27] state that during the last 25 years of the twentieth century, passenger traffic increased an average of about 6 percent each year worldwide. The growth rate is not equal in different geographical regions, of course, and depends on various factors including GDP, level of industrialization and market maturity. In the U.S., the recent long-term growth rate has been an average of 4 percent per year, which implies a doubling of passenger traffic every 15 to 20 years. The future growth rates are open to debate and small differences in assumptions will make huge differences in the total passenger traffic forecast for 20 years from now. But one thing is known for sure: passenger and cargo traffic will continue to grow and increase substantially over years to come. The majority of the world's population rarely flies and the market is far from saturation [27]. This can be very promising for air travel, but the growth of infrastructure capacity cannot keep up with the fast growth of demand. Many aviation experts believe that the inability to match available runway capacity with air travel demand growth is the major factor that threatens the future of global air transportation.

Even in recent years, the air traffic system has been operating very close to full capacity, especially in the U.S. According to De Neufville and Odoni (2003) [27], in 1999, flight delays in 29 of the busiest commercial airports in the U.S. were estimated to be more than 20,000 aircraft-hours per year, the threshold at which an airport is considered congested by the Federal Aviation Administration (FAA) standards.

Actions we can take to alleviate congestion and reduce delay levels include [25]:

- Increasing airport capacity: building second, third...airports, adding more runways (extremely expensive and in many cases almost impossible), using more sophisticated Air Traffic Control (ATC) systems, etc.
- Using demand management to charge users for the external congestion costs they generate.
- Flow management: reducing the cost and impacts of unavoidable delays, like aircraft sequencing near the terminal area.
- Embedding flight and passenger delay considerations in airline schedules and schedule recovery models.

We derive our motivation for this thesis from the last method. Being aware that airline delays are likely to increase in the near future, we propose and test models to create *robust* airline schedules, that is, schedules that achieve minimum delay during operations. In section 2.3.1, we investigate in detail the definitions and different interpretations of robustness in the context of airline scheduling. In our model, robustness is defined as the ability to absorb local delays and disruptions and prevent their widespread realization. We place emphasis on passenger delay in our models, as we believe that minimizing passenger delay (either directly or indirectly) is a more appropriate objective in building a robust airline schedule than minimizing flight leg delay (see section 3.2.4 for a comparison between flight leg delay and passenger delay).

1.2 Some Definitions

Before detailing our research, we provide the following glossary of terms used throughout the thesis.

Flight Leg

A flight leg f (sometimes also referred simply to as flight f) is a single non-stop flight between an origin airport and a destination airport.

Passenger Itinerary

A passenger itinerary is a sequence of flight legs that serve a group of passengers from their desired origin airport to their desired destination airport. An itinerary may consist of one, two or in some cases more than two flight legs.

Passenger Types

Passengers are divided into three major types, based on the composition of the flight legs on which they travel:

- Non-stop passengers or "*local passengers*": passengers whose itinerary is composed of only one flight leg. In a typical hub-and-spoke carrier in the U.S., local passengers constitute about 60% to 65% of the total number of passengers.
- One-stop passengers including:
  - "*Through passengers*": passengers whose itineraries are composed of two flight legs both assigned to the same aircraft.
  - "*Connecting passengers*": passengers whose itineraries are composed of two flight legs each assigned to a different aircraft.
- Multi-stop passengers: passengers whose itineraries are composed of more than two flight legs.

The percentage of multi-stop passengers in hub-and-spoke networks is typically well below one percent. As a result, these passengers are not considered in our analysis. Moreover, the percentage of through passengers in a traditional hub-and-spoke network is also very small; therefore, we do not distinguish between through and connecting passengers.

We consider data from a major U.S. hub-and-spoke airline, and find that the majority of passengers are local, but most of the itineraries are connecting, as shown in Tables 1.1 and 1.2. This implies that the average number of passengers traveling on connecting itineraries is less than the average number of passengers traveling on local itineraries.

Itinerary type	% of passengers
Local	66.54%
Through	1.34%
Connecting	32.12%

Table 1.1: Passenger type percentages

Itinerary type	%
Local	10.83
Through	2.49
Connecting	86.68

Table 1.2: Itinerary type percentages

Aircraft Route

Aircraft route is the sequence of flight legs operated by an aircraft. Aircraft routes are usually built so that periodic aircraft maintenance requirements are satisfied. We will discuss the aircraft maintenance routing problem in detail in section 2.1.3.

Aircraft Swap

Aircraft swapping is the act of changing the planned routes of two physical aircraft in order to curtail schedule disruptions during operations. As shown in Figure 1.2, aircraft 1 is scheduled to operate flight legs f_1 and f_3. Similarly, aircraft 2 is scheduled to operate flight legs f_2 and f_4. Also assume that during operations, while flight leg f_2 is on time and aircraft 2 is ready to fly, flight f_1 has a long arrival delay and aircraft 1 is not available. Maintaining the original (planned) aircraft routes will delay the departure time of flight f_3 until aircraft 1 becomes available. To avoid this delay, it might be possible to operate flight f_3 with aircraft 2 and flight f_4 with aircraft 1. That is, it might be possible to swap the original aircraft routes to f_1-f_4 and f_2-f_3 and, in so doing, reduce delay.

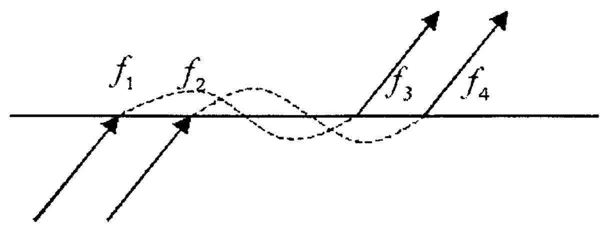


Figure 1.2: Illustration of aircraft swapping

It's important to note that aircraft swaps are feasible only when they do not cause crew work rule or aircraft maintenance violations.

Flight Arrival Delay

Flight arrival delay is defined as the positive difference between a flight's actual and scheduled arrival times.

Passenger Delay

Passenger delay is the positive difference between the actual and scheduled passenger's arrival time at his/her destination.

Disrupted Passengers

In general, a disrupted passenger is a passenger whose planned itinerary of flight(s) differs from her/his actual (set of) flight(s). Based on this definition, a local passenger is disrupted only if her/his flight is canceled, a through passenger is disrupted if one of her/his flight legs is canceled, and finally, a connecting passenger is disrupted if one of her/his flight legs is canceled and/or if s/he misses her/his connection. In Table 1.3 we show that about two thirds of all of disrupted passengers are connecting.

Passenger Type	% of Total Disrupted Passengers
Local	36.72%
Through	1.34%
Connecting	61.94%

Table 1.3: Categorizing disrupted passengers based on their itinerary type

1.3 Contributions and Thesis Outline

1.3.1 Contributions

The contributions of this thesis are summarized as follows:

1. We provide a review of literature on the general topic of airline scheduling, with particular emphasis on robust airline scheduling.

2. We develop a combined flight departure re-timing and aircraft re-routing model to minimize delay propagation. We calibrate and test our model using data from a major U.S. airline. The results show that our model can reduce flight and passenger delay levels and reduce the number of disrupted passengers. We quantify the propagated delay achieved by combining flight departure re-timing and aircraft re-routing models, as compared to those achieved by aircraft re-routing models only.

3. We present two other robust scheduling models. The first model is a combined flight departure re-timing and aircraft re-routing model to minimize the expected number of misconnected passengers. The second model is a flight retiming model to create recovery options for misconnected passengers.

1.3.2 Thesis Outline

In Chapter 2, we provide a review of literature on airline schedule planning and robust scheduling. In Chapter 3, we analyze the components of flight delay, paying particular attention to propagated delay. We quantify propagated delay and measure its impact on passenger delays. We present a flight departure re-timing and aircraft re-routing model, with the objective to minimize total expected propagated flight delay. We present the implementation details and proof-of-concept results using data from a major U.S. airline.

In Chapter 4, we describe the role of disrupted passengers in affecting total passenger delay, and present two robust scheduling models. The first model is a flight departure re-timing and aircraft re-routing model to minimize the total expected number of misconnecting passengers. The second model is a flight departure re-timing model to minimize the number of connecting itineraries without a recovery option, again focusing on misconnecting passengers. We support these modeling ideas with examples.

Chapter 5 provides concluding remarks and directions for future research.

Chapter 2

Airline Schedule Planning

2.1 The Planning Process

The airline schedule planning problem is the problem of generating a schedule that maximizes an airline's profitability and satisfies a set of rules and regulations regarding fleet assignment, aircraft routing and crew scheduling. During this process, the planner should find exact answers to the following questions:

- What markets the airline wants to serve?
- What should be the frequency of service for each of the selected markets?
- What should be the departure time of each flight leg?
- What type of aircraft should be assigned to each flight leg?
- How should a single aircraft be routed in the network to guarantee it receives sufficient maintenance?
- How should crewmembers and flight attendants be assigned to individual flights to guarantee the satisfaction of numerous rules and regulations that labor unions mandate?
- And most importantly, how should all these decisions be made so that the overall profitability of the airline is maximized?

In other words, the airline schedule planning process has two dimensions: maximizing profitability while maintaining feasibility.

For a small network, all these decisions can be made manually or based on a planner's judgment and experience. However, the size of a major airline network is very large. A major American or European network carrier operates more than a thousand flights per day with hundreds of aircraft, with several different aircraft types, serving tens of thousands of markets using hundreds of airports. Today, almost all airline managers are convinced that considering the quantity and complexity of the decisions involved in the airline schedule planning process, applying optimization tools and techniques is inevitable.

Airline Schedule Planning, a Sequential Process

Traditionally, the airline schedule planning process is composed of four distinct steps that are dealt with independently and in a sequential manner. The output from each step provides the input for the next step. As we will see shortly, these four steps are not completely independent and combining some of these steps fully or partially can produce improved schedules. There's no doubt that formulating the entire process as one large model is possible and in theory should generate more profit. With current computer hardware technology and optimization algorithms, however, solving such a model is intractable. This is the main reason for decomposing the airline scheduling process into pieces (Barnhart and Talluri (1997) [12]).

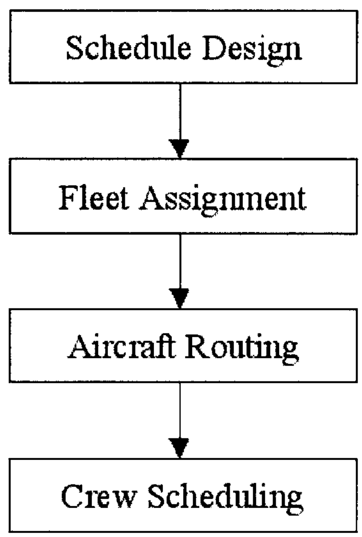


Figure 2.1: A typical sequential airline schedule planning process

2.1.1 Schedule Design

The output of the schedule design step is a complete list of the flight legs the airline wants to offer, specifying each flight's origin, destination and departure time (which roughly determines arrival time as well). To build a schedule, the airline has to decide:

- What origin-destination (OD) markets to serve?
- What type of service to offer for each OD market (non-stop vs. connecting)?
- For connecting service, where should the connection location be located?
- What should be the frequency of service offered in each market?

- And finally, what should be the exact departure time of each flight leg?

Due to the complexity and strategic nature of the decisions made in the schedule design process, operations research has had little impact on this step of airline scheduling. Airlines usually perform this step manually. The way each airline chooses to construct its schedule depends on numerous factors, including: the overall network type served by that airline (hub-and-spoke vs. point-to-point); its long term ambitions (to grow as a big carrier or to remain as a local service provider); its fleet size and composition (not all city pairs can be served by a Boeing 737, for example); its competitors in each OD market and the remaining unrestricted demand that can still be absorbed; and its crew and maintenance base locations. As seen, incorporating all these considerations in a mathematical model is extremely difficult if not impossible. Barnhart and Cohn (2003) [6] mention that the difficulty of modeling the schedule design problem as a mathematical model also arises from the fact that the data needed to formulate such a problem, in particular unconstrained demands and average fares, are very difficult to estimate accurately.

Nevertheless, especially during recent years, considerable amounts of research have been performed on using optimization models to improve a given schedule by applying only a limited number of changes. This class of problems is known as incremental schedule design.

Berge (1994) [14] studies the problem of generating the best sub-timetable within a larger timetable. He finds a feasible subset of flight legs that maximizes market coverage. He defines market coverage as the probability that a random passenger finds at least one path in his/her decision window. The inputs to his model are a list of candidate flight legs (departure and arrival times), a fleet constraint (total number of airplanes), a list of markets (city pairs) and relative importance weightings and an exhaustive list of feasible passenger itineraries. He does not consider supply-demand interactions in his model (the fact that a higher frequency of service in a market stimulates more demand). He proposes a heuristic method and also an integer linear programming formulation to find the optimal sub-network within a network containing 24 aircraft. He reports that his heuristic can solve the model fast under some additional assumptions, but his integer formulation suffers from long runtimes.

Marsten, Subramanian and Gibbons (1996) [41] propose an incremental schedule design approach. They use the OAG (Official Airline Guide) to generate a pool of possible paths (passenger itineraries). They then give a score to each path using a scoring function based on path characteristics (elapsed time, number of connections, etc). In this way, they eliminate a large number of unfavorable paths. Using the remaining sets of paths, they compute the market share for each path using unconstrained passenger demands. After computing the number of passengers on each flight leg, they solve the fleet assignment problem for different combinations of flight leg additions and deletions (with respect to the initial schedule), considering the maximum allowable number of fleet types. Finally, they compare the net profits (the sum of average passenger fares over the selected markets minus the fleet assignment cost) of different combinations of addition and deletion sets and then select the profit-maximizing schedule. They do not consider supply-demand interactions either.

Rexing, et al. (2000) [42] integrate flight schedule design and fleet assignment decisions in their Fleet Assignment Model with Time Windows (FAMTW). FAMTW is an extended fleet assignment model, which includes incremental schedule changes. They generate copies for each flight leg in the initial schedule and incorporate the new set of variables into the basic FAM, assuming that the changes in flight departure times are small enough that there are no changes in demand. By generating the flight copies, the set of feasible aircraft connections increases substantially, leading to better fleet assignment solutions and considerable cost savings. They also show that by using FAMTW, they can free up aircraft by tightening the schedule and increasing aircraft utilization.

Lohatepanont and Barnhart (2004) [40] present an Integrated Schedule Design and Fleet Assignment Model (ISD-FAM). ISD-FAM, in its entirety, is a fleet assignment model with some optional flight legs. The optional flight legs can be candidates from the initial schedule that may be deleted or some new flight legs that may be added to the initial schedule. Optional flight legs can also be flight copies for the initial set of flights. Having said that, FAMTW is a special case of ISD-FAM with no flight addition or deletion. They also introduce new parameters called "recapture rates" to modify each itinerary's demand as a result of flight leg addition or deletion. In the simplified version of the ISD-FAM, they do not consider the change in market share as a result of changing the service frequency in each market. In the Extended ISD-FAM, also known as ESD-FAM, they utilize "demand correction terms" to adjust demands as necessary. Both ISD-FAM and ESD-FAM are large-scale, requiring long runtimes and large amounts of memory to solve. Their validity depends on the correct estimation of OD market demands, recapture rates and demand correction terms. But it is shown that considerable economic benefits are achievable if these models are used correctly.

