Long Haul Fleet Assignment: Models, Methods and Applications

by

Rajiv Chellappa Lochan
B.Tech., Indian Institute of Technology,
Madras (1993)

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Transportation

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1995

Author ....
Department of Civil and Environmental Engineering
May 12, 1995

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

Accepted by ....
Joseph M. Sussman
Chairman, Departmental Committee on Graduate Studies

Long Haul Fleet Assignment: Models, Methods and Applications

by

Rajiv Chellappa Lochan

B.Tech., Indian Institute of Technology,

Madras (1993)

Submitted to the Department of Civil and Environmental Engineering
on May 12, 1995, in partial fulfillment of the
requirements for the degree of
Master of Science in Transportation

Abstract

The fleet assignment problem is an integral part of the airline planning process. Given a schedule of international flights and certain maintenance criteria required by the Federal Aviation Administration (FAA), the objective of the long haul fleet assignment problem is to assign aircraft of different fleet types to different flights such that operating costs are minimized. We develop a new formulation for this problem with the decision variables modeled as sets of flights originating and terminating at maintenance stations. This variable definition enables explicit incorporation of aircraft maintenance constraints in the model. Maintenance requirements need to be considered explicitly in the international problem since opportunities for maintenance are fewer than in domestic operations.

Using this variable definition, the long haul fleet assignment problem is formulated as an integer multi-commodity flow problem with side constraints, defined on a time-line network. The model is solved using a branch and bound procedure in which the bounds are provided by solving a linear program at each node of the branch and bound tree. The definition of the decision variable as a string of flights precludes explicit enumeration of the constraint matrix since it is possible that billions of strings exist. Hence the root node linear programming relaxation of the problem is solved using column generation techniques. Using the data and international schedule of a long-haul airline, near-optimal solutions have been determined. Significant savings in operating costs are achieved by our solution procedure.

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

Acknowledgements

I would like to thank Prof. Cynthia Barnhart for being not only an excellent research advisor but also a great friend, philosopher and guide. She has been a remarkable source of inspiration & encouragement and most understanding, especially at times of frustration.

I wish to express my sincere gratitude to Dr. Pam Vance and Dr. Rina Schneur for being great sources of ideas and for making the many brainstorming sessions fun.

I greatly appreciate the efforts of Rajesh Shenoi, Dr. Dinesh Gopinath, K.Ashok and Daeki Kim who besides being great friends gave me useful ideas and molded my research direction.

I would like to place on record my sincere appreciation of the Large Scale Optimization Group, whose meetings have been excellent fora for exchange of thoughts and research ideas. The pizzas, donuts and bagels are unforgettable!

I wish to thank Chris Caplice, Bill Cowart, Jeff Sriver, Nicola Shaw, Prodyut Dutt, Gordon Coor and David Cuthbert for being wonderful office mates.

I would also like to thank Adriana Bernardino, Amalia Polydoropoulou, Andras Farkas, Jiang Chang, Owen Chen, Yan Dong, Hari Subramaniam, Sudhir Anandarao and all the other folks at the Center for making life easy and fun. Thanks a lot.

Special thanks to Manja, Roof, Syed, Daadh, and the rest of the boys for being such great people. You guys are the best!

Thanks are also due to Nagaraja Harshadeep, Dr.Gokul P.Krishnan, Anil Chakravarthy, Ravi Sundaram, Sankar Sunder, V.T. Srikar, Giri Iyengar, Ravi Kane, Manish Keshive, Abhijit Sarkar, Srihari Balasubramanian, Manish Kothari, Ganti Suryaprakash, Sonu Aggarwal, Amit Dhadwal and the rest of the "desi gang" for being such great people.

Finally, I would like to thank my parents who have always believed in my abilities and whose encouragement is second to none. Their support, love and blessings are most precious and invaluable to me. I also wish to thank Sanjiv, Rohini and Preeti for their support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Airline Problem: An Overview

The overall problem of operating and managing an airline is a very complicated procedure and consists of handling a variety of related issues, each of which is complex. There are a huge number of decisions to be made and a multiplicity of objectives for the different parts of the overall problem. Consequently, the airlines adopt a multi-stage planning process, shown schematically in Figure 1-1. The overall airline problem consists of:

Analysis of Market demand: Forecasts of expected traffic are made for every pair of cities the airline serves. Demand in different origin-destination (O-D) markets is different not only during different times of the year but also during different days of the week. While methods to estimate this demand are available (refer Ben-Akiva and Lerman [6] and Simpson and Belobaba [31]), airlines sometimes estimate the demands by extrapolating historical trends and making amendments based on certain assumptions or hypotheses (Elce [21]).

Flight schedule preparation: Based on the above forecasts and other considerations (such as level of service desired, extent of competition and so forth), a schedule of flights is prepared with times of departure and arrival. These flights depend on allowable routes that can be flown under the "route authority" or bilateral agreements. Depending on the demand forecasted and the size of the airline's operations, different

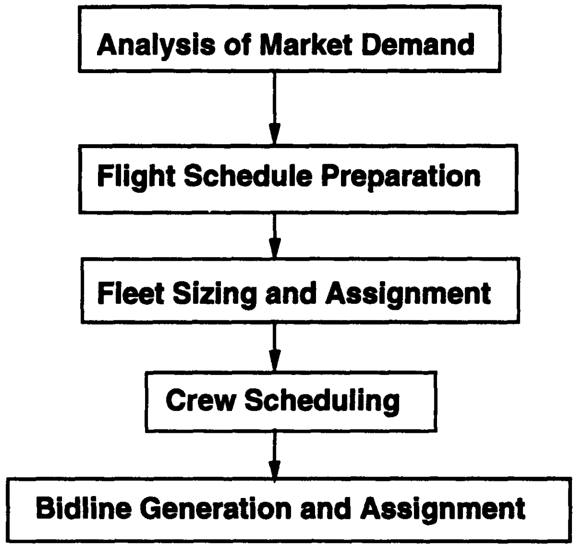

Figure 1-1: Overall View of the Airline Planning Process - A Schematic Diagram

frequencies of flights are flown by the airline in different O-D markets. A detailed analysis of supply of scheduled air transportation services can be found in Simpson and Belobaba [31].

Fleet sizing and assignment: The flight schedule developed governs the minimum size of fleet required. Assignment of equipment to flights in such a way as to match demand with aircraft size is the essence of fleet assignment. This problem is discussed in detail in later sections. Another related problem is that of aircraft rotation. Once the fleeting has been carried out, an optimal routing of each aircraft needs to be determined, or in other words, a routing for each aircraft consisting of a sequence of flight legs and ground connections so as to minimize operating costs or maximize revenues and satisfy maintenance requirements.

Crew scheduling: The next step in the process is to find the optimal allocation of crews to the scheduled flights. It is assumed that each crew is assigned to exactly one fleet type. It is required that each flight is covered by exactly one crew, collective bargaining agreements are satisfied and total crew costs are minimized. This problem has been solved with a variety of techniques, using heuristics and optimization procedures. The interested reader is referred to Barnhart et al. [5], Shenoi [30], Baker et al. [4] and Desrosiers et al. [19].

Bid-line generation: The final step in the process is to generate bidlines which are sets of crew pairings * flown by one crew during the entire planning horizon. The crew bid for the different sets of pairings and the assignment of crews is based on seniority.

Clearly each subproblem influences and interacts with another. For instance, fleet assignment affects crew pairings and in turn the bidline generation. The airline planning process is, therefore, an involved procedure. Each subproblem is non-trivial to solve and this precludes the solution of the overall airline planning problem in a single formulation. Each subproblem is an important piece of the overall problem and is essential to the smooth operation of the airline business.

This thesis undertakes the study of optimization of the fleet assignment problem for long haul (international) carriers. A new model has been developed and special techniques have been implemented to obtain near-optimal solutions for an international carrier.

1.2 Fleet Assignment Problem Definition

The objective of the fleet assignment model, given a schedule and a set of aircraft, is to determine an allocation of equipment type to each leg of the schedule. Specifically, the long haul (international) fleet assignment problem (LHFAP) is studied here with the objective of minimizing total costs. The total costs are the sum of the actual cost incurred in flying a flight leg and the cost of spilled passengers. Spill cost is defined as the revenue lost in turning away passengers as a consequence of allocation of an aircraft with capacity less than the demand for a particular flight leg. This cost can be computed for each flight-fleet pair. In other words, minimization of

*A crew pairing may be defined as a set of duty periods served by the crew that begins and ends at the same crew base.

costs can be achieved by optimally matching aircraft sizes with market demands. Flying an aircraft with capacity less than demand would lower revenue, increase spill and therefore increase operating costs. A more detailed discussion of the objective function costs is presented in Section 2.2.

The constraints of the LHFAP that must be satisfied are as follows:

1. Each flight leg in the schedule must be *covered* or flown by **exactly one fleet type**. (Unprofitable flight legs cannot be eliminated and fractional fleet assignments are inadmissible.)

2. Flow of aircraft by fleet type must be *balanced (or conserved)* with total flights flown by a particular fleet into a station equalling the total flights flown by that fleet out of the station. These constraints force the flow to be a circulation and prevent grounding of any aircraft.

3. The total number of aircraft of each fleet type used may not exceed the total number of aircraft of that fleet type available for the airline's long haul operations.

To summarize, the formulation for solving the long haul fleet assignment is:

Minimize [Operation Costs]

s.t.

Flight Cover Constraints

Flow Conservation Constraints

Plane Count Constraints

Plane Integrality Constraints

The LHFAP is described in detail and is contrasted with the domestic fleet assignment problem in Chapter 2.

1.3 Motivation

The airline industry is highly capital-intensive and has a low-profit environment, the combination of which results in a poor profitability position for the entire industry.

An empty seat on a flight is an instance of lost revenue. "The airline seat is the most perishable commodity in the world" state Subramaniam, et al. [32]. It is the objective of the airline to carry as many people as possible and keep its fleet in the air as long as possible.

It is true that as a consequence of airline deregulation, air travelers have saved millions of dollars. However the airlines are in a major financial crisis and all airlines are reconsidering their strategies and undertaking cost-cutting, revenue-enhancing measures. Predatory pricing, price fixing, code-sharing †, yield management, and so forth are some of the sophisticated tactics of the airlines to capture more passengers in all O-D markets and hence improve overall market share.

Optimization tools in Operations Research provide equally sophisticated techniques to minimize operating costs and maximize revenues and profits and these are being increasingly adopted by airlines. Most of the major airlines have dedicated departments/divisions in Operations Research and are increasingly investing in research and development ([27]).

