

Dynamic Flow Management Problems in Air Transportation

by

Sarah Stock Patterson

Bachelor of Science, Cornell University, 1992

Submitted to the Department of Electrical Engineering and Computer
Science

in partial fulfillment of the requirements for the degree of

Doctorate of Philosophy in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1997

© Massachusetts Institute of Technology 1997. All rights reserved.

Signature of Author *Sarah Stock Patterson*
Department of Electrical Engineering and Computer Science
May 23, 1997

Certified by *Dimitris Bertsimas*
Dimitris J. Bertsimas
Leaders for Manufacturing Professor of Operations Research
Thesis Supervisor

Accepted by *Thomas L. Magnanti*
Thomas L. Magnanti
George Eastman Professor of Management Science
Professor of Electrical Engineering and Computer Science
Codirector, Operations Research Center

*1997
10-03-97
DIT
102607*

Dynamic Flow Management Problems in Air Transportation

by

Sarah Stock Patterson

Submitted to the Department of Electrical Engineering and Computer Science
on May 23, 1997, in partial fulfillment of the
requirements for the degree of
Doctorate of Philosophy in Operations Research

Abstract

In 1995, over six hundred thousand licensed pilots flew nearly thirty-five million flights into over eighteen thousand U.S. airports, logging more than 519 billion passenger miles. Since demand for air travel has increased by more than 50% in the last decade while capacity has stagnated, congestion is a problem of undeniable practical significance. In this thesis, we will develop optimization techniques that reduce the impact of congestion on the national airspace. We start by determining the optimal release times for flights into the airspace and the optimal speed adjustment while airborne taking into account the capacitated airspace. This is called the *Air Traffic Flow Management Problem (TFMP)*. We address the complexity, showing that it is NP-hard. We build an integer programming formulation that is quite strong as some of the proposed inequalities are facet defining for the convex hull of solutions. For practical problems, the solutions of the LP relaxation of the TFMP are very often integral. In essence, we reduce the problem to efficiently solving large scale linear programming problems. Thus, the computation times are reasonably small for large scale, practical problems involving thousands of flights. Next, we address the problem of determining how to reroute aircraft in the airspace system when faced with dynamically changing weather conditions. This is called the *Air Traffic Flow Management Rerouting Problem (TFMRP)*. We present an integrated mathematical programming approach for the TFMRP, which utilizes several methodologies, in order to minimize delay costs. In order to address the high dimensionality, we present an aggregate model, in which we formulate the TFMRP as a multicommodity, integer, dynamic network flow problem with certain side constraints. Using Lagrangian relaxation, we generate aggregate flows that are decomposed into a collection of flight paths using a randomized rounding heuristic. This collection of paths is used in a packing integer programming formulation, the solution of which generates feasible and near-optimal routes for individual flights. The algorithm, termed the Lagrangian Generation Algorithm, is used to solve practical problems in the southwestern portion of United States in which

the solutions are within 1% of the corresponding lower bounds.

Thesis Supervisor: Dimitris J. Bertsimas

Title: Leaders for Manufacturing Professor of Operations Research

Acknowledgments

I would like to begin by offering a well-deserved thanks to my advisor and friend, Dimitris Bertsimas. Thank you for your advice and guidance which have contributed to my growth as a researcher and an educator. Thank you for your patience and encouragement, especially during those times when my vigor for research waned. I would like to thank Amedeo Odoni for inspiring the work in this area and for being a constant source of guidance and support during the course of this research. I would also like to thank Tom Magnanti for serving on my committee, for offering such educational and enjoyable Friday morning seminars and for leading the Operations Research Center with such dedication. One could not have asked for better role models than these three gentlemen.

I would like to acknowledge the FAA and Draper Laboratories for providing the funding for this research undertaking. In particular, I would like to thank Dr. David Winer and Dr. Thomas Mifflin of the FAA and Dr. Steve Kolitz of Draper Laboratories for encouraging and supporting our work. I would like to thank Dr. Kenneth Lindsay for providing data, Ms. Wandy Sae-tan for performing some of the computational experiments and Dr. Eugene Gilbo for providing some of the figures.