Lan, Clarke and Barnhart (2003) [39] propose a flight re-timing model to minimize the expected number of misconnected passengers. Although their model does not increase the planned profitability of the schedule, it can reduce realized disruption costs. Their approach can also be classified as an incremental schedule design model. We will investigate their model in more detail in section 4.2.1.

2.1.2 Fleet Assignment

After the airline generates its flight schedule, it has to decide what type of aircraft (fleet type) to assign to each flight leg, given the set of available aircraft types and numbers. The cost of assigning an aircraft of type k to a flight leg i is the sum of the operational costs of performing flight leg i with an aircraft of type k (such as fuel cost, gate rental, landing and take off costs, etc) and the *spill cost*. The spill cost is the revenue lost due to lack of seat availability. There's clear trade-off between using a big vs. a small aircraft. Bigger aircraft have high operational costs and low spill costs. Smaller aircraft have low operational costs but they can incur high spill costs if assigned to flight legs with high demand. There are also other considerations; for example, small aircraft cannot be assigned to certain long haul flight legs.

The fleet assignment problem also has to guarantee that each flight leg is covered by exactly one aircraft type; aircraft flow balance is satisfied; and the total number of aircraft of a given type assigned to different flight legs does not exceed the total number of available aircraft of that type in the airline's fleet. Additional constraints can also be added to enhance the model with maintenance requirements, gate and noise restrictions and crew considerations. The interested reader is referred to Barnhart and Talluri (1997) [12] for a more complete discussion about these additional constraints.

Integer programming methods have been used extensively to solve the airline fleet assignment problem and the research performed in this area is relatively vast. Historically, the fleet assignment problem has been cast as a multi-commodity flow problem with side constraints. The underlying network to solve the fleet assignment is a directed time-line network (see section 3.3.2 for a definition of a time-line network). Each aircraft type can be regarded as a commodity type and we need to flow each commodity across the network in a way that minimizes total cost, balances aircraft types and assigns each flight leg to exactly one aircraft type.

Daskin and Panayotopoulos (1989) [26] propose an integer linear programming formulation for solving the fleet assignment problem in a hub-and-spoke network with a single hub. They present a Lagrangian Relaxation approach for solving the problem and observe that the solution to the Lagrangian Relaxation problem provides a useful upper bound for the initial primal maximization problem. They also propose a heuristic method to first convert the fractional solution to Lagrangian Relaxation problem into a feasible integer solution for the primal problem, and then to improve that solution.

Abara (1989) [1] formulates and solves the fleet assignment problem using aircraft connection arcs (or aircraft-turns) as decision variables. Two disadvantages of his formulation are the large number of aircraft-turn variables and also the inability of the model to allow different turn times for different fleet types at various locations. He reports that substantial savings are obtained when this model is applied to American Airline's network.

Hane et al. (1995) [32] formulate the basic Fleet Assignment Model (basic FAM) as a multi-commodity network flow problem with side constraints, based on an underlying directed time-line network. Using the special topology of the fleet assignment network, they present several techniques to reduce the size of the problem (such as node consolidation, algebraic preprocessing, islands, etc). They compare different solution approaches (like dual steepest edge simplex and specialized branching methods) and report improved solution times compared to the conventional LP-based branch-and-bound. The authors solve fleet assignment problems for large U.S. carriers (with 10-14 fleet types and 2000-2500 flight legs) within 10-20 minutes of computation time on 1995 workstation class computers.

Gu et al. (1994) [31] study the complexity of the basic fleet assignment problem presented by Hane et al. They study the specific conditions under which the problem is an easy network flow problem, and when it is NP-complete, NP-hard, etc. They also investigate the behavior of the solution as a function of the number of fleet types and present analytical expressions for the minimum number of planes required of each fleet type to cover a schedule.

Clarke et al. (1996) [19] provide modeling techniques to enhance the basic fleet assignment model with crew and maintenance considerations, while preserving its solvability. They argue that without any maintenance or crew considerations, the solution of the fleet assignment model might not be feasible for the maintenance routing and crew scheduling problems.

Although basic FAM is a strong formulation and can be solved relatively fast, it has some major shortcomings. It is very hard to consider the network effects or leg interdependencies when calculating spill costs. FAM also does not accurately model recapture of spilled passengers on other itineraries. In light of these facts, Barnhart, Kniker and Lohatepanont (2001) [9] present an Itinerary based Fleet Assignment Model (IFAM), considering network effects and more accurately calculating the profitability of fleet assignments. They introduce new sets of variables and parameters and embed the spill optimization problem within FAM. Although their model results in improved solutions compared to those of basic FAM, its LP relaxation is often fractional, resulting in increased solution times. To alleviate some of these tractability issues, Barnhart, Farahat and Lohatepanont (2002) [7] present an equivalent, yet revised, IFAM, one based on composite variables that can consider network effects in calculating spill costs. Their formulation and solution approach yields tighter LP bounds, better solution times and more opportunities for model integration.

2.1.3 Aircraft Maintenance Routing

After assigning aircraft types to flight legs, the airline has to assign the physical aircraft (or tail numbers) to the flight legs. In other words, the airline must determine the sequence of flight legs (route) each tail number has to fly in its daily operation. The factor that plays a key role in determining these routes is the aircraft maintenance requirement. Hence, the problem of routing the aircraft tail numbers in the network is also called the aircraft maintenance routing problem. Because aircraft maintenance routing is a focus of this thesis, here we study this problem in more detail.

Airlines are required by law to conduct periodic maintenance checks on each individual aircraft in their fleet after a certain number of flying hours. These requirements are very strict and the aircraft will be grounded if the airline violates them [12].

In the U.S., the first major check required by the FAA should occur at least every 60 hours of flying. This check is called an *A-check.* Airlines usually perform the A-check after 40-45 hours of flying, which translates into every 3 to 4 calendar days. An A-check involves a complete visual inspection of all the major systems of the aircraft and requires 10-20 man-hours. B-checks are required every 300 to 600 hours of flying and involve a more thorough inspection. C and D-checks are the most exhaustive series of checks. They happen every one to two years and require that the aircraft be taken out of service for about one month to go through a complete set of visual and mechanical inspections. To ensure that A-check requirements are met, airlines have maintenance facilities at a few airport locations across their network (usually in hubs and also cities with a high number of daily flight departures, to maximize maintenance opportunities). Because the number of maintenance bases is limited, airlines try to route individual tail numbers in a way that each aircraft spends a night at one of these maintenance bases after at most 3 to 4 days of flying. Except for some special circumstances, these checks are scheduled during the night so that aircraft utilization is not reduced. The airline aircraft maintenance routing problem is then to find a set of routes that satisfy these A-check requirements, cover all the flights and guarantee flow feasibility.

According to Clarke and Smith (2000) [21], airlines have developed efficient solution procedures (either heuristic or optimization-based) to build feasible maintenance routes.

One of the most powerful formulations to solve the aircraft routing problem is the *string-based* aircraft maintenance routing model ([24]) stated as a set partitioning linear integer program. A string is a sequence of connected flights that starts and ends at a maintenance station, satisfies flow balance and captures maintenance requirements such as maximum flying time between stations. The model can consider complicated maintenance requirements that are hard to capture using leg-based formulations. The basic string-based aircraft routing model is stated as follows:

Decision variables:

x_s	equals 1 if string s is in the solution; 0 otherwise
$y_{(e_1,e_2)}$	equals to the number of aircraft on the ground at a maintenance station between adjacent events e_1 and e_2

Sets:

S	set of maintenance strings
F	set of flights
S_i^-	set of strings ending with flight i
S_i^+	set of strings starting with flight i
G	set of ground arcs

Parameters:

c_s	maintenance cost of string s
a_{is}	equals 1 if flight i is in string s; 0 otherwise
N	number of available aircraft
r_s	number of times string s crosses the count time
p_j	number of times ground arc j crosses the count time

The model:

$$\min \sum_{s \in S} c_s x_s \tag{2.1}$$

subject to

$$\sum_{s \in S} a_{is} x_s = 1 \qquad \forall i \in F \tag{2.2}$$

$$\sum_{s \in S_i^+} x_s - y_{(e_{i,d}^-, e_{i,d})} + y_{(e_{i,d}, e_{i,d}^+)} = 0 \qquad \forall i \in F \tag{2.3}$$

$$\sum_{s \in S_i^-} x_s + y_{(e_{i,a}^-, e_{i,a})} - y_{(e_{i,a}, e_{i,a}^+)} = 0 \qquad \forall i \in F \tag{2.4}$$

$$\sum_{s \in S} r_s x_s + \sum_{j \in G} p_j y_j \le N \tag{2.5}$$

$$y_j \ge 0 \qquad \forall j \in G \tag{2.6}$$

$$x_s \in \{0,1\} \qquad \forall s \in S \tag{2.7}$$

The objective function, as stated in Equation 2.1 is to minimize the total maintenance costs of selected strings. Constraints 2.2 guarantee that each flight is assigned to exactly one string. Constraints 2.3 and 2.4 are flow balance constraints ensuring that the number of aircraft arriving at each maintenance station at any time is equal to number of aircraft departing that station. Constraint 2.5 is the aircraft count constraint ensuring that the number of aircraft used does not exceed the number available. Constraints 2.7 require the string variable values to be binary. Because ground arc variables can be stated as a summation of strings variables, their integrality is relaxed in constraints 2.6.

Some airlines might enforce other requirements on the aircraft routes. For example, airlines like to route aircraft tail numbers in a way that they experience equal wear and tear across the network. One way to satisfy this requirement is to force each aircraft to fly all flights assigned to its fleet. For a more detailed discussion about this additional requirement, interested readers are referred to Barnhart et al. (1998) [5].

The cost of operating string s can be viewed as the aircraft maintenance cost at the beginning and at the end of the string. But this cost is more or less similar for different combinations of aircraft routes. Some researchers use an objective function for this problem that maximizes *through revenue*. Through revenue is the incremental revenue passengers are willing to pay for connections that do not require them to transfer between aircraft. In reality though, many of these planned through itineraries are broken in the operation phase as a result of aircraft swapping. Given this, the aircraft maintenance routing problem is most suitably cast as a feasibility problem. If no feasible solution can be found for the aircraft routing problem, we must implement changes in the fleet assignment or in the schedule.

The major disadvantage of the string-based aircraft routing formulation is the large number of integer string variables. The solution of the LP relaxation tends to be fractional providing poor bounds on the optimal solution. The large number of variables requires that we use column generation at every node of the branch-and-bound tree. This leads to long solution times and large memory requirements. Considering these disadvantages, some researches have developed heuristics for solving the aircraft routing problem.

Feo and Bard (1989) [29] propose models to first find the optimal number and location of the maintenance stations and then to develop flight schedules that meet the periodic four-day maintenance routing requirement. They cast the problem as a min-cost multi-commodity network flow problem with integer variables and solve it with a two-phase heuristic method.

Kabbani and Patty (1992) [34] propose a set partitioning formulation for the three-day aircraft maintenance routing problem at American Airlines. In their formulation, each column represents a weeklong route (not necessarily satisfying maintenance requirements) and each row represents a flight. They develop pseudo-costs to penalize routings with unfavorable characteristics, like those that violate aircraft connection times, maintenance requirements, etc.

Clarke et al. (1996) [20] construct a flight-based formulation to solve the aircraft maintenance routing problem. They select a group of sub-tours that cover all the flights, satisfy maintenance requirements and maximize through revenue. They use Lagrangian relaxation and subgradient optimization techniques to solve their model based on data from a major airline.

Gopalan and Talluri [30] present graph-theoretic approaches to solve the aircraft maintenance routing problem to satisfy the three-day A-check requirement. They model the problem as that of finding special Euler tours in a directed graph G = (V, E), where V represents the set of stations where the aircraft overnights and E represents the set of lines of flying (LOF's). A LOF is a sequence of consecutive daily flights starting and ending at stations where the aircraft overnights.

Barnhart et al. (1997) [5] present a string-based model to solve the combined fleet assignment and aircraft maintenance routing problem. Because the number of string variables is large, they use a branch-and-price approach to solve the integer programming formulation. Branch-and-price is branch-and-bound with linear programming relaxations solved using column generation at nodes of the branch-and-bound tree. Their approach eliminates the possibility of finding a fleet assignment solution that is maintenance infeasible. They show how they can extend their model to include additional constraints to achieve equal wear and tear on aircraft.

Finally, we note that realized routings are usually different from planned routings because airline operations controllers use aircraft swapping as a mechanism to recover from schedule disruption (see section 1.2 for a definition of aircraft swapping).

2.1.4 Crew Scheduling

The last step in the airline schedule planning process is to assign crewmembers to individual flight legs. Pilots are usually certified to operate only certain types of aircraft and therefore, the crew-scheduling problem is solved after the solution to the fleet assignment problem is produced. Traditionally, the crew-scheduling problem is divided into two sequential steps.

Step 1: The *Crew Pairing* Problem

To understand the definition of a *pairing,* we need to introduce another concept. A *duty period* is a daylong set of consecutive flights that can be assigned to a single crew. Duty periods, or duties are always followed by a rest period. Any duty, in order to be crew feasible, must satisfy a host of rules. For example, there is an upper limit on the total number of hours a crewmember can fly during a day and there is a lower limit on his/her connection time between consecutive flight legs. A pairing is a sequence of duty periods separated by rest periods that start and end at the same crew base (domicile) location. The duration of a pairing is usually three to four days, but it can span from one day to multiple weeks. The crew-pairing problem can be cast as a set-partitioning problem. The objective is to choose a subset of pairings that cover each flight leg exactly once with minimum total cost. Although the number of possible pairings is very large (in the order of billions of pairings for a problem with several hundred flight legs), using the set-partitioning formulation allows us to capture non-linearities in the cost structure of pairings. Typically, the cost of each pairing is a non-linear function of the cost of flying, of elapsed time away from base and of minimum guaranteed pay. Another advantage of the set-partitioning formulation is the ability to satisfy the complex work rules when generating pairings, and hence, to eliminate the need to model them in the problem formulation.

Step 2: The *Crew Assignment* Problem

The objective of the crew assignment problem is to assign individual crewmembers to cost minimizing *schedules.* A schedule is a month long sequence of consecutive pairings, separated by rest periods, vacations, training, etc., that begins and ends at the same crew base.

The focus of crew scheduling research has been on the crew pairing problem, partly because of the tractability issues associated with the crew assignment problem and partly because major cost savings (crew costs constitute the airline's second largest operating costs after fuel costs) are associated with optimizing the crew pairing problem. The objective of the crew assignment problem is typically centered around satisfying crew preferences and achieving equitable assignments ([24]).

Some recent work in the area of crew pairing optimization includes Anbil et al. (1992) [4], Hoffman and Padberge (1993) [33], Barnhart et al. (1994) [8], Beasley and Cao (1996) [13], Chu et al. (1997) [18], Desaulniers et al. (1997) [28], Vance et al. (1997) [45], Barnhart et al. (1998) [11], Klabjan and Schwan (1999) [38] and Klabjan et al. (1999) [36].