Table 1.1 gives the percentage breakdown of operating costs for the major airlines' operations in the US for the years 1978, 1985, 1993 and 1994 ‡. It is clear that flying operation costs (in terms of fuel costs, crew costs less depreciation and insurance costs) and maintenance costs (in terms of flight maintenance and aircraft servicing) account for about 45% of the total operating costs. In order to enhance profitability (profits are defined as revenues less operating costs), airlines aim at increasing revenues and lowering costs. From the fleet assignment problem standpoint, a better matching of aircraft sizes to passenger demands increases revenue. A consequence is more efficient utilization of the fleet (and hence lowering of direct flying operation costs), an improved maintenance schedule (or lowered maintenance costs) and hence a lowering of total operating costs. In fact, improved fleet assigment for their domestic operations at Delta Air Lines Inc., resulted in savings of upto $100 million per

† Code-sharing refers to the concept of different airlines agreeing to share the same code for certain flights in order that these flights are listed earlier in the computerized reservation system screen

‡ Source: Schedule 41 reports of financial statistics submitted by carriers to the Department of Transportation

Table 1.1: Airline Operations of Major Carriers - Breakdown of Operating Costs (%)

Category	1978	1985	1993	1994
Flying Operations	34.72	34.73	28.73	27.69
Maintenance	12.86	9.76	10.94	10.95
Passenger Service	9.76	9.61	9.08	9.00
Aircraft and Traffic Servicing	18.12	15.29	15.98	16.17
Promotion and Sales	11.97	16.58	17.63	17.09
General and Administrative	4.51	5.47	5.08	5.22
Depreciation and Amortization	5.67	6.35	5.62	5.84
Transport Related	2.39	2.20	6.95	8.03
Total Operating Costs	100.00	100.00	100.00	100.00

year (Subramaniam et al. [32]). To our knowledge, the solution of the long haul fleet assignment problem has not been reported in the literature to date, even though, as in the domestic case, a fleet assignment model for international operations could have a significant impace in terms of cost savings.

Existing models and solution procedures for fleet assignment (Gopalan and Talluri [24], Hane et al. [25]) do not include maintenance considerations because their explicit incorporation:

1. is unnecessary (feasible aircraft rotations satisfying maintenance can be generated); and

2. destroys tractability and computational efficiency.

The domestic fleet assignment problem is, therefore, solved without any maintenance considerations. Solution of the international fleet assignment problem without the maintenance constraints, however, often provides a solution that is infeasible in that maintenance requirements cannot be satisfied. The reason for this is primarily due to differences in the structure (in terms of the network) and the scale (in terms of number of flights offered and aircraft utilized) of domestic and international operations of airlines (explained in greater detail in Section 2.1.1). Therefore, we provide a formulation in Section 2.2 and a solution methodology in Chapter 3 that allow maintenance considerations to be explicitly incorporated in the solution of the international fleet assignment problem.

1.4 Contributions of thesis

The main contributions of this thesis are:

1. Formulation of a new model for the long haul fleet assignment problem with maintenance considerations explicitly incorporated.

2. A computationally efficient implementation of the new formulation.

3. Presentation of computational experience using real-world data of a long haul airline in the form a case study (Chapter 4) and an analysis of results. This includes an optimal fleeting for the airline's international operations and a significant savings in operating costs of compared to those of the current fleeting flown by the carrier.

1.5 Outline of thesis

The long haul fleet assignment problem is formally defined and the integer programming formulation is presented in Chapter 2. A comparison of the long haul formulation with that for domestic operations is presented and the relative merits discussed. Chapter 3 presents alternative solution methodologies and outlines the solution strategy adopted. In Chapter 4, a case study involving the international schedule and data of a long-haul carrier is presented. Implementation details, computational results and relevant experience are presented. Future research and scope are discussed in Chapter 5.

Chapter 2

Problem Definition and Formulation

In this chapter, a thorough definition of the long haul fleet assignment problem is presented. In particular, the LHFAP is contrasted with the domestic fleet assignment problem to motivate the need for a new model. A new mathematical model is then presented.

2.1 Problem Definition

The problem studied here is the long haul fleet assignment problem (LHFAP). Given an international schedule of flights and certain maintenance criteria required by the Federal Aviation Administration (FAA) * and Feo and Bard [22], the LHFAP requires the determination of which fleet type should fly each flight leg in the schedule such that operating costs are minimized or operating revenues/profits are maximized. In this thesis, specifically, operating costs are minimized.

The FAA requires that maintenance (A and B checks) be performed every 65

*The FAA requires different levels of maintenance checks. The checks are called *A*, *B*, *C* and *D* and vary in frequency and duration. These range from a visual inspection of all major systems such as landing gear, engines and control surfaces, every 65 flying hours (Check *A*) to a *balance check* which is a more expensive and extensive operation but less frequent. For more details of the maintenance requirements and industry practice, the reader is referred to Talluri and Gopalan [24].

flying hours. Industry practices are more stringent and require maintenance every 40 to 45 flying hours or (a more conservative) every three to four elapsed days. (Talluri and Gopalan [24]). Thus the maintenance is scheduled either based on a flying time criterion or on an elapsed time criterion.

2.1.1 Domestic vs International Fleet Assignment: A Contrast

One difference between long haul or international and domestic fleet assignment is the planning horizon. In domestic operations, flights are often repeated each day of the week and a daily planning horizon is appropriate. However, in international operations, flights do not typically repeat daily and a weekly planning horizon is required.

Domestic operations of an airline are characterized by a hub-and-spoke network. Technically, a hub is defined as a city which accounts for at least a certain percentage of enplanments (Phillips [29]). A major hub is a city that accounts for at least one percent of total domestic enplanements; medium hubs are those with 0.25% to 0.99% of domestic enplanements; small hubs have 0.05% to 0.24%; and non-hubs have less than 0.05% domestic enplanements. Phillips [29] defines a "major hub" as an airport that accounts for 10% or more of total domestic passengers enplaned, or 10% or more of total domestic departures, for a *particular air carrier* [†]. Markets are served either out of the hub or through a hub. A consequence of hub-and-spoke operations is a reduction in direct service between smaller cities with lower demand. Instead, connecting service is offered through one or more hubs (Kanafani and Ghobrial [26]). Passengers are, therefore, consolidated in links (or spokes) to and from the hub and this allows the airline to exploit economies of aircraft size. Also airlines and passengers are in a position to take advantage of economies of increased schedule frequency. This provides the motivation for the airlines to resort to a hub-and-spoke network.

[†]For example, American enplanes 61% of the passengers at its major hub, Dallas/Fort Worth while Delta enplaned about 90% of the passengers at Atlanta, Delta's major hub

Carriers usually tend to be dominant at their hubs in terms of gates, terminal space, groundside constraints, airport landing slots, etc. The airlines usually maintain facilities for maintenance of their aircraft at their hubs (Phillips [29]). This fact coupled with the daily planning horizon for the domestic operations has an interesting consequence. Typically, there exists a period of inactivity during the day at a hub when routine maintenance is performed. In contrast, international operations involve point-to-point service resulting in a more sparse network with fewer opportunities for maintenance. There is one further implication of the hub-and-spoke operations. The increased activity at the hub (in other words, greater number of flights from or through the hub) implies that it is possible to "swap" aircraft to fly different flights after the fleet assignment has been done, while still preserving the validity of the fleet assignment. An aircraft requiring maintenance can therefore be either swapped with another aircraft and be maintained at the hub or can fly another departing flight out of the hub in order to reach a maintenance facility. This facilitates adherence to the FAA's maintenance requirements. Such opportunities are limited in the long haul case. For a detailed analysis of the swapping applications in the domestic fleet assignment problem and an efficient algorithm for the same, the interested reader is referred to Talluri [34]. For the domestic problem, therefore, experience shows that it is possible to find a feasible (with respect to maintenance) aircraft routing given a solution to a fleet assignment model that does not explicitly incorporate maintenance constraints.

To illustrate the effect of the hub-and-spoke nature of domestic operations, consider the airline whose international schedule and data are used in the case study in Chapter 4.

The airline's domestic operations involve 2500 flights *a day* serving about 150 cities with 475 aircraft of 11 fleet types and 5 major hubs. The international operations, on the other hand, involve about 1200 flights *a week* serving 55 cities with 75 aircraft of 11 fleet types and 8 maintenance bases. The fact that most domestic flights arrive at or pass through hubs implies that maintenance opportunities are greater in the domestic problem than in the international case. Also the possibility of swapping of aircraft given the larger number of aircraft and given the hubs in domestic operations, makes fleet assignment without explicitly considering maintenance constraints possible. Such swapping opportunities are limited in international operations.

2.1.2 Solving the Domestic Fleet Assignment Problem: State-of-the-art

The fleet assignment problem has been described as one of the largest and most difficult problems in the airline industry (Subramaniam et al. [32]). Considerable research has been done on the domestic fleet assignment problem. The interested reader is referred to Abara [1], Dillon et al. [20], Hane et al. [25], Daskin and Panayotopoulos [14], Subramaniam et al. [32]. Different operations research-based techniques have been applied, with different decision variable definitions and different objectives. For example, Daskin and Panayotopoulos [14] applied a Lagrangian Relaxation approach to fleet assignment, others have used classical mixed integer programming techniques to solve the problem. Hane et al. [25] defined the decision variable as a particular flight flown by an aircraft of a particular fleet type with an objective of minimizing total costs of operation, while Abara [1] defined the decision variable as a feasible turn and aircraft combination with an objective of maximizing the operating profits (essentially revenues less operating costs). Both Hane et al. [25] and Abara [1] solved the domestic fleet assignment problem using mixed integer programming techniques available in commercial optimization software.

A common attribute of all models is that maintenance constraints are not explicitly incorporated for the reasons discussed in the previous section. However, all the standard constraints of flight coverage, flow balance and aircraft count discussed in Section 1.2 have been included in all the models. A common feature of the models is that the formulations are flight based. Since the number of flights is finite, it is possible to explicitly enumerate the variables and hence the constraint matrix. Thereafter, the problem is solved as a mixed integer program using a standard optimizer, e.g. OSL (Hane et al. [25], Subramaniam et al. [32]), or using other techniques such as

Lagrangian Relaxation (Daskin and Panayatopoulos [14]).

Substantial savings have been reported by major US airlines for fleeting in their domestic operations using these models - American Airlines (Abara [1]), Delta Air Lines (Subramaniam et al. [32]) and USAir (Dillon et al. [20]). For example, a 1.4% improvement in operating margin has been reported by Abara [1] and a 100 million dollars per year savings in operating costs by Subramaniam et al. [32].

2.1.3 Solving the Long Haul Fleet Assignment Problem: State-of-the-art

To our knowledge, no published literature exists for the solution of the long haul fleet assignment problem specifically. This thesis presents the first published model for the *international fleet assignment problem.*

The discussion in Section 2.1.1 establishes that the domestic fleet assignment problem could be solved without explicitly incorporating maintenance constraints. While different hypotheses for the inapplicability of a similar solution strategy for the long haul fleet assignment problem have been stated, the need for a model different from the ones available for the domestic problem has not been established. In order establish this need, we tried to solve the long haul problem ignoring maintenance requirements. The decision variable was defined as x_{fk}, a flight f flown between by a fleet type k (as defined by Hane et al. [25]). Constraints of flight coverage, flow balance and aircraft count (see Section 1.2) were incorporated and the linear programming relaxation of the problem was solved to optimality using the CPLEX Release 3.0 optimization software. The results are summarized in Table 2.1. While fractional optimal solutions were obtained, these were found to be infeasible solutions in that maintenance requirements could not be satisfied ‡. In other words, ignoring maintenance requirements results in infeasible solutions to the long haul problem §

‡The optimal basis from this model was infeasible when used as a starting basis for the model with maintenance considerations included.