To Paulette, Laura, Lisa, and Katana, thank you for making the ORC a nice place to be and for going out of your way for me on so many occasions. To all my friends at MIT: Kyle, Elaine, Thalia, Rodrigo, Edi, Beril, Jim, Gina and Yannis, who have all contributed to the happiness I have found there. To Georgia, thanks for listening and telling me that things will work out in the long run. To Emily, thanks for your cherished friendship and for making some very difficult years enjoyable. I would like to thank my family; with a special thanks to my niece, Emi, for keeping everyone smiling. To my dog, Ziggy; thanks for keeping me company during long hours at the terminal.

Last, but not least, to Doug, whom I love and admire with all my heart.

Dedicated to my father,
Dr. T.A.C. Stock
who would have loved to read this.

Contents

1	Introduction	11
1.1	The Air Transportation Industry	11
1.2	Dynamic Flow Management Problems	13
1.3	Literature Review	15
2	The Air Traffic Flow Management Problem	18
2.1	The 0-1 IP Formulation	18
2.2	The Multi-Airport Ground-Holding Problem as a Special Case	28
2.3	Insights from the Polyhedral Structure	32
2.4	Complexity of the TFMP	33
2.5	Modeling Variations	35
2.5.1	Dependence Between Arrival and Departure Capacities	36
2.5.2	Hub Connectivity with Multiple Connections	38
2.5.3	Banks of Flights	38
2.5.4	Rerouting of Aircraft	40
3	Computational Results for the Traffic Flow Management Problem	43
3.1	Ground-Holding Problem Computations	44
3.2	Air Traffic Flow Management Problem Computations	47
4	The Air Traffic Flow Management Rerouting Problem	52
4.1	The Integer Multicommodity Dynamic Network Flow Formulation	54

4.2	Modeling the Multiple Airline Problem	62
4.3	The Lagrangian Generation Algorithm	62
4.3.1	Lagrangian Techniques	64
4.3.2	Randomized Rounding Heuristic	65
4.3.3	The Integer Packing Formulation	67
4.4	TFMP and MAGHP as Special Cases	70
4.4.1	The TFMP Dynamic Network Flow Formulation	70
4.4.2	The MAGHP Dynamic Network Flow Formulation	72
5	Computational Results for the Lagrangian Generation Algorithm	74
5.1	Computations for Weather Scenario I	75
5.2	Computations for Weather Scenario II	80
5.3	Computations for Weather Scenario III	84
6	Conclusions	88
6.1	Contributions	88
6.2	Other FAA Issues	90
A	On the Polyhedral Relationships Between Ground Holding Formulations	93
B	On the Non-integrality of the Polyhedron P_{MAGHP}	95
C	Facet Defining Constraint Proofs	99

List of Figures

2-1	US Map with sector regions.	20
2-2	Two possible flight routes.	22
2-3	Complete Runway Configuration for Logan Airport.	36
2-4	Runway Departure/Arrival Allocation for a Specific Configuration.	37
3-1	Sector flow model for test case.	49
4-1	Alternative Routes Taken as Flights Avoid a Low Capacity Region.	53
4-2	A network corresponding to four airports and six sectors.	55
4-3	An airport is modeled with four nodes.	55
5-1	Sector map of the US with southwest region shown.	75
5-2	Weather Scenario I at 8am representing normal operating conditions before the weather front has hit this region.	76
5-3	Weather Scenario I at: (a) 9:45am (b) 10:35am (c) 11:25am (d) 12:15pm.	77
5-4	Two routes used at different times to fly from Las Vegas to Phoenix.	78
5-5	Time lines of two flights that use route 2 on Figure 5-4.	79
5-6	Weather Scenario II: (a) 8:20 to 9:20 am, (b) 9:25 to 10:25am, (c) 10:30 to 11:30am, (d) 11:35am to 12:35pm, (e) 12:40 to 1:40pm, (f) 1:45 to 2:45pm.	81
5-7	Three routes used to fly between Salt Lake City and Phoenix.	83
5-8	Weather Scenario III for 200 flights: (a) 8:20 to 9:00am, (b) 9:05 to 9:45am, (c) 9:50 to 10:30am, (d) 10:35 to 11:15am, (e) 11:20am to 12:00pm, (f) 12:05 to 12:45pm.	85