2.2 Integrated Models

In recent years, considerable amounts of research have been performed on integrating some of the sub-problems in the sequential airline schedule planning process. Researchers have realized that merging steps produces opportunities for improving the schedule. The existing integrated models in the literature are formed based on one of the following strategies:

- Merging two sub-problems completely to create an integrated larger problem that fully captures the objectives and constraints of both problems (full integration); and
- Enhancing a sub-problem model by incorporating some of the key elements of another sub-problem (partial integration).

In Table 2.1, we summarize some of the integrated models developed to date.

	Integration Level		Area of Integration (*)			
	Full	Partial	1	2	3	4
Clarke et al. (1996) [19]		■		■		■
Barnhart et al. (1998) [5]	■			■	■	
Barnhart, Lu and Shenoi (1998) [10]		■		■		■
Cordeau et al. (2000) [23]	■				■	■
Rexing et al. (2000) [42]		■	■	■		
Cohn and Barnhart (2002) [22]		■			■	■
Klabjan et al. (2002) [37]		■	■		■	■

(*) 1: Schedule Design, 2: Fleet Assignment, 3: Aircraft Routing, 4: Crew Scheduling

Table 2.1: Some Integrated Airline Scheduling Models

2.3 Robust Airline Scheduling

2.3.1 Robustness Definition

The concept of *robust airline scheduling* is based on a very simple fact: schedules are affected by irregularities in operations, making it impossible to operate them as planned. The goal of robust airline scheduling then is to identify irregularities that occur in the operations phase (like bad weather, aircraft mechanical problems, passenger and crew disruptions, etc) and model them in the planning phase. As Ageeva and Clarke state [2], using 'optimal' schedules that are not robust to disruptions is like driving a Porsche in a street with un-synchronized traffic lights.

Two methods for achieving schedule robustness are:

- Re-optimizing the schedule after a disruption occurs during operations. This is also known as schedule recovery; and
- Incorporating robustness into the schedule in the planning phase, before disruptions happen.

Literature on airline recovery models is relatively rich, with a growing amount of research performed in this area. Building robustness into the schedule in the planning stage, however, is a subject that has little share among published research.

Lan, Clarke and Barnhart's (2003) [39] definition of a robust airline schedule is one that possesses at least one of the following characteristics:

- It minimizes some *realized* cost. For example, it can minimize the expected cost or the cost under the worse case scenario, among all possible realizations of the planned schedule.
- It minimizes aircraft or passenger delays and disruptions.
- It is easy to recover aircraft, crews and/or passengers once disrupted.
- It isolates disruptions in the sense that it prevents system-wide propagation of disruptions.

2.3.2 Literature Review on Robust Airline Scheduling

Ageeva (2000) [2] proposes a method to incorporate operational considerations into the aircraft routing problem in order to achieve robust schedules. In her model, Ageeva maximizes the number of aircraft swap opportunities by identifying and selecting aircraft routes that meet more than once (two aircraft routes meet if they have a common station at the same time). Once two planes are swapped at the first meeting point, they can be swapped back to their original routes at the second point. By making swap opportunities, Ageeva provides flexibility for planners during schedule recovery.

Rosenberger, Johnson and Nemhauser (2001) [43] develop a robust fleet assignment and aircraft routing model that embeds many short cancellation cycles and isolates hubs. Airline controllers have to cancel a cycle of flights to maintain balance. Hence, it is very favorable for them if the cycle contains the minimum possible number of flights. This is what Rosenberger et al. call a short cycle. They also show that decreasing hub connectivity can increase the number of short cycles in the schedule.

Schaefer et al. (2001) [44] present a heuristic approach for finding robust crew schedules. They first find an approximate expected cost for each crew pairing, making some assumptions on the underlying crew recovery procedure and using a simulation tool called SimAir. Using these approximate costs, they solve the deterministic crew pairing problem. Because operational crew costs can be as much as eight to ten times as the planned costs, they argue it is essential that crew solutions identified in the planning stage perform well in practice.

Yen and Birge (2001) [46] develop a two stage stochastic integer-programming model to solve the crew pairing problem. They devise a methodology to integrate disruptions into the evaluation of crew pairings. They also present a branching algorithm using the structure of the problem that branches on several variables at the same time.

Chebalov and Klabjan (2002) [17] present a planning model and a solution methodology to produce more robust crew pairings. Besides the objective of minimizing usual crew costs, they introduce the objective of maximizing the number of move-up crews, that is, the number of crews that can potentially be swapped in operations. They use a Lagrangian decomposition technique to solve their model, but they do not present a method to evaluate the robustness of the new crew pairing solution.

Kang and Clarke (2003) [35] develop a methodology to create airline schedules that are easier to recover when disruptions occur. They partition the airline's flights into distinct groups or layers. Each layer is isolated from the other layers in the sense that delay produced and propagated in each layer will not impact other layers. They claim that airlines can use this idea to prioritize their flights based on the layer to which they belong and simplify recovery operations. When resources become limited due to weather problems for example, the airline will recover layers in sequence, first recovering the layer with highest priority.

Lan, Clarke and Barnhart (2003) [39] develop a robust aircraft routing model to minimize the expected delay propagated along aircraft routes. Column generation and a specialized branch-and-bound technique are used to solve their mixed integer programming formulation. They also propose a flight departure retiming model with the objective of minimizing the expected number of passenger misconnections. They use data from a U.S. major airline to calibrate and evaluate their models.

Chapter 3

Robust Aircraft Routing with Time Windows to Minimize Total Expected Propagated Delay

3.1 Flight Delay and its Components

Flight delay can be broken into two components:

- Propagated delay (PD): Propagated delay is the component of flight delay caused by waiting for incoming aircraft. In other words, propagated delay is a function of aircraft routing.

- Independent delay (ID): Independent delay is caused by any reason except previous operations and is not a function of routing. Several factors can cause independent delays including bad weather, aircraft mechanical problems and Ground Delay Programs (GDP). For example, when airlines schedule more arrivals than an airport's capacity, FAA's Air Traffic Control System Command Center (ATCSCC) will issue Controlled Times of Arrival (CTA's) to each flight, causing some flights to have arrival delays. This kind of flight delay is independent of the history of previous operations and is referred to as independent delay. In section 3.2.3, we will discuss a method to calculate the value of independent delay for each flight.

  In the next section, we will study propagated delay in detail. This component of flight delay is a major element of our first model.

3.2 Propagated Delay

3.2.1 The Resources Involved in Flight Delay Propagation

An airline schedule is an interwoven schedule of aircraft, crews and passengers. The rules governing an aircraft's routing schedule are different from the rules governing a crewmember's monthly schedule. An aircraft tail number must overnight in a maintenance station after a certain number of flying hours, and airline schedulers tend to route aircraft to guarantee equal wear and tear across the fleet. The time it takes for a specific tail number to visit a city again can exceed a month in some cases. On the other hand, crew and flight attendants return to their home city (crew base) after a limited amount of time, and strict work rules and regulations for pilots and flight attendants must be satisfied by every crew schedule. For example, every crewmember needs a minimum amount of rest period after a certain amount of flying time.

For the reasons mentioned above, airlines don't typically keep aircraft and crew together at all times, and hence a late arriving flight can affect many other flights that use the aircraft and crew or transport connecting passengers of the late arriving flight. In other words, aircraft, crew and passengers are three major elements that can cause and/or escalate flight delay propagation.

3.2.2 The Domino Effect

It is typical that flight delays exhibit a "domino effect"; once they begin (particularly if they occur early in the day), they propagate to other parts of the network.

Figure 3.1 shows the domino effect of flight delay propagation. Arrival delays for flights in our airline's data set are averaged for each one-hour time window on August 31^{st}, 2000. Airline operations controllers, well familiar with this propagation effect, try to curtail its effects as early as possible to limit the number of disrupted passengers and crews, and control the number of flight cancellations. There are not many recovery choices for passengers disrupted late in the day, causing them to overnight at the disruption location. This translates into additional passenger recovery costs and loss of passenger goodwill.

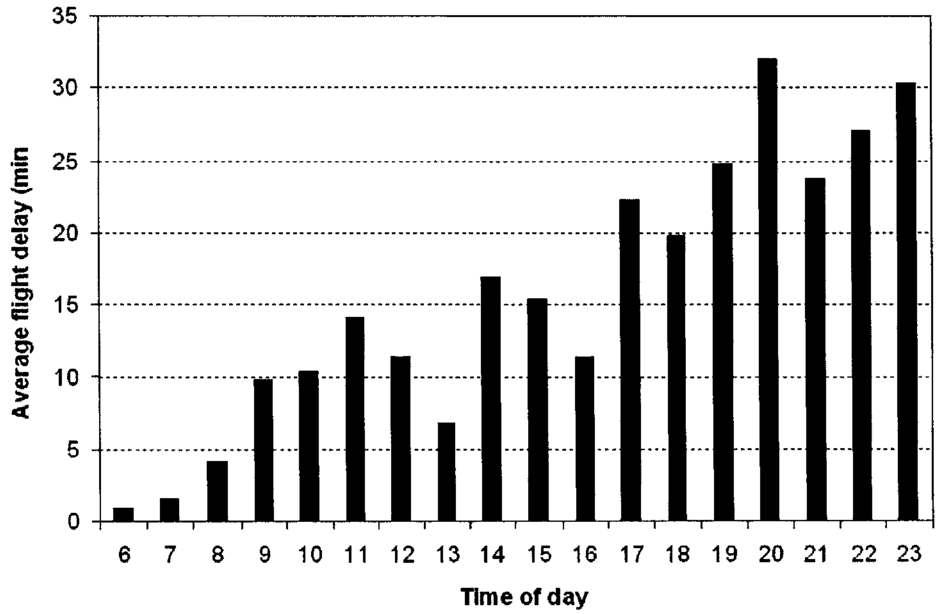


Figure 3.1: Average flight delay increases by time of day

3.2.3 Quantifying the Propagated Delay

Any non-stop flight leg f is identified by its flight number, origin and destination airport, planned departure time (PDT) and planned arrival time (PAT). Actual flight departure and arrival times are almost always different from their scheduled times. Hence, ADT and AAT denote the actual departure and arrival times, respectively. Total arrival delay (TAD) is then calculated as follows:

$$TAD = \max((AAT - PAT), 0) \tag{3.1}$$

According to our definition, negative delay is regarded as zero. Also note that TAD is composed of two parts: propagated delay (PD) and independent delay (ID).

$$TAD = PD + ID \tag{3.2}$$

We define the planned turn time (PTT) as the difference between the incoming flight's PAT and the outgoing flight's PDT.

$$PTT_{ij} = PDT_j - PAT_i \tag{3.3}$$

Minimum turn time (MTT) is the minimum time required to turn the aircraft. Slack time is the difference between PTT and MTT.

Assume that in Figure 3.2, flights i and j are performed with the same aircraft. There's a delay transferred (propagated) from flight i to flight j only if the arrival delay of flight i is greater than the slack time between i and j. Hence:

$$\begin{aligned} &Slack_{ij} = PTT_{ij} - MTT \\ &TAD_i \leq Slack_{ij} \Rightarrow PD_{ij} = 0 \\ &TAD_i > Slack_{ij} \Rightarrow PD_{ij} = (TAD_i - Slack_{ij}) \end{aligned} \tag{3.4}$$

and,

$$PD_{ij} = \max(TAD_i - Slack_{ij}, 0) \tag{3.5}$$

Therefore, if we are given the arrival delay of the incoming flight (i) and the slack time between flights i and j, we can calculate the propagated delay from flight i to j. The arrival delay of flight i in turn is a combination of the propagated delay transferred to flight i from its predecessor flight in the route of the aircraft assigned to i and the independent delay of flight i (ID).

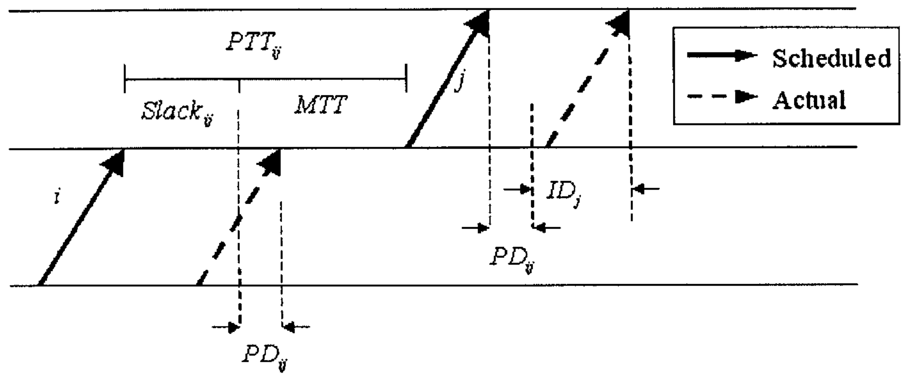


Figure 3.2: Illustrating the propagated delay

From Equation 3.5, increasing slack time between flights can absorb delay. Airlines, however, are not willing to put large amounts of slack in their schedules because it results in decreased aircraft utilization. A schedule with tightly scheduled flights, although achieving high aircraft utilization, is, however, vulnerable to delay propagation and schedule disruptions. Many optimization models are formulated to maximize the planned throughput or productivity of a system, but in doing so, they make the system less robust to disruptions. In operations, the system has to pay the extra cost incurred as a result of being less robust, and this extra cost might outweigh the anticipated savings of an optimized plan.

The above argument is summarized in Figures 3.3-a and 3.3-b. In any system, increasing redundancy will increase robustness. In an airline schedule, we can increase redundancy by increasing slack times, scheduling more reserve crews, maintaining more spare aircraft in the network, etc. For the purpose of the model we develop in this chapter, redundancy is equated with slack time.

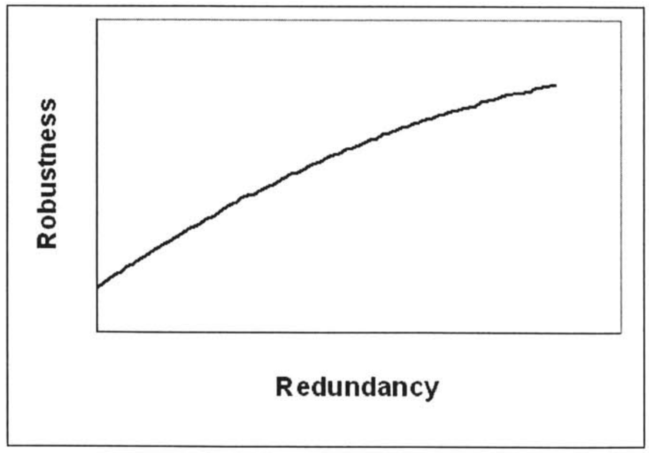


Figure 3.3-a: Redundancy vs. robustness

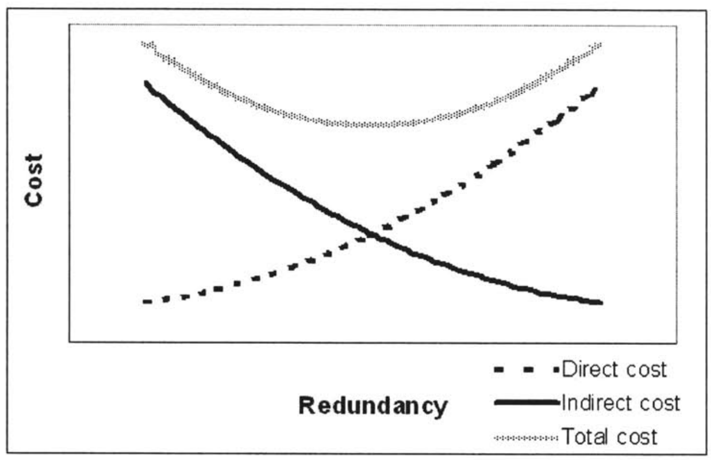


Figure 3.3-b: Redundancy vs. cost

Direct costs increase and indirect costs decrease with increasing redundancy. For an airline, the direct costs incurred as a result of increasing slack times are the cost of lower aircraft utilization and the indirect costs are primarily passenger and crew recovery costs. The challenge is to provide the system with enough redundancy (or enough robustness) to minimize total *realized* costs.