§Note that domestic models have "pseudo-maintenance constraints" included. That is, bounds are placed on the number of aircraft of different fleet types that need to be in maintenance stations each night. Inclusion of these constraints *might* result in feasible solutions for the long haul problem.

Table 2.1: Results of Long Haul Problem Solved without Maintenance Constraints

# Flights (# fleets)	# rows	# columns	# non-zeroes	Solution value	Time to optimality
126 (2)	516	892	2322	2960369.00	9s
408 (2)	1572	2794	7298	10439404.00	1m 24s
360 (4)	2472	4988	13000	8053843.00	1m 56s
536 (4)	3496	7244	18956	13539728.00	3m 22s
546 (7)	5831	12922	33873	15557153.50	8m 50s

and hence, a new model is required.

2.2 Problem Formulation

For the long haul fleet assignment problem, we propose a formulation that ensures conformity to maintenance regulations. Before presenting the mathematical formulation, we provide a description of the network representation of the domestic and long haul fleet assignment problems and present the variables and notation we adopted.

2.2.1 The Network

Before describing the network used in solving the LHFAP, an understanding of that used in solving the domestic problem (as defined by Hane et al. [25]) is appropriate.

The Domestic Fleet Assignment Network

In Hane et al. [25] and Subramaniam et al. [32], the domestic fleet assignment problem is modeled on a time-expanded multi-commodity flow network, spanning one day. This network consists of:

- *Nodes* which represent flight departures or arrivals at a station at a given point in time. Each departure node is associated with the departure location and time of a flight, while each arrival node is associated with a flight arrival location

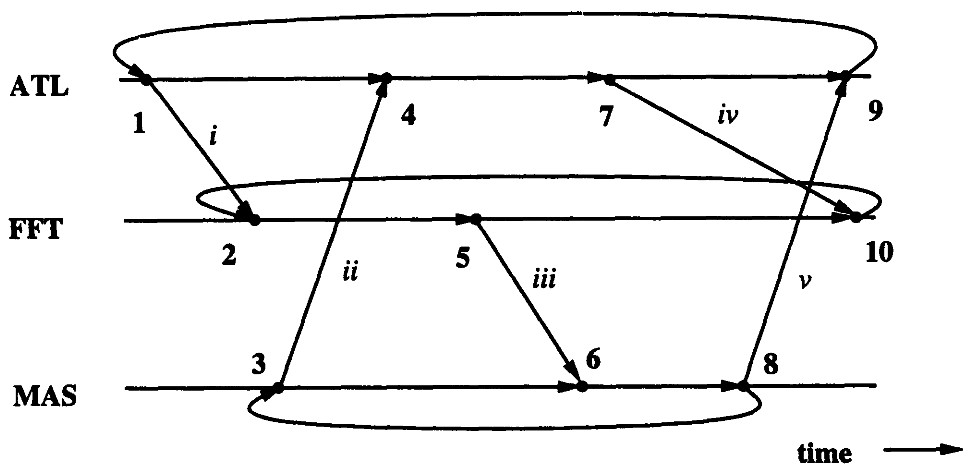

Figure 2-1: The Network for the Daily Problem - Flight-based.

and its arrival time increased by refuelling and baggage handling time. Nodes are numbered chronologically in increasing order of time.

- *Flight arcs* which correspond to flight legs of the schedule. Flight arcs are numbered chronologically in increasing order of departure time.

- *Ground arcs* which permit an aircraft to sit on the ground, either to make a connection or be maintained. Essentially ground arcs connect different nodes at a given station. *Overnight arcs* are also ground arcs but these allow aircraft to sit on the ground overnight at a station. These arcs are also called wraparound arcs since they wraparound the network, given that the planning horizon is one day.

Example: *Consider the network in Figure 2-1, with 3 stations and 5 flights. Nodes 2, 5 and 10 represent Frankfurt at different points of time. Arc i represents a flight leg from Atlanta to Frankfurt and arc v represents one from Madras to Atlanta at a later point in time. Between an aircraft arrival at a node and the next departure*

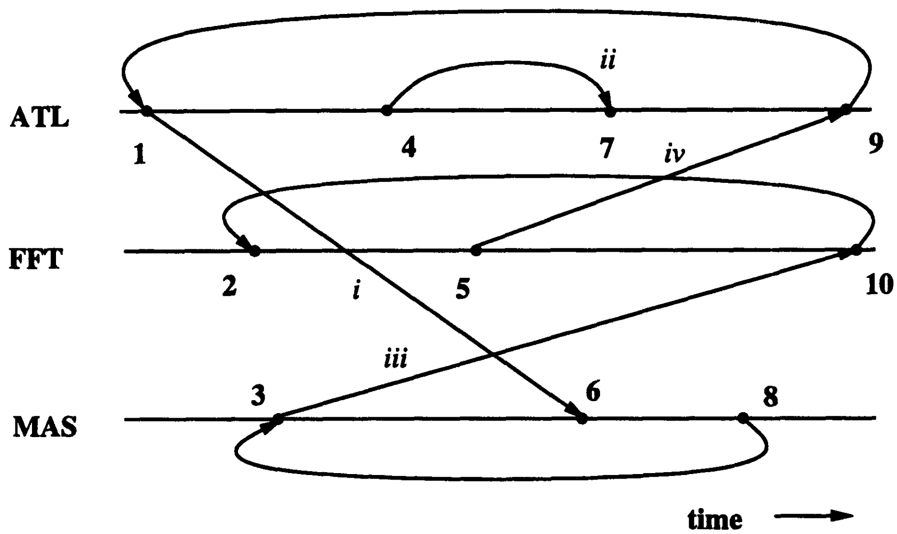

Figure 2-2: The Network for the Weekly Problem - String-based.

out of the city, there is a ground arc which represents the plane sitting on the ground. The arc between nodes 4 and 7 at Atlanta is an example of a ground arc. Overnight arcs or wraparound arcs are between nodes 8 & 3, 9 & 1 and 10 &2.

The Long Haul Fleet Assignment Network

The LHFAP is also modeled on a similar topologically sorted time-line network. While the domestic fleet assignment network spans one day, the network for the LHFAP spans one week. Another distinction is that the nodes and arcs are interpreted differently. ¶ In the long haul network, a node represents *a maintenance station* and there is a departure node and an arrival node for each string. The departure nodes are associated with the location and time of departure of the first flight in the string. The

¶This distinction results as a consequence of the definition of strings. A string is defined as a set of flights flown by a fleet type originating and terminating at a maintenance station for that fleet type.

arrival nodes are associated with the location and arrival time plus refuelling/baggage handling time and *maintenance time* [||] at the arrival of the last flight in the string.

There is one arc in the LHFAP network for each string (recall that decision variables in the formulation are string-based). Given a schedule of flights, the number of strings may measure in millions, even billions. The following example illustrates this point.

Example: *Using the network in Figure 2-1 it is possible to construct a string-based network such as the one shown in Figure 2-2. Without loss of generality, assume all three cities are maintenance bases. To illustrate the enormity of the problem of constructing strings, consider flights i, iii and v in Figure 2-1. At least three legitimate strings (i, ii and iv in Figure 2-2) can be constructed (by combining flights i and iii, flights iii and v and flights i, iii and v respectively). As the size of the Figure 2-1 network increases, that is, as the number of flights in the schedule increases, the possible number of strings explodes. To illustrate, a major long haul airline flying about 1200 flights a week, maintaining about 30 to 40 maintenance bases and using about 10 different fleet types results in more than a few hundred billion of strings.*

Note that the network in Figure 2-2 is defined for each equipment type. Different fleet types have different maintenance stations and therefore, different networks.

2.2.2 Notation and Definition of Variables

Definition of Variables

The set of flights in the schedule is denoted by F, the maintenance stations (cities) in the schedule by M, the set of available fleets by K and the available number of aircraft of each fleet type by $N(k)$ for each $k \in K$. A string is defined as a set of flights flown by a particular fleet type, where the flights originate and terminate at a maintenance station for that fleet. Then, the set of all possible strings is denoted by J. A maintenance station for a particular fleet type at an arrival/departure time is

[||]In our application, maintenance time of 8.5 hours has been assumed.

represented as $\{mtk\}$, with $m \in M$ at a take-off/arrival time t for fleet type $k \in K$. The set of all nodes in the long haul network (as described in the previous section) is denoted by D.

The decision variable, x_{jk}, has a value 1 if the string $j \in J$ is flown by fleet $k \in K$, and 0 otherwise. This decision variable definition ensures that maintenance constraints are explicitly satisfied since only "legal" strings are considered, i.e., strings that satisfy FAA maintenance requirements regarding number of hours flown before maintenance.

Other Notation

The coefficients in the constraint matrix are defined as follows: a_{ijk} has a value of 1 if string $j \in J$ flown by fleet $k \in K$ covers flight $i \in F$, and 0 otherwise; b_{ljk} has a value 1 if string $j \in J$ flown by fleet $k \in K$ terminates at node $l \in D$, -1 if it starts at node $l \in D$ and 0 otherwise; d_{kj} has a value 1 if fleet $k \in K$ flies string $j \in J$ and 0 otherwise.

The objective coefficients, c_{jk} are the costs incurred if string j is flown by fleet k. String costs are merely the sum of the costs for each of the flights covered by the string. "Spill costs" are also included in these coefficients. Spill is defined as the positive part of the difference between projected demand for seats in a given pair of cities and the seating capacity of the aircraft. Since some of this spill is recaptured, the objective coefficient also includes the reduction in cost due to revenues from the recaptured passengers. As a result, the costs of flying a string varies by fleet type.

The objective function cost parameters play an important role in enforcing certain constraints that either cannot be easily formulated or are specific to one fleet type. For example, gate or noise restrictions might disallow aircraft of certain fleet types from landing at certain airports. These restrictions can be captured by imposing a higher cost of operation for the disallowed equipment type.

2.2.3 The Mathematical Model

The basic integer programming model for the LHFAP is as follows:

$$min \sum_{k \in K} \sum_{j \in J} c_{jk} x_{jk}$$

$$\sum_{k \in K} \sum_{j \in J} a_{ijk} x_{jk} = 1.0 \quad \forall\, i \in F \tag{2.1}$$

$$\sum_{j \in J} b_{ljk} x_{jk} = 0.0 \quad \forall\, l \in D, \forall\, k \in K \tag{2.2}$$

$$\sum_{j \in J} d_{jk} x_{jk} \leq N(k) \quad \forall\, k \in K \tag{2.3}$$

$$x_{jk} \in \{0, 1\} \tag{2.4}$$

The formulation essentially consists of three sets of constraints. The first set of constraints (2.1) is the “flight coverage” constraints. There is one constraint for each flight, requiring that each flight is covered by exactly one string. The second set of constraints (2.2) is the “flow balance” constraints, ensuring conservation of flow of aircraft of all fleet types. The third set of constraints (2.3) is the “fleet size” constraints, ensuring that the number of aircraft used of a particular equipment type does not exceed the available number of aircraft of that type. There is one such constraint for each equipment type. The set of constraints (2.4) ensures that each string is binary, guaranteeing that a string is either flown or not flown by a single aircraft type.

A typical constraint matrix for the long haul fleet assignment problem is shown in Figure 2-3. The total number of constraints in the constraint matrix is:

$$N_f + \sum_{k \in K} N_{mk} + N_k$$

where N_f represents the total number of flights in the schedule, N_{mk} represents the number of maintenance nodes for fleet type k and N_k represents the total number of fleet types maintained by the airline.

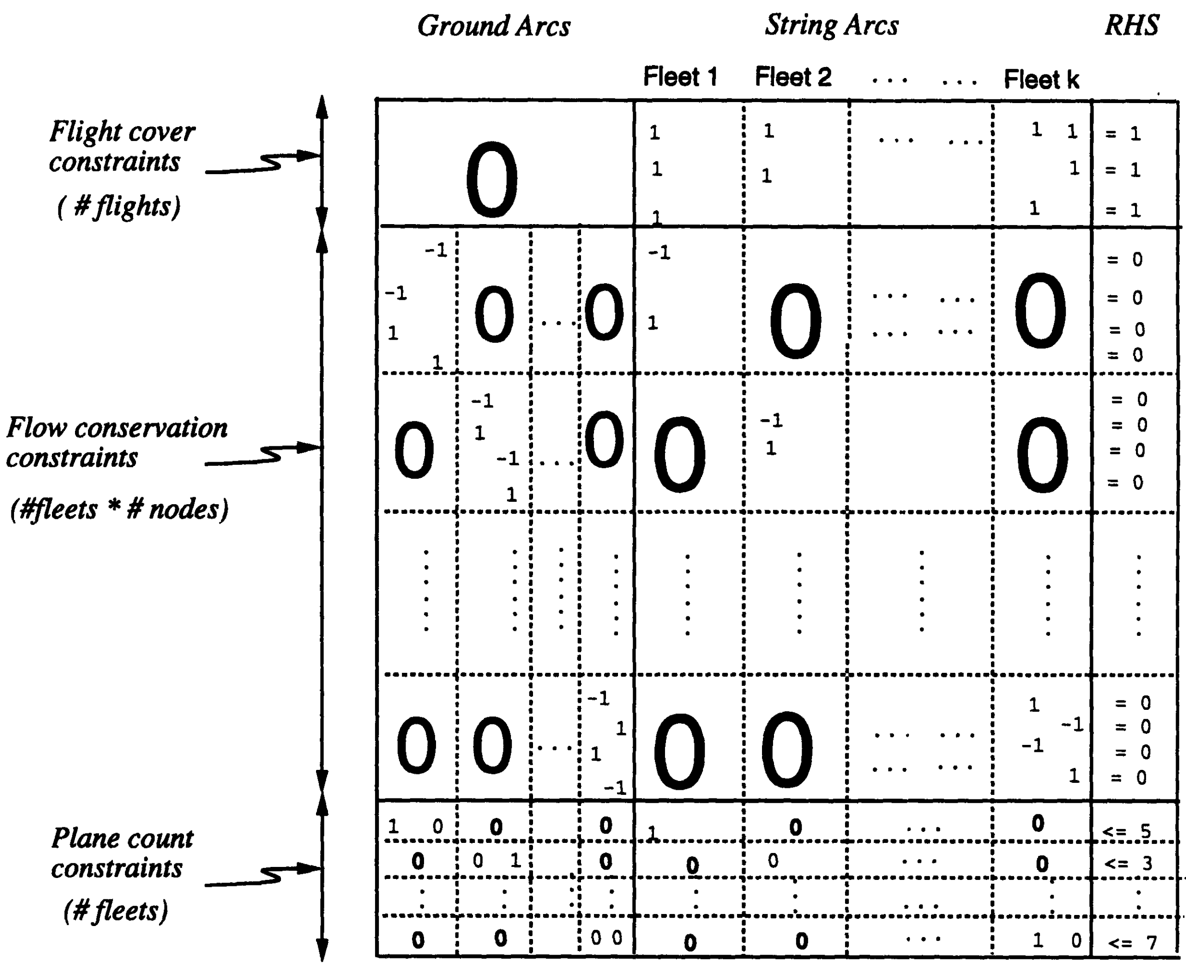

Figure 2-3: Typical Constraint Matrix

Example: *Consider a schedule of 1200 flights with 10 fleet types and 10 maintenance stations, resulting in about 500 maintenance nodes. The total number of constraints is 1200 + 500 * 10 + 10 = 6210 constraints and the number of variables in the matrix measures in the billions.*

Chapter 3

Solving the Long Haul Fleet Assignment Problem

This chapter outlines the overall solution strategy adopted in solving the LHFAP. Section 3.1 includes a description of the different methodologies used to solve integer programs and large linear programs containing numerous variables. These methodologies are fairly standard and have been extensively researched and applied to a variety of problems. Section 3.2 outlines the overall solution algorithm used to solve these large scale problems and the strategy we adopted to solve the LHFAP in particular.

Traditionally, the domestic fleet assignment problem has been solved using a branch and bound strategy, which involves solving a linear relaxation of the problem (LP) at each node of an enumeration tree. Solution of the LP is achieved using the simplex algorithm since explicit enumeration of the constraint matrix is possible. However, the huge number of variables (strings) in the LHFAP precludes direct solution of the LP. Instead column generation techniques are used to solve the LHFAP LP relaxation at each node of the branch and bound tree. These methods are discussed in the following sections.

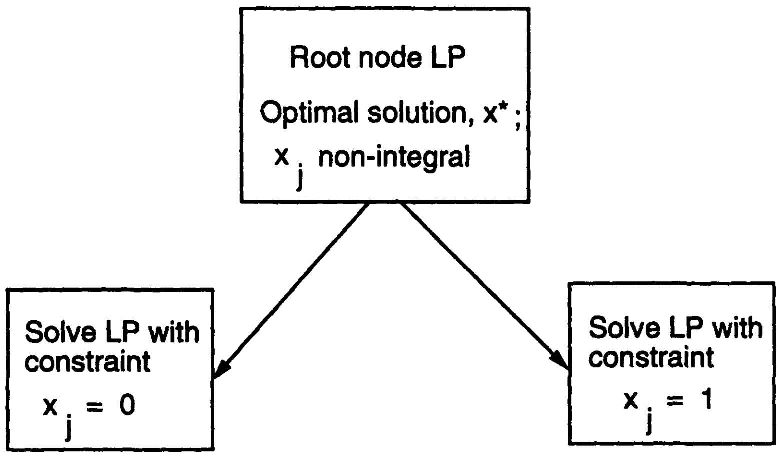

Figure 3-1: A Branching Rule for a Binary Decision Variable.

3.1 Solution Methodologies

A branch and bound procedure is used to determine the optimal integer solution to the LHFAP. This entails solution of many LP relaxations. Since explicit enumeration of the LHFAP constraint matrix is impractical and impossible due to limitations of memory, the LP relaxations are solved using column generation techniques.

3.1.1 Branch and Bound

Branch and bound is a "divide and conquer" algorithm used to solve an integer program to optimality. The rationale behind the strategy is to construct an *enumeration tree* as follows. At the root node of the tree, the linear programming relaxation of the problem is solved to optimality. If this solution is integral, then the original integer problem is solved. If however, there exists a variable $x_i{}^*$ in the optimal LP solution that is non-integral, then additional nodes in the tree can be constructed using a branching rule on the variable, x_i. A branching rule essentially divides the feasible IP solution space into mutually exclusive, collectively exhaustive regions, each of which corresponds to a node in the tree. The rule is similarly applied to all non-integral solutions at each node and the resulting tree is called the *branch and bound tree.*

There are a number of different branching rules that are possible and hence different bases on which to construct the tree. One such rule is shown in Figure 3-1. Consider a minimization problem with variables taking binary (i.e., 0 or 1) values only. The root node LP solution, x^*, has a non-integral component, x_j. The branching rule creates two branches (nodes) with an additional constraint over the root node LP. The left node has the constraint $x_j = 0$ while the right node has the constraint $x_j = 1$.

Some insight about the nomenclature, branch and bound, is quite relevant at this point. The LP solution at every node of the branch and bound enumeration tree has four possible outcomes:

1. The LP is infeasible, which implies that feasible solution space is empty and hence further branching from that node cannot result in an improved, feasible solution.

2. The optimal LP solution value is worse than the current best integer solution, which means further exploration at this node has no benefit. This is because the LP solution in a minimization problem is a lower bound on the IP solution.

3. The optimal LP solution value is better than the current best integer solution and the LP solution is integral. This means that a better integer solution has been found and further exploration is unnecessary.

4. The optimal LP solution value is better than the current best integer solution and the LP solution is non-integral. This means that further branching is required since an improved integral solution may be found.

The first three outcomes above provide a *bound* on the IP solution value and result in the branch-and-bound tree being fathomed *, while the fourth provides an opportunity to *branch* further in the enumeration tree. Also, the four outcomes are mutually exclusive and collectively exhaustive, that is, exactly one of them must occur. For a thorough treatment of branch and bound, the interested reader is referred to Bradley et al. [9].

3.1.2 Column Generation

Given the fact that the number of variables in the LHFAP is enormous, it is impractical to explicitly enumerate the constraint matrix. Consequently, it is impossible to solve the LP relaxation of the LHFAP at each node of the branch and bound tree directly using the Simplex algorithm. This motivates the use of *column generation*, also known as *Dantzig-Wolfe decomposition* [12], [13]. These techniques do not require explicit enumeration of the constraint matrix, but instead generate columns (or variables) "as needed" (Ahuja et al. [2]).

Applied to the LHFAP, column generation methods use a subset of the set of feasible strings as a starting basis, B. Associated with B, is a set of simplex multipliers, π, such that the reduced costs of the basic variables are zero. These simplex multipliers are then used to *price out* † the non-basic columns (or variables). Assuming a minimization formulation, if a variable has a negative reduced cost, it may improve the solution and it is therefore introduced into the constraint matrix. Adding variables to the constraint matrix is referred to as column generation. The generation of columns stops when optimality has been achieved, i.e., when there are no more variables with negative reduced cost. Column generation procedures work best when columns with negative reduced cost can be generated without examining all variables. Generating such columns or determining that none exist is called the *sub-problem*, while the solution of the linear program with a restricted subset of the variables is

*To *fathom* is defined as "to get to the bottom of or to understand thoroughly. In our context, fathoming may be more appropriately defined as "understood enough or already considered". Outcome 1 above is termed "fathoming by infeasibility", outcome 2, "fathoming by bounds" and outcome 3, "fathoming by integrality". (Bradley et al. [9])

†Pricing out essentially means computing the reduced costs of a string

called the *restricted master problem.* Column generation is a well researched area of large scale optimization, detailed descriptions of which can be found in Ahuja et al. [2] and Berstimas & Tsitsikilis [8].