3.2.4 Flight Delay vs. Passenger Delay

Bratu and Barnhart (2004) [15] perform an intensive study on airline passenger delays and conclude that there is a major difference between flight delay and passenger delay, and with flight delays underestimating passenger delays in most cases. Using a Passenger Delay Calculator algorithm (PDC), they observe for a major U.S. airline in August 2000 that average flight delay is 15.4 minutes, while average passenger delay is 25.6 minutes. This difference is explained by passenger disruptions. Recall that a passenger is disrupted if at least one of his/her flight legs is cancelled or if he/she does not have enough time to connect between his/her scheduled flights (that is, they misconnect). For the majority of airline passengers who are not disrupted, average passenger delays are about equal to average flight delays. But the small fraction of passengers who are disrupted, experience long delays, accounting for 39% of all passenger delay minutes (Table 3.1).

Passenger type	Average delay	% Passengers	% Total passenger delays
Disrupted	303 minutes	3.2%	39
Non-disrupted	16 minutes	96.8%	61

Table 3.1: Delay of disrupted and non-disrupted passengers

Not only do flight and passenger delays not equal, there is not even a clear correlation between them. In other words, higher levels of flight delays and cancellations do not always translate into higher levels of passenger delays. Bratu and Barnhart (2004) [15] compare two days during August 2000 for a major U.S. airline (August 1^{st} and August 2^{nd}) and show that although day 2 has a higher average flight delay and cancellation rate than day 1, it has shorter average delays for disrupted passengers (Table 3.2).

Day	Average flight delay (minutes)	Cancellation rate	Average delay of disrupted passengers (minutes)
1	9.5	1.0%	495
2	40.4	8.1%	334

Table 3.2: Disrupted passenger delays vs. flight delays

To illustrate why flight delays and passenger delays are not always correlated, consider the example in Figure 3.4. Suppose we have two aircraft, one on route 1 and one on route 2. Also assume that both of these aircraft routes have a large amount of propagated delay. As a result, flights f and g both experience a relatively long total arrival delay (TAD). Passengers scheduled to connect from flight f to flight g can still make their connection, with relatively little delay. Now compare this scenario with another scenario where only aircraft route 1 is delayed and all flights on route 2 are on time. In this case, passengers connecting from flight f to flight g misconnect and experience relatively long delays.

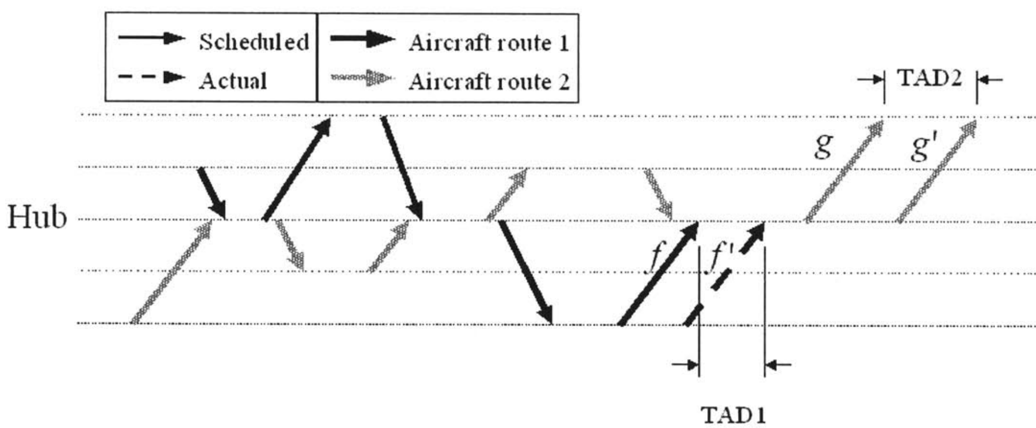


Figure 3.4: Passengers miss their connection because of a difference in relative flight delay

The above discussion highlights that flight and passenger delays are not always positively correlated. Hence, minimizing flight delays does not always result in minimized passenger delays. We use this objective, however, as a proxy for achieving minimized passenger delays. Although not an exact representative of passenger delay, high levels of average flight delay generally correspond to high levels of average passenger delay and high numbers of disrupted passengers (Figure 3.5). The values of passenger delay and disrupted passenger percentages are calculated using PDC (Appendix I) for our airline during August 2000. We evaluate the effectiveness of this proxy by measuring the actual change in passenger delay we achieve using the solution to our model.

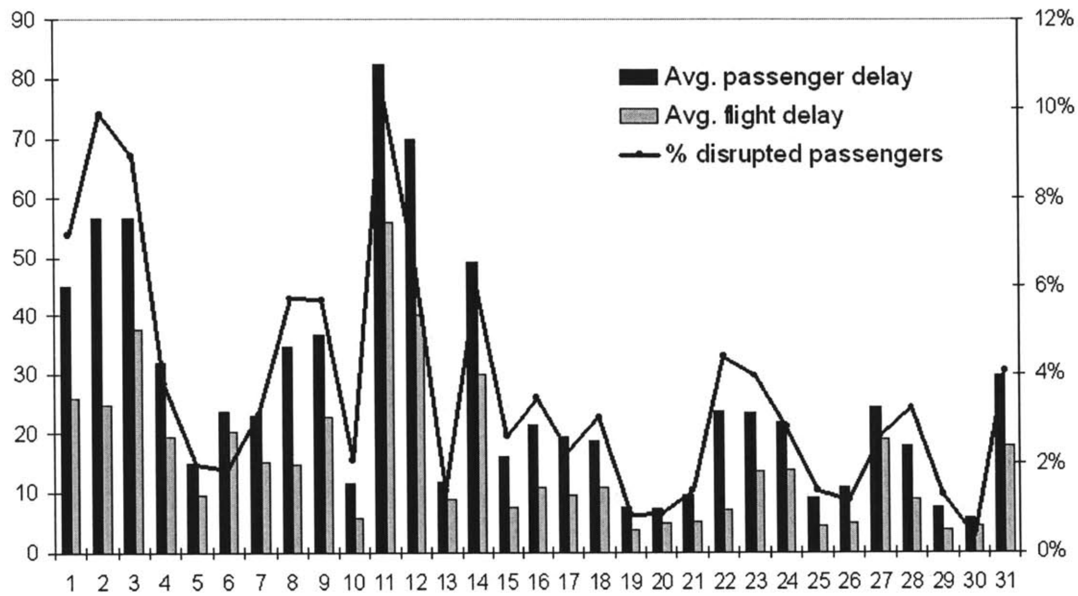


Figure 3.5: In most cases, flight and passenger delays track one another

3.3 Methodology

3.3.1 Retiming Flight Departures and Changing Routes

One way to reduce propagated delay is to increase slack time between selected pairs of consecutive flights and to reduce it elsewhere. Assume that flights f_1 and f_2 are consecutive flights in aircraft routing s and flights f_3 and f_4 are two different consecutive flights in the same routing. Also assume that historical delay data shows that flight f_1 is on average late ($f_1^{'}$) and its average arrival delay is larger than the current slack time between f_1 and f_2. Flight f_3, however, is on-time on average and the slack between flight f_3 and flight f_4 is seldom needed. In this case, reducing slack time elapsed between flights f_3 and f_4 and increasing it by an equal amount between flights f_1 and f_2 can reduce delays and improve robustness without increasing the time length of aircraft route s. To achieve this, we push earlier the planned departure of flight f_1 and push later the planned departure of flight f_3, as shown in Figure 3.6.

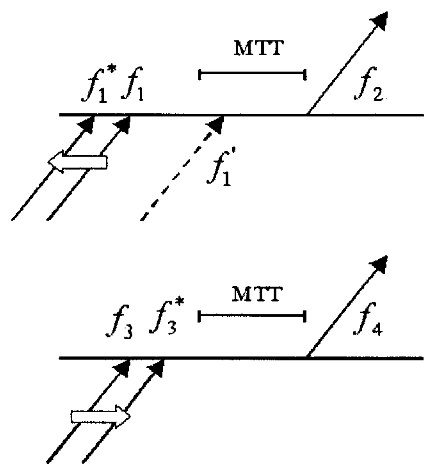


Figure 3.6: Illustration of the re-timing idea

Changing aircraft routings is another approach for improving schedule robustness. To illustrate the idea, consider Figure 3.7. Assume that flights f_1 and f_2 are consecutive flights in aircraft routing s_1 and flights f_3 and f_4 are consecutive flights in aircraft routing s_2. Historically, f_1 arrives late and delays the downstream flight, f_2, while f_3 historically arrives on time on the average. By re-routing aircraft to include flights f_1 and f_4 in the same route, and flights f_3 and f_2 in another route, we can create new, more robust routings in terms of flight delay propagation.

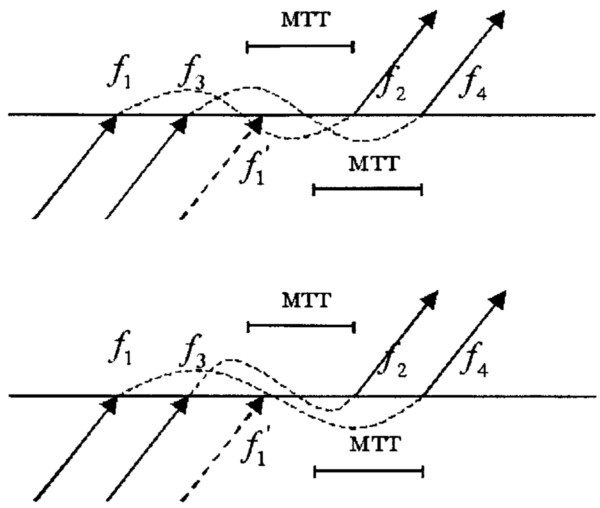


Figure 3.7: Illustration of the re-routing idea

We use these ideas in our model, described in Section 3.4, to distribute slack times and generate more robust routing solutions.

3.3.2 Time-Space Network

Underlying our model is a directed time-line network. The arc set in this network corresponds to the set of flights and the set of ground arcs allowing aircraft to wait at airports between flights. The node set corresponds to either a departure or an arrival of a flight. The time corresponding to a departure node is Planned Departure Time (PDT) and for arrival nodes is the Planned Arrival Time plus the minimum aircraft turn time (PAT + MTT). Ground arcs connect two consecutive nodes at a single location and include the wrap-around (or overnight) arc that connects the last node to the first node in any location. In our implementation, the duration of the time-line network is three days and wrap-around arcs connect the last node of day three to the first node of day one. We select a point in time, called the *count time*, to count the total number of aircraft of each type in the system (either on the ground or in the air). To minimize computation efforts, the count time is usually chosen to coincide with the time at which the maximum number of aircraft is on the ground (Barnhart et al. (1997) [5]). In Figure 3.8, we illustrate a schematic time-line network.

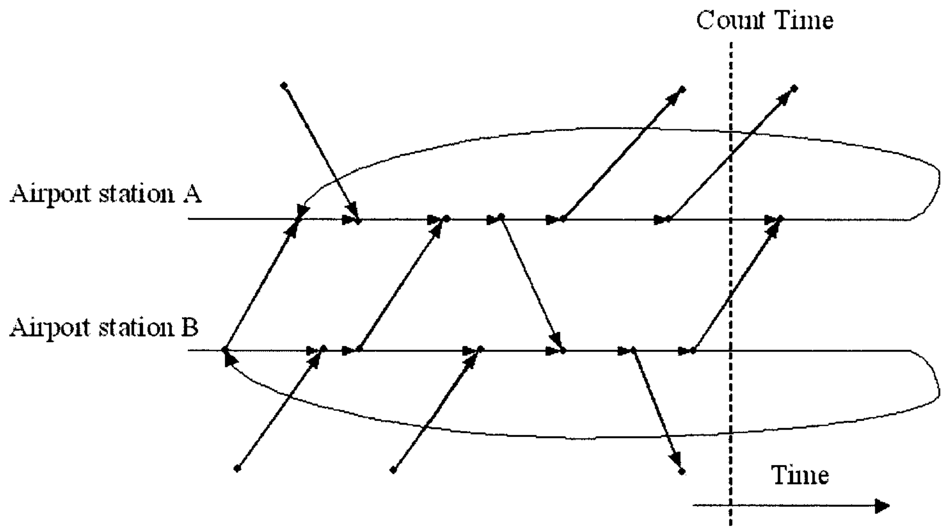


Figure 3.8: Illustration of the time-line network

Flight copies

To model flight departure re-timing in our formulation, we use the *flight copy* concept. A copy of a flight arc is a new flight arc in the time-space network parallel to the original flight arc, indicating a different departure time for that flight. Flight copies are generated in a short time window so that flight leg demands remain unchanged. In our formulation, we consider 5 copies of each flight leg in 5-minute intervals, as shown in Figure 3.9. We then add a constraint to ensure that among different copies of the same flight leg, exactly one is selected in the optimal solution.

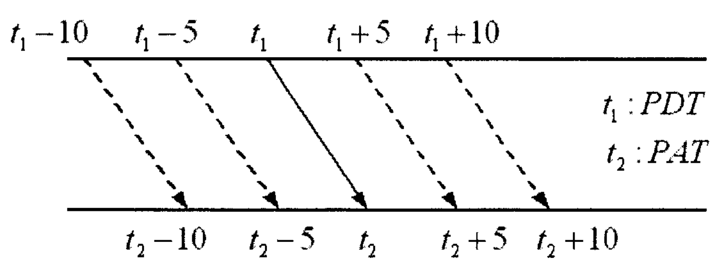


Figure 3.9: Illustration of flight copies

3.3.3 String-Based Formulation

We use a string-based formulation to solve our model. According to Barnhart et al. (1997) [5], a maintenance string, or more succinctly a string, is defined as a sequence of connected flights such that:

- It starts and ends at one of the airline's maintenance bases
- It satisfies flow balance, that is, the origin of the $(j+1)^{th}$ flight in the string is always the destination of the j^{th} flight for any $j \geq 1$.
- It is maintenance feasible, that is, it satisfies all FAA and airline requirements such as limits on the total or daily flying times, limits on the maximum number of days between maintenance, etc.

3.3.4 Calculating the propagated delay for a given string

We generate a large number of feasible strings and then solve our robust aircraft routing and flight departure re-timing model as a set partitioning problem with side constraints. The attractiveness of a given string is measured by it total expected propagated delay.

In section 3.1 we introduced the two components of flight delay, namely independent delay and propagated delay. According to our definition, independent delay refers to that part of flight delay that is independent of aircraft routing. In light of this property, we use the following algorithm to calculate the total expected propagated delay of a given string.

Algorithm 1: Calculating independent and propagated flight delays

1. For each day j in a selected period of time:
   a) Construct the planned aircraft routings.
   b) For each routing on day j, calculate the propagated delay for each flight in the routing:

$PD^j_{i-1,i} = \max(TAD^j_{i-1} - slack^j_{i-1}, 0)\ \forall i \in \{2,...,m\}$ (m is equal to the number of flights in the route).

c) For each routing, calculate the independent arrival delay for each flight in the routing. Let:

$IAD^j_1 = TAD^j_i$,

$IAD_i = TAD_i - PD_{i-1,i}\ \forall i \in \{2,...,m\}$.

2. Let $E[IAD_i] = \frac{1}{n}\sum_{j=1}^{n} IAD^j_i$, in which n is the number of days flight i is operated during the time period.

3. For a given string, calculate the total arrival delays and propagated delays as follows:

   for the first flight of the string, let: $E[TAD_1] = E[IAD_1]$, and

   for all other flights, let: $E[PD_{i-1,i}] = \max(E[TAD_{i-1}] - slack_{i-1,i}, 0)$, and

   $E[TAD_i] = E[TID_i] + E[PD_{i-1,i}]$, $\forall i \in \{2,...,m\}$.

4. Finally, the expected total propagated delay of string s is equal to $\sum_{i=2}^{m} E[PD_{i-1,i}]$.

3.4 Problem formulation

3.4.1 Notation

Before presenting our model to determine flight departure times and aircraft routings that minimize delay propagation, we define relevant decision variables, sets and parameters.

Decision variables:

$f_{i,n}$	equals 1 if copy n of flight i is in the solution; 0 otherwise
z_s	equals 1 if string s is chosen; 0 otherwise
y_{m,k,t^-}	number of aircraft of type k on the ground before time t at maintenance station m

y_{m,k,t^+}	number of aircraft of type k on the ground after time t at maintenance station m
$y_{g,k}$	number of aircraft of type k on ground arc g

Sets:

S_k	set of strings for fleet type k
F	set of flights
I	set of connecting itineraries (composed of flight leg i followed by flight leg j)
G	set of ground arcs
$I(m,k,t)$	set of strings of aircraft type k terminating at maintenance station m at time t
$O(m,k,t)$	set of strings of aircraft type k commencing at maintenance station m at time t
C_i	set of copies of flight leg i
M	set of maintenance stations
K	set of fleet types
T	set of string departure or arrival times at maintenance stations, indexed by order of increasing time

Parameters:

$E[PD_{ij}^s]$	the expected amount of delay propagated from flight i to flight j in string s
$a_{i,n}^s$	equals 1 if flight i copy n is in string s; 0 otherwise
N_k	number of available aircraft of type k
p_g	number of times ground arc g crosses the count time
r_s	number of times string s crosses the count time
$PDT_{i,n}$	planned departure time of flight i copy n
$PAT_{i,n}$	planned arrival time of flight j copy n
MCT	Minimum (passenger) Connection Time

3.4.2 The Model

$$\min \sum_{s\in S}\sum_{(i,j)\in s} E[PD_{ij}^{s}]z_s \qquad (3.6)$$

Subject to

$$f_{i,n} = \sum_{s} a_{i,n}^{s} z_s \qquad \forall i \in F, \forall n \in C_i \qquad (3.7)$$

$$-\sum_{s\in I(m,k,t)} z_s + \sum_{s\in O(m,k,t)} z_s - y_{m,k,t^-} + y_{m,k,t^+} = 0 \qquad \forall m \in M, \forall k \in K, \forall t \in T \qquad (3.8)$$

$$\sum_{s\in S_k} r_s z_s + \sum_{g\in G} p_g y_{g,k} \le N_k \qquad \forall k \in K \qquad (3.9)$$

$$\sum_{n\in C_i} f_{i,n} = 1 \qquad \forall i \in F \qquad (3.10)$$

$$\sum_{n\in C_j} f_{j,n} PDT_{j,n} - \sum_{n\in C_i} f_{i,n} PAT_{i,n} \ge MCT \qquad \forall (i,j) \in I \qquad (3.11)$$

$$z_s \in \{0,1\} \qquad \forall s \in S \qquad (3.12)$$

$$0 \le f_{i,n} \le 1 \qquad \forall i \in F, n \in C_i \qquad (3.13)$$

$$y_{g,k} \ge 0 \qquad \forall g \in G, \forall k \in K \qquad (3.14)$$

The objective function (Equation 3.6) is to minimize the total expected propagated delay of the selected strings. Constraints 3.7 guarantee that a string is selected if and only if all the flight copies in that string are selected. Constraints 3.8 are aircraft flow balance constraints, ensuring that the number of aircraft of type k flowing into and out of every maintenance station m at each time t is the same. Constraints 3.9 are aircraft count constraints that guarantee the number of aircraft of type k used at the count time does not exceed the total number of type k aircraft available. Constraints 3.10 are flight coverage constraints. They ensure that for any given flight, exactly one flight copy is chosen. Constraints 3.11 prevent choosing flight copies that do not include sufficient connection times in passenger itineraries. Constraints 3.12 require string variables to be binary. Note that because all f variables can be written as the sum of binary z variables, their integrality is ensured by integrality of z. Hence, constraints 3.13 bound f variables between 0 and 1. Finally, constraints 3.14 guarantee that the number of aircraft of type k on the ground is non-negative for each fleet type. Again, integrality of these variables is guaranteed by integrality of the z variables.

3.5 Implementation

Data

We generate string variables using the domestic flights included in our airline's schedule for August 31st, 2000. Each string's expected propagated delay is calculated based on the airline's schedule performance during July 2000, as extracted from the Airline Service Quality Performance (ASQP) database. Because some of the flight legs operated on August 31st, 2000 were not operated on all days in July 2000, we calculate average delays based on the number of days each flight is operated.

Assumptions

We made the following assumptions in order to build and implement our model:

1) In constructing flight strings, minimum and maximum aircraft turn times are assumed to be 20 minutes and 1 hour, respectively.
2) Aircraft maintenance checks are carried out during the night. Hence, to limit the generation of inefficient strings, all strings start before noon and end after 4pm.
3) All maintenance strings are one, two or three days long.
4) Independent flight delays are the same for different copies of the same flight leg.
5) Passenger itinerary demands are constant for different copies of the same flight leg.

Computational Environment

All data preprocessing was performed in the Java object oriented programming environment. We created a Java class called NetworkModel, which is responsible for first reading ASQP data and then instantiating the necessary Node, Flight and Arc objects that constitute the topology of the underlying time-line network. An instance of this NetworkModel is then passed to another class called StringModel, which generates a specified number of different MaintenanceString objects in a random fashion. A final Main class is responsible for communicating between these classes and constructing the necessary vectors and matrices to run the optimization model.

The model is implemented in OPL Studio as a Mixed Integer Program (MIP). The input file is constructed to take advantage of the sparse nature of the constraint matrix.

Problem Size

The network, composed of 1075 flights, is operated by at least four different major fleet types. Considering the large number of flight copies and string variables in our model, we decompose the problem by fleet type. In doing so, we relax minimum connection times (constraints 3.11). In our analysis, we consider the Boeing 757 fleet because of its manageable network size. After solving the model and obtaining the flight departure times, all passenger itineraries are evaluated to determine if any passenger connection times are less than the minimum allowed time. If so, for any itinerary with violated connection time, a new constraint is added to the model (in the form of constraints 3.11), the model is re-solved, and the process is repeated until all passenger connection times are legal.

Because the number of feasible maintenance string variables is very large (in the order of hundreds of millions) even for the single fleet type sub-network that we consider, it is computationally intractable to generate all of these strings and cast them into the model. Column generation can be used, starting with only a subset of the strings and generating strings as needed by solving pricing sub-problems successively. We have avoided such an approach in this thesis. Instead, we generate about 10,000 string variables and solve the resulting model. Hence, we do not guarantee that we generate optimal solutions to our model. Our goal, in fact, is to compare the behavior of the schedule we generate to that of the original airline's schedule.

The size of the problem we consider is summarized in Table 3.3.

Network size		Model size	
Daily flights	98	String variables	10,000
Flight copies	1,470	Flight copy variables	1,470
Nodes	2,940	Ground arc variables	2,939
Arcs	33,518	Constraints	2,373

Table 3.3: Problem size

Generating Strings

We generate strings of length one day, two days and three days. Before describing the algorithm to generate strings, we provide the following definitions:

Predecessor Node

Node i is said to be node j's predecessor if there is an arc connecting node i to node j. In Figure 3.10, nodes i_1, i_2 and i_3 are predecessors of node j.

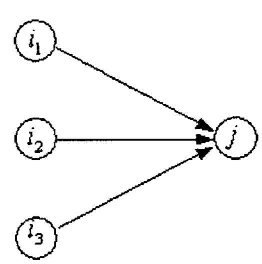


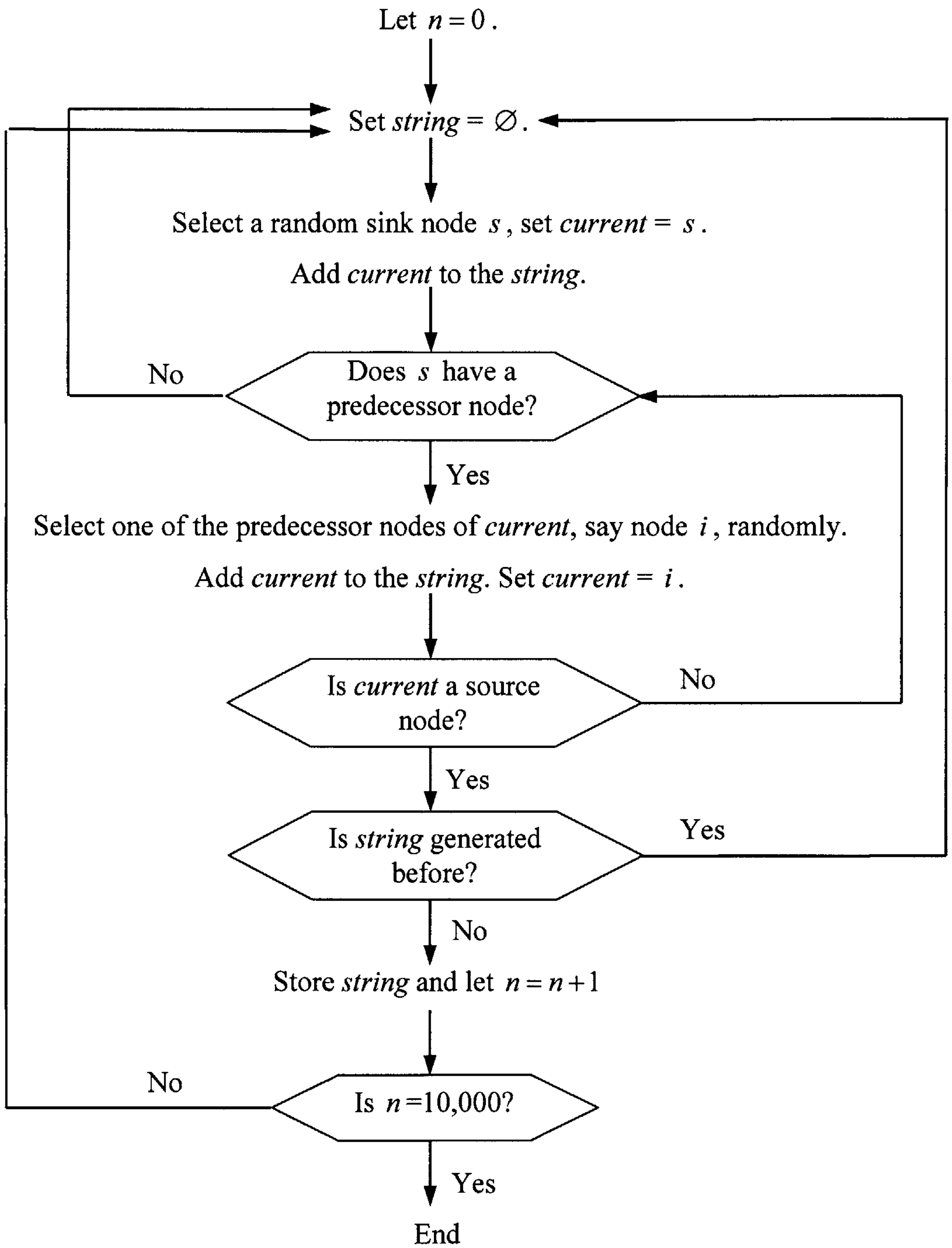
Figure 3.10: Illustration of a node's predecessors

Source and Sink Nodes

Every node (n) in the time-line network is defined by three properties: location (n_l), day (n_d) and time of day (n_t). A node n is defined to be a source node, the starting node of a string, if $n_l \in M$, n_d=1 and n_t<12:00, in which M is the set of maintenance stations.

Similarly, a node n is defined to be a sink node if $n_l \in M$ and n_t>16:00 (n_d can be 1,2 or 3). The algorithm to generate strings is thus as follows:

Algorithm 2: Generating strings



Every string is stored with its associated unique key number. This prevents storing duplicate strings in the set of maintenance strings. The above algorithm, tested in Java, created hundreds of thousands of strings in less than five seconds.

3.6 Results

Impact on Flight Delay

Using July 2000 data, we compute the model's parameters and solve the model to generate flight departure times and aircraft routings. We then generate the actual departure and arrival times of all Boeing 757 flight legs during August 2000, based on the new aircraft routings and flight departure times. Our model's parameters are determined using July 2000 delay information because in practice a model must be calibrated using historical data and then be applied to future operations. Algorithm 3 details this procedure.

Algorithm 3: Re-generating actual departure and arrival times

1) Determine the independent departure delay (IDD) and independent arrival delay (IAD) for each August 2000 flight leg based on delay data for July 2000.
2) Determine the new total departure delay ($\overline{TDD}$) and total arrival delay ($\overline{TAD}$) for each flight leg according to the new routings and flight departure times generated by our model, as follows:
    a) For the first flight j in any string, $\overline{TDD}_j = IDD_j$ and $\overline{TAD}_j = IAD_j$.
    b) For all other flights in the string, $\overline{TDD}_j = IDD_j + \overline{PD}_{ij}$ and $\overline{TAD}_j = IAD_j + \overline{PD}_{ij}$, where flight i is the flight immediately before flight j in the string and $\overline{PD}_{ij}$ is the new propagated delay associated with our new routings and planned flight departure and arrival times.
3) Determine the new actual departure and arrival times for each flight leg j as follows:
    - $\overline{ADT}_j = PDT_j + \overline{TDD}_j$
    - $\overline{AAT}_j = PAT_j + \overline{TAD}_j$

With the new actual departure and arrival times, we calculate the total propagated delay of our new strings. We then compare the total monthly propagated delay of our new aircraft routings for August 2000 with that of the planned August 2000 aircraft routings used by the airline. The results are summarized in Table 3.4. Column *Old PD* indicates the total monthly propagated delay based on the original August 2000 routings. Column *New PD without flight copies* indicates the total monthly propagated delay of the August 2000 routing solution suggested by our revised model in which flight departure retiming is not allowed (that is, we do not include more than one copy of any flight leg in our model). Column *New PD with flight copies* indicates the total monthly propagated delay of the August 2000 routing solution for the case in which we allowed flight departure re-timing.