3.1.3 Column Generation Subproblem

With column generation techniques, the repeated solution of the subproblem is often the bottleneck in the overall solution procedure. In this section, we show that the column generation subproblem of the LHFAP can be cast as a shortest path problem.

Shortest path problems find wide applicability in transportation, communication, inventory planning, DNA sequencing, and so forth. An extensive bibliography has been compiled by Deo and Pang [15] and a thorough theoretical treatment can be found in Ahuja et al. [2]. For discussion of the role of shortest path subproblems in column generation procedures, the reader is referred to Shenoi [30] and Desrochers and Soumis [16] [17].

Consider a variable in the formulation of the LHFAP discussed in Section 2.2. The reduced cost of the string represented by this variable, x_{ij}, can be written as:

$$\bar{c}_{jk} = c_{jk} - \sum_i \alpha_{ij} + \beta_{sjk} - \beta_{ejk} + \gamma_{jk} \tag{3.1}$$

where c_{jk} is the cost of string j for fleet k, α_{ij} is the dual variable associated with the cover constraint for flight i of string j (constraints 2.1), β_{sjk} is the dual associated with the flow balance constraint corresponding to the start node s of string j for fleet k (constraints 2.2), β_{ejk} is the dual variable associated with the flow balance constraint corresponding to the end node e of string j for fleet k (constraints 2.2) and γ_{jk} the dual variable associated with the fleet size constraint for string j and fleet k (constraints 2.3). Note that for a given node pair and fleet type, the terms β_{sk}, β_{ek} and γ_k are constant. Therefore, for each node pair it is possible to price out all strings between those nodes by running a shortest path procedure (on the network described in Figure 2-1) with modified arc costs:

$$c'_{jk} = c_{jk} - \sum_i \alpha_{ij} \tag{3.2}$$

Shortest path problems can be broadly classified into three categories depending on whether or not additional constraints or multiple optimality criteria exist. They are:

1. *A simple, unconstrained shortest path problem* with the objective of finding the cheapest (least cost) path between two nodes based only on the costs of the arcs in the network. Unconstrained shortest path problems can be solved using label setting or label correcting algorithms. The interested reader is referred to Ahuja et al. [2] for further details and complexity analyses of these algorithms.

2. *A multi-criterion shortest path problem* arises when multiple optimality criteria exist or multiple arc costs exist. For example, in some crew scheduling problems (Shenoi [30]), variable costs are defined as the maximum of three costs. To determine the variable with minimum cost, a shortest path procedure with three costs associated with each arc is solved. The multi-criterion shortest path problem can be solved using dynamic programming based methods such as those described by Desrochers and Soumis [16], [17].

3. *Constrained shortest path problems* are those where the shortest path between two nodes is required to satisfy certain additional constraints. Since the least cost path may not satisfy the additional constraints, all possible paths between the two nodes may have to be evaluated to find the optimal path. The constrained shortest path problem can be solved using methods such as those described by Desrochers and Soumis [16], [17].

For the LHFAP, the subproblem can be modeled either as multiple simple, unconstrained shortest path problems or as multiple constrained shortest path problems, depending on the specific maintenance scenarios. Recall from Section 2.1 that there are two maintenance scenarios; namely Maximum Flying Time and Maximum Elapsed Time. If the maximum flying time requirement that maintenance occur at least once in every 45 flying hour period is adopted, the shortest path subproblem is a constrained shortest path problem, i.e., the shortest path should satisfy the additional constraint that it not contain more than 45 hours of flying. Since all possible paths between two nodes in the LHFAP network may need to be evaluated in order to determine the shortest path satisfying the additional constraint, this procedure is computationally intensive. However, if the maximum elapsed time requirement that maintenance occur at least once every 3 (or 4) elapsed days is adopted, the shortest path subproblem is an easy, unconstrained shortest path problem. The shortest path procedure needs to consider only those strings that exist in a three (or four) day time window beginning at each source node.

Solution of the LHFAP subproblem using the appropriate shortest path procedure results in the identification of a single string for each node pair. The reduced cost of each string is computed using Equation 3.1. If the reduced cost of a string is negative, addition of this string to the constraint matrix may improve the solution. Any one of the following strategies can be adopted in adding strings to the constraint matrix:

- All strings with negative reduced cost can be added to the constraint matrix; or
- The string with the most negative reduced cost can be added to the constraint matrix; or
- A few (most) negative reduced cost strings can be added to the constraint matrix.

Using one of these strategies, a new master problem is constructed and solved. This iterative procedure of solving a restricted master problem and a subproblem is continued until no columns with negative reduced costs are found and hence the LP relaxation of the LHFAP is solved.

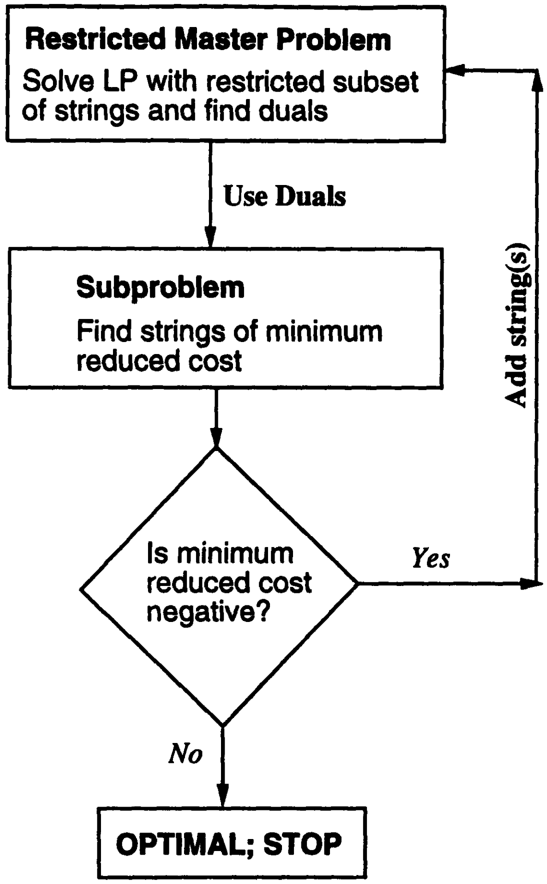

Figure 3-2: Schematic Representation of Solution of Root-node LP.

3.2 Solution Strategy

This section presents a column generation algorithm to solve the LP relaxation of the LHFAP and describes the method adopted to obtain integer solutions to the LHFAP.

3.2.1 Linear Programming Solution to the Long Haul Fleet Assignment Problem

A column generation algorithm shown in Figure 3-2 is used to solve the LHFAP LP relaxation. An iteration of the algorithm requires the solution of:

1. *The Restricted Master Problem*, i.e., the LP relaxation of the formulation discussed in Section 2.2.3 containing a subset of the total set of strings. The

restricted master problem is solved using the simplex algorithm and the optimal dual solution is used to compute modified arc costs (see equation 3.2) for each arc in the network; and

2. *The Subproblem*, i.e., the generation of negative reduced cost strings using one of the shortest path procedures discussed in Section 3.1.3.

The LHFAP LP relaxation is solved when no strings with negative reduced costs exist.

3.2.2 Integer Programming Solution to the Long Haul Fleet Assignment Problem

Integer solutions to the LHFAP can be obtained using the branch and bound techniques described in Section 3.1.1. However, the solution of LP relaxations at each node of the enumeration tree leads to practical difficulties, namely:

- Branch and bound is a computational and memory-intensive procedure even when the LP relaxations can be solved directly, i.e., without column generation; and
- Conventional branching strategies based on variable dichotomy complicate the subproblem structure making it impractical to solve the LP relaxation at each node of the branch and bound tree.

As a result, we adopt a simpler approach, a heuristic branch and bound approach, that is not guaranteed to result in an optimal solution. The heuristic solves only the root-node LP using column generation, with LP's at other nodes solved using only a fixed subset of columns. This subset of columns may be the exact set of columns generated in solving the root-node LP, or it may be a subset. Since additional columns are not generated within the branch and bound tree, some variables with negative reduced cost may be excluded and a suboptimal LHFAP solution may be determined. This approach has, however, been successfully adopted by Hane et al [25], Anbil et al [3], Barnhart et al [5], Desrochers and Soumis [18] and Shenoi [30] to solve similar problems.

A heuristic solution strategy described by Hane et al [25] uses the optimal LP solution. In the optimal solution to LP relaxation of the LHFAP, it is possible that values of some of the variables are close to zero or one. The idea of Hane, et al is to fix the values of these variables to zero and one respectively, eliminate appropriate rows and columns from the constraint matrix and solve the IP using branch and bound. Substantial enhancements in computational efficiency are achieved in their application (Hane et al [25]). The danger in adopting this strategy is that as a result of fixing the values of some variables, a feasible solution may not exist. As an alternative, CPLEX (the optimization software used) [11] provides a heuristic procedure to fix values of variables close to integer. Redundant rows and columns are eliminated from the constraint matrix and the IP is solved using branch and bound to obtain an integer solution, termed *the initial integer solution.* This initial integer solution is then used as an upper bound to prune the branch and bound tree and thereby enhance the computational efficiency of the procedure. Note that in this strategy, there is no danger of making the problem infeasible.

Though an *optimal integer solution* to the problem cannot be determined unless all nodes in the branch and bound enumeration tree are evaluated, it is possible to terminate the branch and bound procedure when either the current-best integer solution is a certain (small) percentage from optimality or when a certain threshold number of nodes of the branch and bound tree have been scanned. No hard-and-fast rule exists for this threshold number. Hane et al [25] adopt a node limit of 2000, while Shenoi [30] adopts node limits of 1000 and 10000 depending on problem size.

In summary, the overall strategy adopted to obtain an integer solution to the LH-FAP is a heuristic branch and bound procedure with only the root node LP relaxation solved exactly using column generation and an initial integer solution generated using the CPLEX heuristic. The following is the step-by-step procedure adopted:

1. The LHFAP LP relaxation is solved to optimality using the column generation techniques highlighted in Sections 3.1.2 and 3.1.3. Specifically the column generation subproblem is solved based on *a maximum three day elapsed time criterion* (recall that this is corresponds to an unconstrained shortest path subproblem. See Section 3.1.3).

2. The entire set of columns in the optimal root node LP solution are the only columns considered at all other nodes of the branch and bound tree. No columns are generated at the other nodes of the enumeration tree.

3. An initial integer solution, determined using the CPLEX heuristic described above, is used to prune the branch and bound tree.

4. A branch and bound procedure is executed and is terminated either when an integer solution 0.5% from optimality is determined or when 1000 nodes of the branch and bound tree are scanned.

Further details of the solution strategy adopted are presented in Section 4.2.4.

Chapter 4

Case Study

This chapter presents a case study using the international schedule of a major US long haul airline and elaborates on computational experience gained in solving the long haul fleet assignment problem. The formulation was coded in the C programming language using the CPLEX Release 3.0 optimization software (CPLEX Optimization Inc. [11]). The computational tests were run on an IBM RS 6000 Model 370 workstation.