Old PD	New PD without flight copies	% of PD reduced	New PD with flight copies	% of PD reduced
14,690	8,279	43.64%	7,816	46.79%

Table 3.4: Propagated delays for August 2000, Boeing 757 fleet type

Using flight copies in the model reduces flight delays only 3.15% compared to the case where no flight copies are considered as illustrated in Table 3.4. This suggests that adding the flight copies to the model might not be worthwhile given the small decrease in total propagated delay and the relatively large computational complexity introduced with the additional flight copies.

Because we do not use column generation, the routing solution we obtain is not guaranteed to be optimal. To produce improved solutions, we ran the model 5 times; each time using 10,000 randomly generated strings. We then selected the best solution. This resulted in obtaining solutions that reduced the August 2000 propagated delay for both re-routing-and-re-timing and re-routing-only scenarios by about 3-5%.

Impact on Passenger Delay

We investigate the effects of our new aircraft routings and departure times on passenger delays and disruptions. After calculating the new actual flight departure and arrival times using algorithm 3, we use PDC (Appendix I) to estimate the values of passenger delay for the old and new plans. We summarize the results for the entire month of August 2000 in Table 3.5. We believe that applying our model to all fleet types instead of only one single fleet will result in larger reductions in passenger delays and disruptions.

	Old	New	% reduction
Total passenger delay minutes	68,578,419	66,795,380	2.6%
Total number of disrupted passengers	91,489	90,391	1.2%

Table 3.5: Old and new passenger delays and disruptions

It is important to note that the total passenger delay minutes and total number of disrupted passengers are likely overestimated for the new routing and flight departure time solution. In our analysis, we assume that if a flight is cancelled in actual operations, it is cancelled in our new simulated operations as well. We expect, however, the cancellation rate to decrease as a result of the flight delay reductions achieved by the new plan.

Chapter 4

Designing Robust Airline Schedules that Minimize Expected Passenger Delay

In this chapter we present two robust airline scheduling models to decrease expected passenger delays. The model presented in section 4.2 is an integrated flight departure re-timing and aircraft routing model that minimizes the expected number of misconnecting passengers. Like the model presented in the previous chapter, this model extends the model proposed by Lan, Clarke and Barnhart (2003) [39]. The model presented in section 4.3 is a flight departure re-timing model, based on the idea of providing potential misconnecting passengers with alternative recovery options. In this chapter, we introduce our ideas and models and illustrate the underlying concepts with examples.

4.1 Misconnecting Passengers

In both models presented in this chapter, we focus on misconnecting passengers. In this section, we argue the importance of paying attention to disrupted passengers and then we perform analysis related to the proportion of disrupted passengers represented by misconnecting passengers.

In section 1.2, we observed that passengers are disrupted either because one (or more) of their flight legs is cancelled or because they have insufficient time to connect between their scheduled flights. As we stated, although disrupted passengers constitute a tiny fraction of the total passengers (less than 5%), they account for more than one third of the total passenger delay minutes for our airline (Bratu and Barnhart (2004) [15]). Not only do disrupted passengers suffer from delays, they inflict costs on the airline. The airline has to re-book them on other flights and in some cases, must pay for their hotel and meal voucher expenses. A second cost associated with passenger disruption is the loss of "passenger good will". A business passenger, who paid a premium ticket price and missed a business meeting as a result of disruption, might choose another airline the next time he/she flies.

Disrupted passengers suffer relatively long delays, incur extra costs and might not come back. Hence, minimizing the number of disrupted passengers can be a good objective function choice in trying to build robust airline schedules.

During August 2000 for our airline, only 34.3% of the disrupted passengers misconnected (type D2), and the rest 65.7% were disrupted as a result of fight cancellation (type D1), as shown in Table 4.1. This follows because passengers disrupted by flight cancellation are disrupted in bulk numbers, while the number of passengers traveling on a specific disrupted connecting itinerary is typically small. According to Bratu and Barnhart (2004) [15], "when a flight cancellation occurs, an average of 81.0 passengers get disrupted, while only 3.4 passengers are disrupted per missed connection". The result is that the delays of type D1 and type D2 passengers represent 59.7% and 40.3%, respectively, of the total disrupted passenger delay (Table 4.2).

Disrupted passenger type	Disruption cause	Percentage
D1	Flight cancellation	65.7%
D2	Misconnection	34.3%

Table 4.1: Flight cancellation causes the majority of passenger disruptions

Disrupted passenger type	Disruption cause	% of the total disrupted passenger delay minutes
D1	Flight cancellation	59.7%
D2	Misconnection	40.3%

Table 4.2: The largest source of passenger delay is flight cancellation

Given that more than two thirds of the disrupted passengers are of type D1, a valid question is why don't we try to minimize their number instead of number of D2 passengers? Or, why don't we try to minimize the potential number of flight cancellations? The answer is that minimizing flight cancellations does not result in a policy that minimizes passenger delay. In Fact, flight cancellations can be an effective mechanism for reducing passenger delay, when the appropriate flight legs are cancelled. Rosenberger, Johnson and Nemhauser (2001) [43] provide a short cycle algorithm to facilitate cancellations and mitigate their impacts on passengers.

Another issue with trying to minimize the number of flight cancellations in planning is that in many cases, canceling a flight is unavoidable due to the presence of inclement weather, aircraft mechanical problems or crew unavailability.

4.2 Robust Aircraft Maintenance Routing with Time Windows: Minimizing the Total Expected Number of Misconnected Passengers

4.2.1 Basic Idea

Lan, Clarke and Barnhart (2003) [39] present a flight departure re-timing model to minimize the total expected number of misconnecting passengers. They ensure that the solution of their re-timing model does not violate the aircraft routing plan. We present a formulation that combines their model with a string based aircraft routing model allowing aircraft routes to be changed simultaneously with the flight departure times. Our goal is to generate a schedule resulting in lower expected numbers of misconnected passengers.

The basic idea of Lan, Clarke and Barnhart's model is illustrated in Figure 4.1. Suppose there are 100 passengers connecting from flight f_1 to flight f_2. The head of the arcs in this figure correspond to the flight arrival time plus the 30-minute aircraft minimum turn time (MTT). For simplicity, we assume that the minimum passenger connection time (MCT) is also 30 minutes.

Assume that 70% of the time the arrival time of flight f_1 plus MTT is between t_1 and t_2. Hence, because we have assumed MTT = MCT, 70% of the time passengers make their connection on this itinerary, and with probability 0.3 passengers miss their connection. Also assume that 20% of the time, the arrival time of flight f_1 plus MTT is larger than $t_3 = t_2 + 10$ minutes. Hence, if in the planning stage, we postpone the departure of flight f_2 from t_2 to t_3 we can decrease the misconnection probability by 10%. In other words, the expected number of misconnected passengers decreases from 30 to 20.

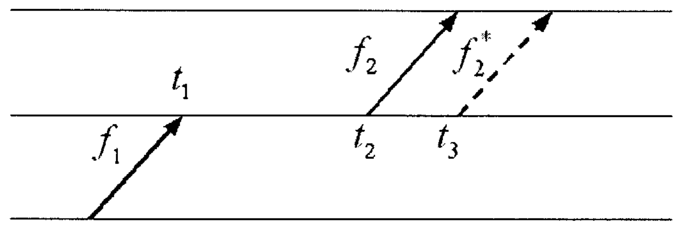


Figure 4.1: Adding slack can decrease misconnection probability

Another way to decrease the misconnection probability is to move earlier the departure time of flight f_1. Although adding slack to every passenger connection can be effective in decreasing the number of misconnected passengers, it results in reduced aircraft productivity. The challenge then is to add slack to passenger connections where the misconnection probability is high and to remove it where the misconnection probability is low. The goal is to determine a revised schedule that can be operated with the same number of aircraft, but allows fewer expected numbers of misconnected passengers.

Combining aircraft routing decisions with flight re-timing decisions gives us more flexibility to move slack in the network. This fact is illustrated in Figure 4.2. Assume that passengers connect from flight f_3 to flight f_4. Also assume that according to historical data, flight f_3 has relatively long average delay, causing a high misconnection probability for passengers connecting from f_3 to f_4.

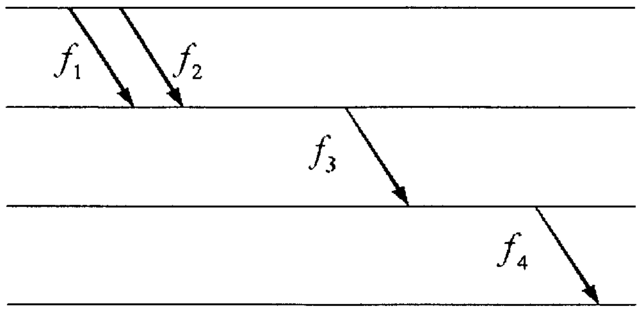


Figure 4.2 Combined re-routing and re-timing can be more effective

One approach to decrease the expected number of misconnecting passengers is to re-time flight legs f_3 and f_4. If , however, there are limitations on moving the departure times of flights f_3 and f_4, so that retiming alone does not sufficiently increase passenger connection time, it is advisable to also adjust the up-stream routing of flight f_3. If we change the routing of flight leg f_3 from f_2 - f_3 to f_1 - f_3, we can decrease the misconnection probability for passenger itinerary f_3 - f_4, if routing f_1 - f_3 has less propagated delay than routing f_2 - f_3.

To calculate the expected number of misconnected passengers for each passenger itinerary, we begin by generating flight copies, to create new passenger connection opportunities. Let $X_{i,n}^{j,m}$ be a binary variable equal to 1 if the passenger connection consisting of flight i copy n and flight j copy m is in the solution and zero otherwise (Figure 4.3). Also, let n_{ij} be the number of passengers connecting from flight i to flight j. Then, we represent number of passengers who miss their connection on this itinerary copy, denoted $dp_{i,n}^{j,m}$, as:

$$dp_{i,n}^{j,m} = \begin{cases} n_{ij} \text{ with probability } p \\ 0 \text{ with probability } 1-p \end{cases} \tag{4.1}$$

The probability p can be determined as follows:

$$p = prob(ADT_{j,m} - AAT_{i,n} < MCT), \quad (4.2)$$

where $ADT_{j,m}$ is the actual departure time of flight j copy m and $AAT_{i,n}$ is the actual arrival time of flight i copy n.

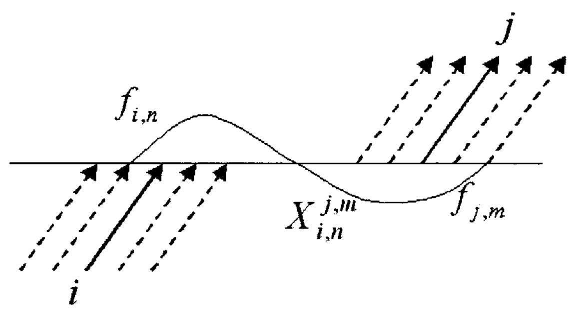


Figure 4.3: Generating flight leg copies creates passenger connection opportunities

Once the aircraft routes are known, ADT and AAT of each flight can be calculated using the approach introduced in algorithm 3 of chapter 3. For example, if for 5 days out of 30 days during July 2000, $ADT_{j,m} - AAT_{i,n}$ is less than the minimum passenger connection time, we let p equal to 5/30 for the itinerary composed of flight copies $f_{i,n}$ and $f_{j,m}$. In our model, however, aircraft routes are not known in advance. To calculate the probability p, we propose approximate procedures. The probability that passengers miss their connection is a function of the passenger planned connection time (PCT) and also the delays of incoming and outgoing flights, which in turn are a function of preceding operations. Using historical operations data, we compute the probability of misconnection as a function of the PCT in a specific location. In Figure 4.4, we illustrate the misconnection probability at one of our airline's hubs in August 2000.

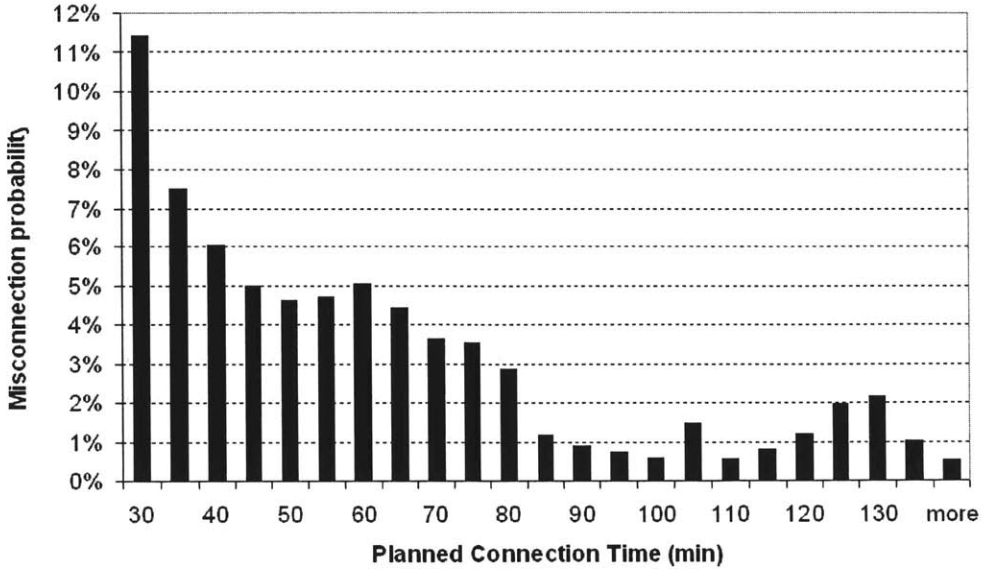


Figure 4.4: Misconnection probability as a function of PCT

For the purpose of our model, we compute misconnection probabilities for each bank during the day at each hub, using historical data (if enough data is available), or based on simulation. Typically, the misconnection probability is greater for later banks, even with similar passenger planned connection times. This is due to delay propagation, as illustrated in Figure 4.5.

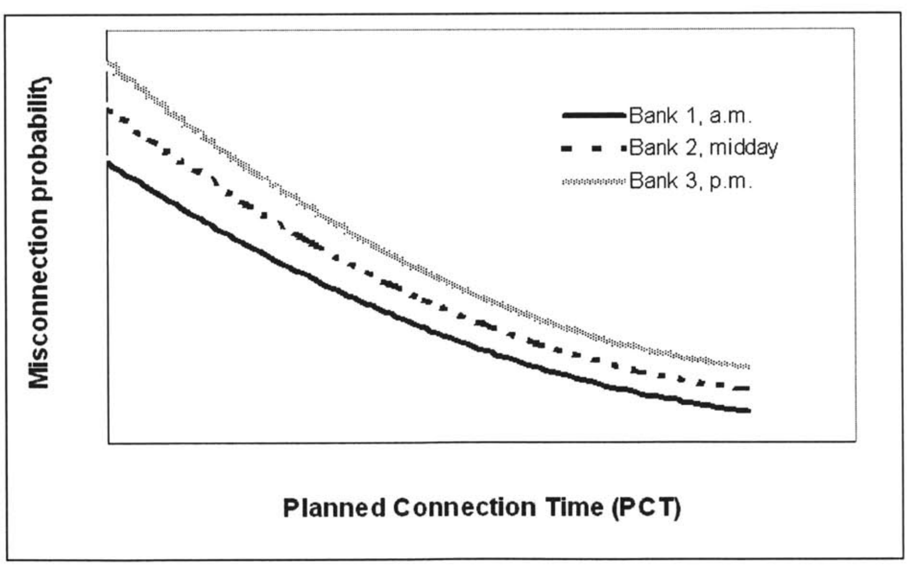


Figure 4.5: Misconnection probability as a function of PCT for banks at different times in the day

Using these estimated probabilities, the misconnection probability p of an itinerary $X_{i,n}^{j,m}$ can be readily calculated, and the expected number of disrupted passengers for the itinerary composed of flight copies $f_{i,n}$ and $f_{j,m}$ can be calculated as:

$$E[dp_{i,n}^{j,m}] = p \times n_{ij} . \tag{4.3}$$

4.2.2 Notation

The notation used in this model is very similar to the notation used for our previous model in chapter 3. We introduce new sets, parameters and passenger connection decision variables to allow us in formulating our new model.

Decision variables:

$f_{i,n}$	equals 1 if copy n of flight i is in the solution; 0 otherwise
z_s	equals 1 if string s is in the solution; 0 otherwise
y_{m,k,t^-}	number of aircraft of type k on the ground before time t at maintenance station m
y_{m,k,t^+}	number of aircraft of type k on the ground after time t at maintenance station m
$y_{g,k}$	number of aircraft of type k on ground arc g
$x_{i,n}^{j,m}$	equals 1 if copy n of flight i and copy m of flight j are in the solution; 0 otherwise

Sets:

S_k	set of strings of fleet type k
F	set of flights
F^I	set of flights from which passengers connect

F^O	set of flights to which passengers connect
I	set of connecting itineraries
G	set of ground arcs
$I(m,k,t)$	set of strings of aircraft type k ending at maintenance station m at time t
$O(m,k,t)$	set of strings of aircraft type k starting in maintenance station m at time t
C_i	set of copies of flight i
M	set of maintenance stations
K	set of fleet types
T	set of string departure or arrival times at maintenance stations, indexed by order of increasing time

Parameters:

$E[dp_{i,n}^{j,m}]$	expected number of disrupted passengers on the connecting itinerary consisting of flight i copy n and flight j copy m
$a_{i,n}^s$	equals 1 if flight i copy n is in string s; 0 otherwise
N_k	number of available aircraft of type k
p_g	number of times ground arc g crosses the count time
r_s	number of times string s crosses the count time
$PDT_{i,n}$	planned departure time of flight i copy n
$PAT_{j,n}$	planned arrival time of flight j copy n
MCT	Minimum (passenger) connection time

4.2.3 The Model

$$\min \sum_{i\in F^o}\sum_{n\in C_i}\sum_{j\in F^I}\sum_{m\in C_j} E[dp_{i,n}^{j,m}]x_{i,n}^{j,m} \tag{4.4}$$

Subject to

$$f_{i,n} = \sum_{s} a_{i,n}^{s} z_s \qquad \forall i \in F, \forall n \in C_i \qquad (4.5)$$

$$-\sum_{s \in I(m,k,t)} z_s + \sum_{s \in O(m,k,t)} z_s - y_{m,k,t^-} + y_{m,k,t^+} = 0 \qquad \forall m \in M, \forall k \in K, \forall t \qquad (4.6)$$

$$\sum_{s \in S_k} r_s z_s + \sum_{g \in G} p_g y_{g,k} \le N_k \qquad \forall k \in K \qquad (4.7)$$

$$\sum_{n \in C_j} f_{j,n} PDT_{j,n} - \sum_{n \in C_i} f_{i,n} PAT_{i,n} \ge MCT \qquad \forall (i,j) \in I \qquad (4.8)$$

$$\sum_{n \in C_i} f_{i,n} = 1 \qquad \forall i \in F \qquad (4.9)$$

$$f_{i,n} = \sum_{m \in C_j} x_{i,n}^{j,m} \qquad \forall (i,j) \in I, \forall n \in C_i \qquad (4.10)$$

$$f_{j,m} = \sum_{n \in C_i} x_{i,n}^{j,m} \qquad \forall (i,j) \in I, \forall m \in C_j \qquad (4.11)$$

$$0 \le f_{i,n} \le 1 \qquad \forall i \in F, n \in C_i \qquad (4.12)$$

$$0 \le x_{i,n}^{j,m} \le 1 \qquad \forall (i,j) \in I, \forall n \in C_i, \forall m \in C_j \qquad (4.13)$$

$$z_s \in \{0,1\} \qquad \forall s \in S \qquad (4.14)$$

$$y_{g,k} \ge 0 \qquad \forall g \in G, \forall k \in K \qquad (4.15)$$

The objective function of this model (Equation 4.4) is to minimize the total expected number of disrupted passengers. Constraints 4.5, 4.6, 4.7, 4.8 and 4.9 are the same as the corresponding constraints in our model in chapter 3, ensuring compatibility of flights and strings, flow balance, aircraft count, passenger connection feasibility and flight coverage respectively (refer to section 3.4.2 for more discussion). Constraints 4.10 and 4.11 guarantee that a passenger connection is selected if and only if both of its constituent incoming and outgoing flight copies are selected. Constraints 4.12 to 4.15 are non-negativity and integrality constraints. Again, by enforcing integrality requirements on the maintenance string variables (z's), integrality of the remaining decision variables is guaranteed.

A passenger who is disrupted at the beginning of the day is easier to recover than a passenger who is disrupted at the end of the day. The amount of delay a disrupted passenger experiences depends on the quality and availability of recovery options for that passenger. Nonetheless, although disrupted passengers are not equal in the sense that they experience different levels of passenger delay, the objective of minimizing the number of misconnected passengers is a good proxy for minimizing the overall passenger delay, and it translates to a model with reduced computational complexity and enhanced tractability.

4.3 Robust Flight Departure Re-Timing Model: Minimizing Expected Passenger Delay

4.3.1 Basic Idea

In spite of all the actions taken in the scheduling phase to reduce expected flight and passenger delays, passengers still get disrupted. In this section, we present an idea that might be effective to reduce overall passenger delay once a passenger is disrupted.

As mentioned in Chapter 1, passenger delay is the positive difference between the actual and scheduled passenger arrival times. When passengers get disrupted for any reason, the airline tries to re-assign them on a *recovery itinerary*. A recovery itinerary is defined as a set of consecutive flights (usually one to three), starting from the point of disruption and ending at the passenger's desired destination. At the point of disruption then, the airline agent will try to build a list of recovery options for the disrupted passenger and assign him/her to the *best* recovery itinerary. We define the best recovery itinerary as the one with at least one available seat and with earliest arrival time at the final destination.

Sometimes, such a recovery itinerary does not exist because the arrival time is too late at the destination. In this case, the passenger must either overnight or be recovered by another airline. Both of these cases translate into direct and indirect costs for the airline and long delays for the passengers themselves.

A feasible recovery itinerary does not exist if either of the following conditions is met:

1. No recovery itinerary exists for the rest of the day; or
2. Recovery itineraries exist but none has extra capacity to accommodate a disrupted passenger.

In the model we present in this section, we try to decrease the probability of the first case, that is, that no recovery itinerary exists for the remainder of the day. We restrict our attention to disrupted passengers who miss their connections only, and we allow only flight leg re-timings within defined time windows. We do not allow the original aircraft routings to be changed.

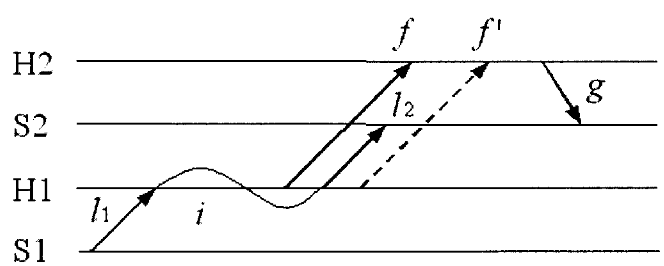


Figure 4.6: Postponing flight f creates a new recovery option for connecting itinerary i

To illustrate the idea, consider Figure 4.6. Assume that passengers are traveling on connecting itinerary i from spoke city S1 to spoke city S2 through hub H1. Also assume that flight leg l_1

(the incoming flight) has a long arrival delay and passengers on itinerary $l_1 - l_2$ miss their connection at H1. The disrupted passengers either have to wait for another direct flight from H1 to S2, or travel to S2 through another hub, say H2. The first option (waiting for a direct flight from H1 to S2) is not always available (for example when passengers are connecting on the last bank of the day). The availability of the second option depends on the availability of a flight f departing H1 after the departure time of l_2, and on another appropriately scheduled flight g from H2 to S2. If flight f is scheduled to depart H1 just before the departure time of l_2, postponing it (in the scheduling phase) five to ten minutes can create a new recovery itinerary for disrupted passengers, and potentially eliminate overnight delays for them.

To have a better understanding of how using this idea can reduce passenger delay, consider the example from our airline's schedule in August 2000, shown in Figure 4.7. Connecting BOS-DFW itinerary i consists of two flights, with a planned connection in CLE of 1605 – 1509 = 56 minutes. There is no other direct flight leaving CLE to DFW after 1605. Hence, disrupted itinerary i passengers must overnight in CLE. There is, however, another flight leaving CLE at 1605 to IAH. By postponing the departure time of this flight ten minutes from 1605 to 1615 we can create a recovery itinerary for i and reduce the expected value of passenger delay. Figure 4.8 illustrates the initial and recovery itineraries for this example on a map.

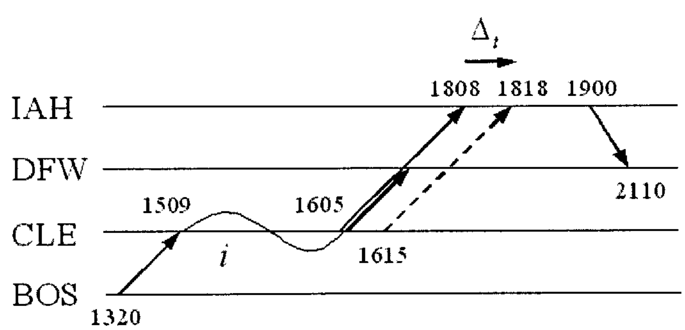


Figure 4.7: creating a new recovery option

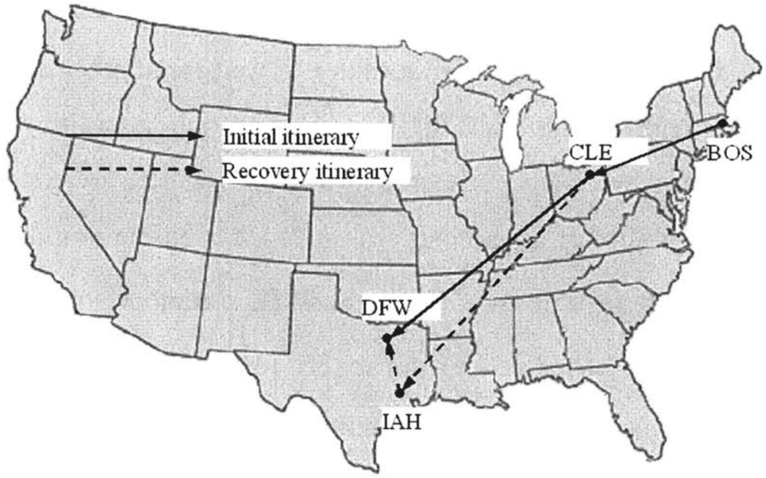


Figure 4.8: Initial and recovery itineraries for BOS-CLE-DFW itinerary

Analyzing our airline's data during August 2000 using PDC, we find that if we decrease the value of the minimum passenger connection time from 30 minutes to 20 minutes, the expected total number of misconnected passengers for August 2000 decreases from 76,387 to 56,671, a 25% decrease. Hence, we believe that even small changes (± 10 minutes) in the planned departure times can have significant impact on passenger delays.

4.3.2 Notation

Decision variable:

q_i	equals 1 if at least one recovery option exists for itinerary i; 0 otherwise
x_i^r	equals 1 if recovery option r for itinerary i is in the solution; 0 otherwise
$f_{j,n}$	equals 1 if copy n of flight j is in the solution; 0 otherwise

Sets:

I	set of itineraries
R^i	set of recovery itineraries for itinerary i, with each recovery itinerary specified by the flight copies it contains
F	set of flights
F^r	set of flight copies (j,n), $j \in F$, $n \in C_j$, belonging to recovery option r
C_j	set of copies of flight j

Parameters:

$PDT_{i,n}$	planned departure time of flight i copy n
$PAT_{j,n}$	planned arrival time of flight j copy n
MCT	Minimum (passenger) connection time

4.3.3 The Recovery Options Model

$$\text{Min} \sum_{i \in I} (1 - q_i) \tag{4.16}$$

Subject to

$$q_i \le \sum_{r \in R^i} x_i^r \qquad \forall i \in I \tag{4.17}$$

$$x_i^r \le f_{j,n} \qquad \forall i \in I; \forall r \in R^i; \forall (j,n) \in F^r, j \in F, n \in C_j \tag{4.18}$$

$$\sum_{n \in C_j} f_{j,n} = 1 \qquad \forall j \in F \tag{4.19}$$

$$\sum_{n \in C_j} f_{j,n} PDT_{j,n} - \sum_{n \in C_i} f_{i,n} PAT_{i,n} \geq MCT \qquad \forall (i,j) \in I \tag{4.20}$$

$$x_i^r \in \{0,1\} \qquad \forall i \in I, \forall r \in R^i \tag{4.21}$$

$$f_{j,n} \in \{0,1\} \qquad \forall j \in F, n \in C_j \tag{4.22}$$

$$q_i \in \{0,1\} \qquad \forall i \in I \tag{4.23}$$

The objective function (Equation 4.16) is to minimize the number of connecting itineraries without a recovery option. Constraints 4.17 state that any itinerary i has a recovery option only when at least one x_i^r variable is in the solution. If $\sum_{r \in R^i} x_i^r$ is strictly greater than 1 for some itinerary i, the corresponding binary variable q_i will equal one. Constraints 4.18 guarantee that a recovery option is in the solution only if all of its constituent flight copies are in the solution. Constraints 4.19 ensure that exactly one copy of each flight leg in our initial set of legs is included in the solution. Constraints 4.20 ensure that flight copies are selected that allow adequate connection time for each passenger itinerary. Finally, constraints 4.21, 4.22 and 4.23 ensure integrality of the solution.

We do not have to consider all itineraries in this model. Instead, we can focus on a subset of "important" itineraries. For example, we can try to ensure that itineraries late in the day have at least one recovery option, and high-demand itineraries have several alternate recovery options. This would require only minor modifications to the model. Itineraries with several recovery options, however, need not be considered.