4.1 The Data

The data, provided by a major U.S. long haul airline, consists of a weekly schedule of 1162 flights serving 55 cities worldwide. There are 75 aircraft and 11 equipment types. The airline has 8 maintenance stations with each maintaining on an average 4 fleet types.

Initially, the full problem was too large to solve and as a consequence, subproblems were created to test the model and the solution strategy outlined in Section 3.2. The creation of subproblems is a non-trivial task since the schedule needs to be balanced, with the number of departures and arrivals at each station being the same. The current fleeting adopted by the long-haul airline was available and this was used to generate the case study problems. Essentially, a subset of fleets was picked and the schedule with only those flight legs currently flown by the subset was constructed. The

Table 4.1: Case Study Problem Sizes

Problem Name	# Fleets	# Flights
P1	2	126
P2	2	408
P3	4	360
P4	4	536
P5	7	546
P6	11	1162

feasibility of the airline's current fleeting guarantees the feasibility and balance of the case study problem generated. Table 4.1 shows the sizes of the various subproblems constructed.

4.2 Implementations

This section provides a history of the implementation of the solution strategy used to solve the LHFAP. Some of the improvisations and strategies used to enhance the computational efficiency of the model are presented.

4.2.1 Node Aggregation

Node aggregation in the network model is a technique that eliminates a large number of rows (constraints) from the problem formulation. This concept has been successfully implemented by Hane, et al [25] in solving the domestic fleet assignment problem. The rationale behind the concept is that there is no need and no advantage for two arrival nodes (Figure 4-1) to be treated as distinct if there are no departure nodes between them. Similarly, there is no advantage in placing departures as distinct nodes in the time-line network later than the previous arrival. This concept of consolidating nodes into one is referred to as node aggregation or node consolidation.

In Figure 4-1, for example, only two nodes are required instead of seven, one

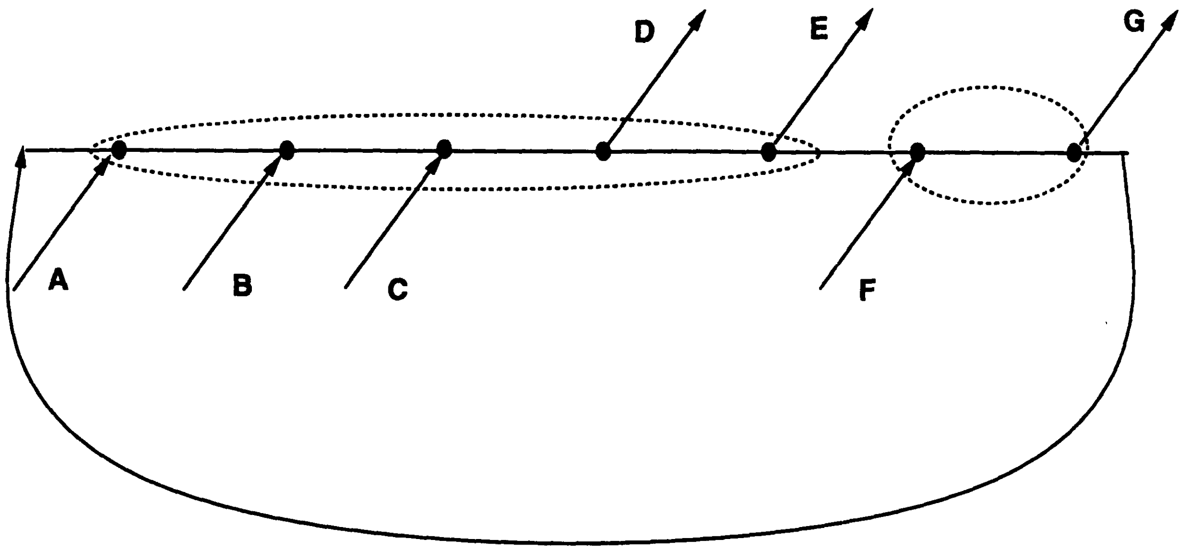

Figure 4-1: The Concept of Node Aggregation

for flights A, B, C (arrivals), D and E (departures) and another for flights F and G. Note that flights E and F cannot be combined into one node, since that would legitimize a string in which a flight would depart (E) before it arrived (F), a physical impossibility.

The reduction of the number of nodes in the network has a very interesting consequence. Note that in the LHFAP constraint matrix (Figure 2-3), the number of constraints is proportional to the product of the number of nodes in the network and the number of fleet types. A reduction in the number of nodes in the network, therefore, has a dramatic effect in decreasing the size of the problem. Table 4.2 shows the results of node aggregation on the different problems. The size of the model (in terms of number of constraints, nodes and ground arcs) reduces by a factor of 2 to 5 depending on the problem.

Empirical experiments show that the number of simplex iterations required by CPLEX to solve an LP is 2 to 3 times the number of rows in the constraint matrix. For the LHFAP, the effect of a reduced number of constraints is even more dramatic because the column generation solution procedure involves repeated solutions of LP's.

Node aggregation, therefore, reduces the computation time considerably. As a

Table 4.2: Sizes of Case Study Problems with and without Node Aggregation

Name	Consolidation	# constraints	# ground arcs	# nodes
P1	No	648	520	504
	Yes	270	142	294
P2	No	1294	884	1632
	Yes	605	195	581
P3	No	2542	2228	1440
	Yes	901	537	527
P4	No	3648	3108	2144
	Yes	1257	717	739
P5	No	4125	3572	2184
	Yes	1458	905	754
P6	No	9857	8684	4648
	Yes	2857	1684	1409

result, all further runs were performed with node aggregation incorporated in the network.

4.2.2 Preprocessing and Advanced Basis

CPLEX includes a procedure to reduce the size of the constraint matrix using substitution to eliminate rows and columns. This is achieved by CPLEX's Presolver and Aggregator by identifying and eliminating redundancies.

Table 4.3: Computation Times with and without Preprocessing and Advanced Basis

Problem Name	# flights (# fleets)	Solution Time incl Preprocessing and Advanced Basis	Solution Time without Preprocessing and Advanced Basis
P1	126 (2)	7s	9s
P2	408 (2)	1m 29s	4m 11s
P3	360 (4)	13m 54s	1h 59m
P5	546 (7)	20m 20s	1h 24m

CPLEX offers another feature allowing the current basis to be saved and used as a starting point to solve a problem similar in nature. This can be achieved by appropriately setting CPLEX's advanced basis indicator. Since column generation requires repeated solution of LP's that are similar in nature, it is possible to exploit this feature to improve the computational efficiency of the solution procedure. In other words, the basis of the LP solution at the end of each restricted master problem is saved and the solution procedure for the next iteration starts using this advanced basis.

Table 4.3 presents a comparison of computation times taken to solve problems $P1$, $P2$, $P3$ and $P5$ using the dual simplex algorithm. Clearly, preprocessing of the constraint matrix and use of an advanced basis at each iteration of the column generation solution procedure results in substantial improvement of computation times. Hence, all further runs had the CPLEX indicators set appropriately to ensure preprocessing and use of advanced basis.

4.2.3 Solving the LP: Initial Runs

Solving the LHFAP LP relaxation is possible using different simplex algorithms - primal simplex, dual simplex or network simplex. CPLEX has the following features with regard to different simplex procedures:

1. *optimize* which executes the primal simplex algorithm;
2. *dualopt* which executes the dual simplex algorithm. Problems with high degeneracy and with little variability in the right hand side coefficients, but with significant variability in the cost coefficients (note that the LHFAP fits into this class of problems) are reported to be solved much faster using the dual simplex algorithm than the primal simplex; and
3. *netopt* which exploits the network structure of a given formulation and solves the model (or part of it) using the network simplex algorithm. CPLEX has the capability to extract the network structure from a model and solve it using

Table 4.4: Primal Simplex vs Dual Simplex - Computation Results

Problem Name	Primal Simplex Soln Time	# CG Iters w/ Primal Simplex	Dual Smplx Soln Time	# CG Iters w/ Dual Simplex
P1	10s	17	32s	15
P2	3m 26s	52	32m 56s	45
P3	25m 11s	104	3h 50m 43s	95

the network simplex algorithm. The complete solution to the problem is then obtained by using either the dual simplex algorithm (*dualopt*) or the primal simplex algorithm (*optimize*), given the initial (infeasible) solution provided by solving the network portion of the formulation with the network simplex algorithm.

Primal simplex vs Dual simplex

First, the primal and dual simplex algorithms were tested. The computational results for problems $P1$, $P2$ and $P3$ are summarized in Table 4.4. Hane et al [25] in their solution of the domestic fleet assignment problem achieved substantial gains in solution time for some data sets, using a steepest-edge pricing strategy, the results of which for our problems are reported in the next section.

Steepest-Edge Pricing

All variants of the simplex method move from one vertex of the polyhedron of feasible solutions to another along edges that are "downhill" such that the objective function decreases *. The idea behind steepest-edge pricing is that at each iteration, "the most downhill" edge or the one with steepest improvement in objective function is chosen (Forrest and Goldfarb [23]). A steepest edge pricing strategy was, therefore, tested for case study problems $P1$, $P2$, $P3$ and $P5$ using both primal and dual simplex algorithms. The results are summarized in Table 4.5.

*It is assumed here that a minimization problem is being solved.

Table 4.5: Steepest-Edge Pricing Performance

Problem Name	Primal Steepest-edge Soln Time	# CG Iters with PSE	Dual Steepest-edge Soln Time	# CG Iters with DSE
P1	8.5s	17	7s	14
P2	2m 12s	46	1m 29s	41
P3	11m 28s	80	13m54s	86
P5	26m 35s	70	20m 20s	61

The dual simplex algorithm usually performs better than primal simplex when steepest-edge pricing is used. The reduction in run times for both the primal and dual simplex solvers using steepest-edge pricing, results from a reduction in the number of column generation iterations required (Tables 4.4 and 4.5). All further runs are carried out using the dual simplex algorithm using a steepest-edge pricing strategy.

Dual simplex vs Network Simplex

The LHFAP flow balance constraints (Equations (2.4)), have "a pure network" structure. These constraints constitute the major portion of the formulation as can be observed from Figure 2-3. Table 4.6 presents a comparison of the number of "pure network" (flow balance) constraints and "non-network" (flight coverage and aircraft count) constraints in the LHFAP formulation. The flow balance constraints are $\sum_{f \in F} N_{mf}$ in number, where N_{mf} represents the number of maintenance nodes for fleet type $f \in F$, or about 55% of the total number of constraints with node consolidation and about 85% of the total number of constraints without node consolidation.

The case study problems were tested using both dual simplex (with steepest-edge pricing) and network simplex algorithms and the computational results are summarized in Table 4.7. Note that the formulation for the LHFAP does not have a pure network structure. CPLEX extracts and solves the embedded network portion of the formulation using the network simplex algorithm. The primal or dual simplex algorithm is used to solve the overall problem with the other non-network constraints

Table 4.6: Number of "Pure Network" and "Non-network" Constraints with and without Node Consolidation - A Comparison

Problem Name	Node consldn done ?	# network constraints	# non-network constraints	Network constraints as % of total
P1	No	520	128	80.2%
	Yes	142	128	52.5%
P2	No	884	410	68.3%
	Yes	195	410	32.2%
P3	No	2228	364	87.6%
	Yes	537	364	59.6%
P4	No	3108	540	85.2%
	Yes	717	540	57.0%
P5	No	3572	553	86.6%
	Yes	905	553	62.1%
P6	No	8684	1173	88.1%
	Yes	1684	1173	58.9%

Table 4.7: Dual Simplex vs Network Simplex - Computation Results

Name	LP Solver	Solution Time	# rows	# columns	# non-zeroes
P1	DS	7s	270	1146	5124
	NS	9s	270	1132	4982
P2	DS	1m 29s	605	3565	19189
	NS	3m 50s	605	3627	19584
P3	DS	13m 54s	901	8506	50994
	NS	28m 45s	901	9218	55262
P5	DS	20m 20s	1458	11122	61756
	NS	1h 47m 20s	1458	10984	60928

using the optimal basis from the network simplex as the starting basis. Based on the discussion in the previous section, the dual simplex algorithm is used.

It was found that dual simplex (with steepest-edge pricing) was faster than network simplex, especially as the problem size got bigger. This could be explained by the fact that node aggregation eliminates a large number of constraints that constitute the network portion of the formulation. The effect of the non-network portion of the formulation was therefore more pronounced. The dual simplex algorithm was therefore used.

Example: *From Table 4.6 for problem* $P6$ *with 1162 flights and 11 fleets, there are 8684 "pure network" constraints without node consolidation. After node consolidation is performed, the number of the same decreases dramatically to 1684. In each case, the number of "non-network" constraints is* $1162 + 11 = 1173$. *Clearly the network portion of the formulation decreases substantially as a consequence of node consolidation and hence the decrease in the computational efficiency of the network simplex algorithm.*

Solution of Column Generation Subproblem

As discussed in Section 3.1.3, the LHFAP column generation subproblem is modeled as a shortest path problem and the FAA maintenance criteria are explicitly incorporated based either on a maximum flying time scenario (constrained shortest path problem) or on a maximum elapsed time scenario (unconstrained shortest path problem). Specifically, we model the maintenance requirements based on the latter (Section 3.2.2) and incorporate the requirement of maintenance at least once every three elapsed days.

In order to establish that the constrained shortest path algorithm is computationally more expensive, problems $P1$, $P2$, $P4$ and $P5$ are tested with both the constrained shortest path implementation and an unconstrained shortest path implementation in which strings that exceed 45 flying hours are not considered. The results of the run times are summarized in Table 4.8. The maximum flying time scenario takes substantially greater computation time without a comparable improvement in

Table 4.8: Unconstrained vs Constrained Shortest Path Problems - A Comparison

Problem name	# flights (# fleets)	Shortest Path Problem	Solution Time	Solution Value
P1	126 (2)	Unconstrained	7s	2960369.00
		Constrained	17s	2960369.00
P2	408 (2)	Unconstrained	2m 1s	10611441.00
		Constrained	6m 43s	10641120.00
P4	536 (4)	Unconstrained	38m 38s	13624637.45
		Constrained	4h 14m 22s	13563334.35
P5	546 (7)	Unconstrained	23m 11s	16286255.29
		Constrained	1h 45m 46s	16179416.08

solution value. The maximum elapsed time scenario is, therefore, adopted.

Solving to LP Optimality: Summary of Implementation details

The implementation details of the overall algorithm to solve the LP relaxation of the LHFAP is summarized in Figure 4-2. The network is generated and nodes are consolidated, resulting in a reduced size network. The optimization solver's preprocessor is used to eliminate redundancies and further simplify the constraint matrix. The problem is then solved using an artificial basis and the dual simplex algorithm. The basis is saved and, using the dual solution, the subproblem is solved with the three day maximum elapsed time scenario and columns with negative reduced cost are added. Using the advanced basis, the new restricted master problem is solved. This process continues until all columns have non-negative reduced cost and the LHFAP LP is solved. Then, the branch and bound solver is invoked.

4.2.4 Solving the IP: Branch and Bound Implementation Details

After solving the LHFAP LP relaxation at the root node of the branch and bound tree, the entire set of columns in the constraint matrix is passed to CPLEX's IP

Solution Steps:

1. Generate network and consolidate nodes.
2. Build artificial basis
3. Aggregate using CPLEX's preprocessor.
4. Solve LP using dual simplex algorithm with steepest-edge pricing.
5. Save basis to use as an advanced basis for next iteration.
6. Price-out attractive columns using dual solution using a maximum 3-day elapsed time criterion and add to constraint matrix.
7. Repeat Steps 4 through 6 until LP is solved.

Figure 4-2: Solution Algorithm for the LHFAP LP Relaxation

solver.

CPLEX's rounding heuristic is used to obtain a first integer solution. The heuristic fixes values of the variables in the LP solution at or near integer to those values. A depth-first-search branching is carried out and an integer solution is found. This initial integer solution serves as an upper bound on the optimal IP solution. A branch and bound procedure is then carried out with the initial integer solution used to prune the branch and bound tree.

The path of the optimizer through the branch and bound tree is determined by certain user inputs. For example, from a given node, it is possible to either delve deeper into the branch and bound tree or move up the tree (i.e., backtrack). In our solution procedure, a depth-first search strategy is used to select the next node in the branch and bound procedure and a strong branching scheme † is invoked to select the variable to branch on at the node selected. Based on computational experience

†CPLEX partially solves a number of problems with tentative branches and selects the most promising branch under the strong branching scheme of variable selection at a node.

Solution Steps:

1. Solve the root node LHFAP LP relaxation using implementation details described in Fig. 4-2 and pass all columns generated to CPLEX's IP solver.

2. Perform CPLEX heuristic to fix values of variables close to integer, and obtain initial integer solution.

3. Carry out branch and bound procedure with a depth first search for selecting nodes and a strong branching scheme to select the variable on which to branch.

4. Prune the branch and bound tree where possible using the initial integer solution obtained using the CPLEX heuristic.

5. Terminate branch and bound procedure when an IP-LP gap of 0.5% is achieved or when 1000 nodes of the branch and bound tree have been scanned.

Figure 4-3: Overall Algorithm for Solution of the LHFAP in solving the root-node LP, the dual simplex algorithm is used to solve the LP's at each node of the branch and bound tree.

Given we are solving a minimization problem, the root-node LP optimum value Z^*_{lp}, is a lower bound on the integer program optimum value Z^*_{ip}. In the event that the branch and bound procedure results in an integer solution value equal to the LP optimum, the IP solution is optimal. However, this is seldom the case in practical applications. The IP-LP gap is defined as the ratio of the difference of the IP and LP solution values to the LP solution value, i.e.,

$$IP - LP \;\; Gap = \frac{Z^*_{ip} - Z^*_{lp}}{Z^*_{lp}}$$

As explained in Section 3.2.2, the branch and bound procedure is terminated either if this gap is smaller than a threshold value of 0.5% or if 1000 nodes in the branch and bound tree have been scanned.

We define $C_{min} = \sum_{f \in F} C_{f-min}$, where C_{min} is the minimum cost incurred in flying the schedule and C_{f-min} is the minimum cost of flying flight f across all fleet types. C_{min} is therefore the cost incurred in flying the schedule if all constraints are relaxed. Given this cost, we define the optimality gap as:

$$Optimality \;\; Gap = \frac{(Z^*_{ip} - C_{min}) - (Z^*_{lp} - C_{min})}{(Z^*_{lp} - C_{min})}$$

Figure 4-3 summarizes the implementation details of the overall strategy adopted to solve the LHFAP.

4.3 Computational Results and Analysis

This section presents and analyzes the results of the case study and presents the improvement achieved by the model over the current fleeting adopted by the airline.

Table 4.9: Results of Root-node LP

Problem name	# rows	# cols gen	# non-zeroes	Soln time	Solution value	C_{min} value
P1	270	1154	5109	7s	2960369.00	2960369.00
P2	605	3565	19189	1m29s	10606093.00	10439404.00
P3	901	8506	50994	13m54s	8179388.50	8047409.00
P4	1257	12457	123625	37m2s	13561351.23	13486450.00
P5	1458	11122	61756	20m20s	16286255.29	15524764.00
P6	2857	36950	219806	9h6m36s	31692362.23	29597270.00

Table 4.10: Branch and Bound IP Solution

Problem Name	Soln Time	IP Solution Value	LP Solution Value	IP-LP Gap	Optmlty Gap
P1	16s	2960369.00	2960369.00	0.00%	0.00%
P2	1m 50s	10606491.00	10606093.00	0.003%	0.24%
P3	29m 47s	8186500.00	8179388.50	0.001%	5.38%
P4	81m 31s	13626702.00	13561351.23	0.005%	87%
P5	24m 56s	16288810.00	16286255.29	0.01%	0.33%
P6	17h 9m 10s	31746628.00	31692362.23	0.17%	2.59%

4.3.1 Results

The LP relaxation of problems $P1$ through $P6$ were solved using the steps discussed above. Table 4.9 provides the final solution characteristics at optimality. The results of the branch and bound IP solutions to the problems are summarized in Table 4.10. Further results of branching are presented in Table 4.11.

Table 4.12 shows the improvement by the fleeting generated in this thesis compared to the current fleeting as flown by the long haul airline. Our LHFAP solutions consistently have a lower objective function (in other words, a lower operating cost for the airline) compared to that of the airline's current fleeting. While the improvements for the subproblems $P1$ through $P5$, with the exception of problem $P4$, are not very significant, the savings for the overall weekly schedule, namely problem $P6$, is a

Table 4.11: Branch and Bound Results

Problem name	Nodes in B & B	Node of First Integer	Node of Optimal Integer	# of integral Solns	Time in B & B
P1	10	10	10	1	9s
P2	9	9	9	1	21s
P3	66	66	66	1	15m 55s
P4	361	57	361	14	44m 29s
P5	36	36	36	1	9m 36s
P6	1000+	255	851	3	8h 2m 34s

Table 4.12: LHFAP Objective Function Value vs Current Fleeting Objective Function Value - A Comparison

Problem Name	# flights (# fleets)	Optimal Fleeting Obj Fn Value	Current Fleeting Obj Fn	Change
P1	126 (2)	2960369.00	2960369.00	0%
P2	408 (2)	10606491.00	10724828.00	(1.1%)
P3	360 (4)	8186500.00	8332379.00	(1.8%)
P4	536 (4)	13626702.00	14086938.00	(3.3%)
P5	546 (7)	16288810.00	16403081.00	(0.7%)
P6	1162 (11)	31746628.00	33924915.00	(6.4%)

substantial fraction of the cost. This could translate into tens of millions of dollars of savings annually.

Islands - A Boon or a Bane?