Although we did not differentiate among different recovery options, it is rather straightforward to do so. We simply assign weights to recovery options depending on the option's associated passenger delay.

Chapter 5

Conclusion and Directions for Future Research

5.1 Conclusion

The objective of this thesis is to propose methods to develop an airline's schedule in the planning stage so that it is robust during operations. A robust plan, although not necessarily optimal in planning, performs best in operations, as measured by overall flight and passenger delay.
Because it is difficult to determine how much airlines are willing to pay to make their schedules more robust, we developed models and algorithms that generate solutions without significantly increasing operating costs. Our approaches are:

1) Flight departure retiming in a small time window, and
2) Aircraft re-routing.

We developed an integrated flight departure re-timing and aircraft re-routing model in Chapter 3. Considering the large number of flight copies and string variables in our model, we decomposed the problem by fleet type and only considered the Boeing 757 fleet because of its manageable network size. We used historical data (July 2000) from a major U.S. airline to calculate the parameters of our model. We then solved our model to select aircraft routings for the future schedule (August 2000) and evaluated the robustness of the new schedule by comparing the actual August 2000 disruptions with disruptions resulting from our new schedule. We observed that our model reduces total August 2000 propagated delay by more than 46%, and decreases the total passenger delay minutes and numbers of disrupted passengers.

We observed that using flight copies in the model reduces flight delays 3.15% compared to the case where no flight copies are considered. We concluded that adding flight copies to the model might not be worthwhile given the small decrease in total propagated delay and the corresponding increase in computational complexity of adding flight copies to our model.

In Chapter 4, we proposed two additional robust scheduling models to decrease expected passenger delays. The first model is an integrated flight departure re-timing and aircraft routing model that minimizes the expected number of misconnecting passengers. The second model is a flight departure re-timing model, based on the idea of providing potential misconnecting passengers with alternative recovery options. We illustrate the underlying concepts of each model with examples.

5.2 Directions for Future Research

The following list provides some potential future research directions:

1) **Implement "Flight departure re-timing and aircraft re-routing to minimize propagated flight delay" (Model) 1 for all fleet types**

   To reduce computational efforts, we implemented Model 1 for a single fleet type. Hence, we were not able to measure the total potential of this model to reduce passenger delays and disruptions. To minimize passenger delays and reduce the number of disrupted passengers, a network-wide minimization of propagated flight delays is necessary.

2) **Implement "Flight departure re-timing and aircraft re-routing to minimize number of passenger misconnections" (Model 2) and "Flight departure re-timing to provide alternative passenger recovery options for selected connecting itineraries" (Model 3).**

   Although we illustrated these models with examples, we did not provide implementation results. Model 2 can be implemented using a similar framework to that used for Model 1 in Chapter 3. The problem size might create some tractability issues, however, because we need to consider all fleet types in the model simultaneously. Limiting the number of connecting itineraries for which we seek additional recovery options can decrease the size of Model 3.

3) Consider itinerary demands and aircraft seat capacities in Model 3

In Model 3, we assumed any combination of flight legs that begins at the disruption location of a passenger and ends at his/her destination is regarded as a recovery option. We did not, however, consider the possibility that these flight legs might have no available seats to accommodate disrupted passengers. Enhancing the model with itinerary demand-aircraft capacity interactions can result in obtaining more accurate results.

Bibliography

1) Abara J. (1989). "Applying Integer Linear Programming to the Fleet Assignment Problem". Interfaces, 19.
2) Ageeva Y., J. P. Clarke (2000). "Robust Scheduling". Presentation at MIT-NEXTOR symposium. http://www.nextor.org/pubs/symposium00/
3) Ahuja R., T. L. Magnanti and J. B. Orlin (1993). "Network Flows". Prentice Hall.
4) Anbil R., R. Tanga and E. Johnson (1992). "A Global Approach to Crew-Pairing Optimization". IBM Systems Journal, 31: 71-78.
5) Barnhart C., N. L. Boland, L. W. Clarke and R. G. Shenoi (1998). "Flight String Models for Aircraft Fleeting and Routing". Transportation Science, 32: 208-220.
6) Barnhart C. and A. Cohn (2004). "Airline Schedule Planning: Accomplishments and Opportunities". Manufacturing and Service Operations Management, 6: 3-22.
7) Barnhart, C., A. Farahat and M. Lohatepanont (2002). "Airline Fleet Assignment: An Enhanced Revenue Model," submitted with revisions to Operations Research.
8) Barnhart C., E. Johnson, R. Anbil and L. Hatay (1994). "A Column Generation Technique for the Long-Haul Crew Assignment Problem". Optimization in Industry: Volume 2, T. Ciriano and R. Leachman (eds.). John Wiley and Sons, England, 7-22.
9) Barnhart C., T. Kniker and M. Lohatepanont (2001). "Itinerary-Based Airline Fleet Assignment". Transportation Science, 36: 199-218.
10) Barnhart C., F. Lu, and R. Shenoi (1998). "Integrated Airline Scheduling. Operations Research in the Air Industry". International Series in Operations Research and Management Science, 9: 384-403.
11) Barnhart C. and R. Shenoi (1998). "An Approximate Model and Solution Approach for the Long-Haul Crew Pairing Problem". Transportation Science, 32: 221-231.
12) Barnhart C. and K. Talluri (1997). Chapter 10: Airline Operations Research. Design and Operation of Civil and Environmental Engineering Systems: 435-469.
13) Beasley J. and B. Cao (1996). "A Tree Search Algorithm for the Crew Scheduling Problem". European Journal of Operations Research, 94: 517-526.

14) Berge, M. (1994) "Timetable Optimization: Formulation, Solution Approaches and Computational Issues". AGIFORS proceedings, 341-357.

15) Bratu, S. and C. Barnhart (2004). "An Analysis of Passenger Delays Using Flight Operations and Passenger Booking Data". Submitted to Air Traffic Control Quarterly.

16) Bureau of Transportation Statistics website (2003): http://www.bts.gov/oai/

17) Chebalov S. and D. Klabjan (2002). "Robust Airline Crew Scheduling: Move-up Crews". Proceedings of the 2002 NSF Design, Service, and Manufacturing Grantees and Research Conference.

18) Chu H., E. Gelman and E. Johnson (1997). "Solving Large Scale Crew Scheduling Problems". European Journal of Operational Research, 97: 260-268.

19) Clarke, L., C. Hane, E. Johnson and G. Nemhauser (1996). "Maintenance and Crew Considerations in Fleet Assignment". Transportation Science, 30:249-260.

20) Clarke L., E. Johnson, G. Nemhauser and Z. Zhu (1996). "The Aircraft Rotation Problem". Annals of Operations Research, 69: 33-46.

21) Clarke, M. and B. Smith (2000). "The Impact of Operations Research on the Evolution of the Airline Industry: A Review of the Airline Planning Process". Research paper, Sabre Inc.

22) Cohn, A. and C. Barnhart (2003). "Improving Crew Scheduling by Incorporating Key Maintenance Routing Decisions". Operations Research, 51: 387-396.

23) Cordeau J., G. Stojkociv, F. Soumis and J. Desrosiers (2000). "Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling". Technical Report G-2000-37, GERAD, Ecole Polytechnique de Montreal, Quebec, Canada.

24) Course 1.206 (Airline Schedule Planning) class handouts.

25) Course 16.781 (Planning and Design of Airport Systems) class handouts

26) Daskin M. S. and N. D. Panayotopoulos (1989). "A Lagrangian Relaxation Approach to Assigning Aircraft to Routes in Hub and Spoke Networks". Transportation Science, 23: 91-99.

27) De Neufville R. and A. Odoni (2003). "Airport Systems", McGraw-Hill

28) Desaulniers J., J. Desrosiers, Y. Dumas, S. Marc, B. Rioux, M. Solomon and F. Soumis (1997). "Crew Pairing at Air France". European Journal of Operational Research, 97: 245-259.

29) Feo, T. A. and J. F. Bard (1989). "Flight Scheduling and Maintenance Base Planning". Management Science, 35: 1415-1432.

30) Gopalan R. and K. Talluri (1998). "The Aircraft Maintenance Routing Problem". Operations Research, 46: 260-271.

31) Gu, Z., E. L. Johnson, G. L. Nemhauser and Y. Wang (1994). "Some Properties of the Fleet Assignment Problem". Operations Research Letters, 15, 59-71.

32) Hane C., C. Barnhart, E. Johnson, R. Marsten, G. Nemhauser and G. Sigismondi (1995). "The Fleet Assignment Problem: Solving a Large-Scale Integer Program". Mathematical Programming, 70: 211-232.

33) Hoffman K. and M. Padberg (1993). "Solving Airline Crew Scheduling Problems by Branch-and-Cut". Management Science, 39: 657-682.

34) Kabbani N. M. and B. W. Patty (1992). "Aircraft Routing at American Airlines". Proceedings of the Thirty-Second Annual Symposium of AGIFORS. Budapest, Hungary.

35) Kang L. and J.P. Clarke (2003). "Degradable Airline Scheduling". Working paper, Massachusetts Institute of Technology.

36) Klabjan D., E. Johnson and G. Nemhauser (1999). "Solving Large Airline Crew Scheduling Problems: Random Pairing Generation and Strong Branching". Technical Report TLI/LEC-99-11, Georgia Institute of Technology.

37) Klabjan D., E. Johnson, G. Nemhauser, E. Gelman and S. Ramaswamy (2002). "Airline Crew Scheduling with Time Windows and Plane Count Constraints". Transportation Science, 36: 337-348.

38) Klabjan D., K. Schwan (1999). "Airline Crew Pairing Generation in Parallel". Technical Report TLI/LEC-99, Georgia Institute of Technology.

39) Lan S., J. P. Clarke and C. Barnhart (2003). "Robust Aircraft Routing," submitted to Transportation Science.

40) Lohatepanont M. and C. Barnhart (2004). "Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment". Transportation Science, 38: 19-32.

41) Marsten R.E., R. Subramanian and L. Gibbons (1996). "Junior Analyst Extraordinare (JANE)", AGIFORS proceedings, 247-259.

42) Rexing B., C. Barnhart, T. Kniker, A. Jarrah and N. Krishnamurthy (2000)."Air-line Fleet Assignment with Time Windows". Transportation Science, 34: 1-20.

43) Rosenberger J. M., E. Johnson and G. L. Nemhauser (2001). "A Robust Fleet Assignment Model with Hub Isolation and Short Cycles". White papers and reports, the Logistics Institute, Georgia Institute of Technology.
http://www.isye.gatech.edu/research/files/tli0112.pdf

44) Schaefer A., E. Johnson, A. Kleywegt and G. Nemhauser (2001). "Airline Crew Scheduling Under Uncertainty". White papers and reports, the Logistics Institute, Georgia Institute of Technology.
http://www.isye.gatech.edu/research/files/tli0101.pdf

45) Vance P., C. Barnhart, E. Johnson and G. Nemhauser (1997). "Airline Crew Scheduling: A New Formulation and Decomposition Algorithm". Operations Research, 45: 188-200.

46) Yen J. and J. Birge (2001). "A Stochastic Programming Approach to the Airline Crew Scheduling Problem". Submitted to the Operations Research.

Appendix I: Passenger Delay Calculator (PDC)

What is the Passenger Delay Calculator?

Unfortunately, airlines do not record or compute their passenger delay statistics. They only publish their flight leg performance data. The Passenger Delay Calculator, or PDC, is an algorithm developed by Bratu and Barnhart (2004) [15], to estimate the amount of passenger delay on each itinerary, given actual disruptions occurring during operations.

The PDC algorithm first identifies disrupted passengers on local and connecting itineraries, then sorts them (usually they are sorted according to their time of disruption, that is, passengers who are disrupted first are recovered first), and finally searches for the "best recovery option" for each group of disrupted passengers. A recovery option is a feasible set of flights originating from the point of disruption and ending at the passenger's desired destination. The best recovery option then, allows the disrupted passenger to arrive at his/her desired destination as early as possible.

PDC Inputs/Outputs

The PDC algorithm uses the following data as input:

1. Flight data: Containing information about each flight's number, origin airport, destination airport, planned origin time, actual origin time, planned destination time, actual destination time, aircraft capacity, flight cancellation index and aircraft tail number.

   Values of actual origin and destination times as well as each flight's cancellation index can be read from a file (such as the ASQP data base) or can be generated inside the program's body by a random procedure.
2. Itinerary (passenger type) information: Containing information about each itinerary's number, origin, final destination, demand (booked passengers), first leg flight number

and second leg flight number. A second flight leg number, equal to zero indicates a local passenger.

3. Minimum Connection Times: Containing the minimum connection times at each airport location.

The algorithm then uses the above data to calculate the amount of passenger delay for each passenger type, the number of disrupted passengers recovered the same day, the number of overnight passengers, the average delay of local, through and connecting passengers, etc.

PDC Assumptions

The PDC algorithm is based on a set of assumptions, including:

- *Instantaneous information*: We assume that a disrupted passenger is recovered instantaneously on the best recovery itinerary as long as the time interval between the first flight leg in the recovery itinerary and the disruption time is greater than or equal to the minimum connection time (it takes at least MCT time units for a disrupted passenger to walk from the gate of arrival to the departure check-in point). It is assumed that passengers never miss this recovery option. This is not always the case in practice due to lack of information about all possible recovery options or because of poor judgment or carelessness of customer service representatives.

- *Perfect information:* We assume that when the airline recovers disrupted passengers, it has complete information about the remaining operations in that day. That is, it knows with certainty delays and cancellations that will be encountered that day. No disrupted passenger will therefore be disrupted again. Hence the values of passenger delay are underestimated in PDC.

- *Constant aircraft turn time:* We assume that aircraft turn time does not increase with load factor to reflect longer passenger embarking and disembarking times.

- *Complete knowledge of seat availability:* We assume that at the time of recovery, the airline has complete knowledge of the available empty seats on each flight. This is again not always the case in practice. The number of people who show up for a flight is stochastic.

- *Single airline:* We assume that disrupted passengers can only be recovered by the airline on which they are originally booked. If we have the bookings and seat availability data of other airlines, however, PDC has the capability to consider inter-airline passenger re-accommodations. Nonetheless, the single airline assumption is realistic in that it is typically costly for an airline to send disrupted passengers to another airline for recovery, and, moreover, if the capacity of an airport drops because of bad weather and airlines are forced to cancel some of their flights, other airlines will also face a lack of seat availability.

- *Service priority:* We assume that non-disrupted passengers have priority over disrupted passengers. Even if a disrupted passenger has to wait a long time in the airport until he/she can be recovered, we assume that the disrupted passenger is not assigned to a seat booked by a non-disrupted passenger.

- *Rebooking priority:* We assume that disrupted passengers will be recovered according to their time of disruption, that is, on a First-Come-First-Served (FCFS) basis. This rebooking policy does not necessarily minimize total passenger delay in the network or maximize the airline's revenue. It is, however, easy to implement and is sensible. We can easily change the PDC algorithm's rebooking policy to match other practices, such as protecting passengers by fare paid or frequent flyer status and processing passengers from highest-to-lowest priority order.