Hane et al [25] describe a technique that exploits the topology of the fleet assignment network and reduces the size of the model further. The technique forces the minimum number of aircraft to be used, thereby avoiding having unnecessary aircraft on the ground. If, at a time t, there is no aircraft on the ground, then, after an equal number of arrivals and departures, there will be no aircraft on the ground. The

Table 4.13: Island Implementation - IP Results

Problem name	Solution Time	IP Solution value	% inc over LHFAP soln
P1	7.5s	2960369.00	0.00%
P2	35.0s	10680091.00	0.69%
P3	7m 35s	8186396.00	(0.001)%
P4	4m 31s	13663184.00	0.26%
P5	10m 43s	16305356.00	0.10%
P6	6h 21m 58s	31790690.00	0.14%

Table 4.14: Island Implementation - LP Results

Problem name	LP Solution value	LP Solution time	# Cols in LP
P1	2960369.00	4.5s	728
P2	10680091.00	28.1s	2115
P3	8179388.50	4m 51s	5373
P4	13653095.17	10m 35s	8354
P5	16300865.67	9m 2s	7838
P6	31739404.35	3h 20m 32s	22273

ground arcs corresponding to these times can be removed from the network, resulting in a time-line consisting of *islands*. The drawback, however, is that a least cost solution may require one or more aircraft to be on the ground at all times. Thus, creation of islands may prevent generation of strings that might have had negative reduced cost otherwise. This could result in a poorer (higher cost) solution. Even so, we implemented the above concept. Tables 4.13 and 4.14 summarize the IP and LP results of the runs with the island technique incorporated.

Implementation of islands results in a dramatic decrease in solution times, while solution values increase for all problems (with the exception of $P1$ and $P3$, for which the solution values remain the same). Note that there is a decrease in the number of columns generated at root node LP optimality, a consequence of island creation. There is, therefore, a trade-off between solution time and solution quality. Since fleet

Table 4.15: A Comparison of LP and IP Solution Times

Problem name	# flights (# fleets)	Time for Root Node LP as % of total	Time for IP as % of total
P1	126 (2)	44%	56%
P2	408 (2)	81%	19%
P3	360 (4)	48%	52%
P4	536 (4)	45%	55%
P5	546 (7)	82%	18%
P6	1162 (11)	53%	47%

Table 4.16: An Analysis of LP Solution Time

Problem name	# flights (# fleets)	Time taken by Master Problem as % of total	Time taken by Subproblem as % of total	Total # of Col. Gen. Iterations
P1	126 (2)	86%	14%	14
P2	408 (2)	90%	10%	41
P3	360 (4)	83%	17%	86
P4	536 (4)	95%	5%	47
P5	546 (7)	92%	8%	61
P6	1162 (11)	98%	2%	82

assignment is usually carried out once every quarter or once every month at best, solution times are not very crucial. However, from a scenario-analysis standpoint, a decrease in solution time is most desirable. We leave the reader to make his/her choice.

4.3.2 Analysis of results

Table 4.15 shows the fraction of the total solution times taken to solve the root node LP relaxation of the LHFAP and the branch and bound procedure to obtain integral solutions.

On average, about 60% of the total solution time is spent in solving the LHFAP LP relaxation and the remaining 40% is spent in branch and bound. The root node

LP solution time is further broken down (Table 4.16) into the fraction of time spent in solving the restricted master problem and the time spent in solving the subproblems. From Table 4.16, it is clear that the solution of the LP's using the dual simplex algorithm consumes the bulk of the time taken to solve the root node LP. In fact, for problem $P6$, 98% of the time is spent solving the master problem of the root node LP resulting in the following breakdown of the total time spent solving the LHFAP, averaged over all problems:

$$\begin{aligned} Root\ Node\ LP\ Solution &= 55\% \\ Shortest\ Path\ Subproblem\ Solution &= 5\% \\ LHFAP\ IP\ Solution &= 40\% \end{aligned}$$

Chapter 5

Conclusions and Future Research

This main contributions of this thesis are a novel LHFAP formulation capturing maintenance considerations and an effective implementation and solution procedure. The formulation has been tested using the data of a long haul airline and the fleeting produced results in a substantial savings in operating cost over the current fleeting flown by the airline. In the course of this work, additional research has been identified as follows':

1. Work needs to be undertaken to make the constrained shortest path procedure to price out negative reduced cost strings based on the maximum flying time criterion computationally efficient. An implementation with efficient utilization of memory and an effective means of reducing run times is desirable.

2. The implementation of a *branch-and-price* procedure and comparisons of results would be interesting. Recall, branch-and-price procedures generate columns at each node of the branch and bound tree. However, considering the IP-LP gaps obtained, branch-and-price would only be effective if a major savings in computation time can be achieved. This might be possible since the root node LP could be initialized with fewer columns, perhaps only the columns in the optimal basis of the LP relaxation.

3. Different prioritized branching strategies could be tested and these could provide interesting comparisons and insights. See Hane et al. [25] for a discussion of

various prioritized branching strategies.

4. Besides the use of an artificial basis to start the solution of the root-node LP, an advanced basis, e.g., a current fleeting, might result in a tremendous improvement in solution time. Work could be undertaken to develop heuristics that generate strings based on available data, such as the current fleeting adopted by the airline.

5. The high run time for the LHFAP LP relaxation and possibilities of reducing the same, need to be investigated. Recall that the time taken to solve an LP directly without maintenance constraints (Section 2.1.3) is substantially lower than the time taken per column generation iteration in our solution procedure of the LHFAP LP relaxation. For instance, interior point methods could be investigated, since similar applications have been solved efficiently using interior point methods (Hane et al [25]).

Bibliography

[1] J. Abara. Applying Integer Linear Programming to the Fleet Assignment problem. *Interfaces*, 19:20–28, 1989.

[2] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms and Applications.* Prentice Hall, 1993.

[3] R. Anbil, R. Tanga, and E.L. Johnson. A global optimization approach to crew scheduling. Technical Report COC-9105, Georgia Institute of Technology, 1991.

[4] E.K. Baker, L.D. Bodin, W.F. Finnegan, and R.J. Ponder. Efficient heuristic solutions to an airline crew scheduling problem. *AIIE Transactions*, 2(11):79–85, June 1979.

[5] C. Barnhart, E.L. Johnson, R. Anbil, and L. Hatay. A column generation technique for the long-haul crew assignment problem. In T. Ciriano and R. Leachman, editors, *Optimization and Industry*, volume II. John Wiley and Son, 1993.

[6] M. Ben-Akiva and S.R. Lerman. *Discerte Choice Analysis: Theory and application to travel demand.* MIT Press, Cambridge, Massachusetts, 1985.

[7] M.E. Berge and C.A. Hopperstad. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Operations Research*, 41(1):153–168, 1993.

[8] D. Berstimas and J. Tsitsikilis. Introduction to Optimization, 1994.

[9] S.P. Bradley, Hax A.C., and T.L. Magnanti. *Applied Mathematical Programming.* Addison-Wesley Publishing Company, 1977.

[10] A.L. Brearly, G. Mitra, and H.P. Williams. Analysis of mathematical programming problems prior to applying the simplex method. *Mathematical Programming*, 8(1):54–83, 1975.

[11] CPLEX Optimization Inc., California. *Using the CPLEX Callable Library for CPLEX*, release 3.0 edition, © 1989-1994.

[12] G.B. Dantzig and P. Wolfe. Decomposition principles for linear programs. *Operations Research*, 27:101–111, 1960.

[13] G.B. Dantzig and P. Wolfe. The decomposition algorithm for linear programming. *Econometrica*, 29:767–778, October 1961.

[14] M.S. Daskin and N.D. Panayotopoulos. A Lagrangian Relaxation Approach to Assigning Aircraft to Routes in Hub and Spoke Networks. *Transportation Science*, 23(2), 1989.

[15] N. Deo and Pang C. Shortest path algorithms: Taxonomy and annotation. *Networks*, 14:275–323, 1984.

[16] M. Desrochers and F. Soumis. A generalized permanent labelling algorithm for the shortest path problem with time windows. *INFOR*, 26(1):191–212, 1988.

[17] M. Desrochers and F. Soumis. A reoptimization algorithm for the shortest path problem with time windows. *European Journal of Operations Research*, 35:242–254, 1988.

[18] M. Desrochers and F. Soumis. A column generation approach to the urban transit crew schedulig problem. *Transportation Science*, 23(1):1–13, Feb 1989.

[19] J. Desrosiers, Y. Dumas, M.M. Solomon, and F. Soumis. A breakthrough in airline crew scheduling. Working Paper, 1991.

[20] J. Dillon, R. Gopalan, S. Ramachandran, R. Rushmeir, and K.T. Talluri. Fleet Assignment at USAir. In *TIMS/ORSA Joint National Meeting*, Phoenix, AZ, 1993.

[21] I. Elce. The Development and Implementation of Air Canada's Long Range Planning Model. *AGIFORS Proceedings*, 8(1):1–23, 1970.

[22] T.A. Feo and J.F. Bard. Flight Scheduling and Maintenance Base Planning. *Management Science*, 35(12):1415–1432, December 1989.

[23] J.J. Forrest and D. Goldfarb. Steepest-edge simplex algorithms for linear programming. *Mathematical Programming*, 57:341–374, 1992.

[24] R. Gopalan and K.T. Talluri. The Aircraft Maintenance Routing Problem. Working Paper, January 1995.

[25] C.A. Hane, C. Barnhart, E.L. Johnson, R.E. Marsten, G.L. Nemhauser, and G. Sigismondi. The Fleet Assignment Problem: Solving a Large-Scale Integer Program. Technical Report COC-92-04, Georgia Institute of Technology, School of Industrial & Systems Engineering, Atlanta, GA 30332, 1993.

[26] A. Kanafani and A.A. Ghobrial. Airline Hubbing - Some Implications for Airport Economics. *Transportation Research A*, 19A(1):15–27, 1985.

[27] L. Lotosky. Personal communication, April 1995.

[28] M. Parker and J. Ryan. A Column Generation Algorithm for Bandwidth Packing. Working Paper, October 1993.

[29] L.T. Phillips. Air Carrier Activity at Major Hub Airports and Changing Interline Practices in the United States Airline Industry. *Transportation Research A*, 21A(3):215–221, 1987.

[30] R.G. Shenoi. Solving the Long Haul Crew Pairing Problem. Master's thesis, Massachusetts Institute of Technology, Center for Transportation Studies, February 1994.

[31] R.W. Simpson and P.P. Belobaba. Course notes Air Transportation Economics, June 1982, Updated August 1992.

[32] R. Subramaniam, R.P. Scheff Jr., J.D. Quillinan, D.S. Wiper, and R.E. Marsten. Coldstart: Fleet Assignment at Delta Air Lines. *Interfaces*, 24:104–120, January - February 1994.

[33] K.T. Talluri. The Four-Day Aircraft Maintenance Routing Problem. Working Paper, November 1994.

[34] K.T. Talluri. Swapping Applications in a Daily Airline Fleet Assignment. Working Paper, August 1994.

[35] K.T. Talluri and R. Gopalan. Mathematical Models in Airline Schedule Planning: A Survey. Working Paper, November 1994.

[36] Zaphod. *Those were the Best Years of my Life.* Godav, IIT, Madras, India, 1989 - 1993.