C-1 First Half of Matrix of Solutions to IP_{MAGHP} 104
C-2 Second Half of Matrix of Solutions to IP_{MAGHP} 105

List of Tables

3.1	Results at the infeasibility border for 1000 flights.	44
3.2	Results at the infeasibility border for 3000 flights.	45
3.3	Previous Results at the infeasibility border for 1000 flights.	46
3.4	Previous Results at the infeasibility border for 3000 flights.	46
3.5	Results for varying capacity levels for 1000 flights.	47
3.6	Results for varying capacity levels for 3000 flights.	48
3.7	Results for varying capacity levels for 1000 flights.	50
5.1	Computational Results for weather scenario I.	78
5.2	Computational Results for weather scenario II.	82
5.3	Computational Results for weather scenario II for 15 iterations.	82
5.4	Computational Results for 200 flight dataset.	86
5.5	Computational Results for 200 flight dataset for 12 iterations.	87

Chapter 1

Introduction

The thesis is structured as follows. In Chapter 1, we introduce two dynamic flow management problems that are the topic of this thesis and review the literature to provide a framework for our contribution. In Chapter 2, we formally introduce the air traffic flow management problem, present our formulation, address the complexity, discuss some modeling variations and examine the theoretical properties of our formulation. We prove that the proposed constraints are facet defining which provides insight into the computational performance. In Chapter 3, we report computational results for the air traffic flow management problem and an important special case called the multiple airport ground holding problem. In Chapter 4, we discuss the air traffic flow management rerouting problem and present the Lagrangian Generation Algorithm. In Chapter 5, we present computational results for the air traffic flow management rerouting problem based on real data. In Chapter 6, we include some concluding remarks and directions for future research. Technical proofs are provided in the appendices.

1.1 The Air Transportation Industry

In 1995, over six hundred thousand licensed pilots flew nearly thirty-five million flights into over eighteen thousand U.S. airports, logging more than 519 billion passenger miles.

Since demand for air travel is increasing by more than 50% each year and capacity is stagnating, congestion is a problem of undeniable practical significance. For U.S. airlines, the expected yearly cost of the resulting delays was estimated at \$3 billion in 1995. In order to put this number in perspective, the total reported *losses* of all U.S. airlines amounted to approximately \$2 billion in 1991 and \$2.5 billion in 1990. In fact, it was not until 1995 that the total net profit actually became positive. Furthermore, every day 700 to 1100 flights are delayed by 15 minutes or more. European airlines are in a similar plight.

Faced with the realities of congestion, the FAA has been using *ground-holding* policies to reduce delay costs. These short-term policies consider airport capacities and flight schedules as fixed for a given time period, and adjust the flow of aircraft on a real-time basis by imposing "ground holds" on certain flights. Such a flight is then *held* on the ground at its departure airport even if it is otherwise ready for takeoff.

The basis for ground-holding relies on the fact that while a flight is airborne it incurs fuel, safety and other costs that are not applicable before the flight takes off. Thus, airborne delays are much costlier than ground delays. If an aircraft departs on time, only to encounter airborne congestion as it awaits landing clearance at the destination airport, it may incur an airborne delay. However, by delaying its departure with a ground delay, the aircraft arrives at its destination at a later time when minimal congestion is expected, thus, avoiding the costly airborne delay. Therefore, the objective of ground-holding policies is to "translate" costlier anticipated airborne delays to the ground.

Unfortunately, merely imposing ground holds only addresses a small portion of the problem of alleviating congestion. Greater benefit could be realized by taking a more complete view of the national airspace. We can control the time that a flight reaches the arrival airport by controlling its speed throughout its route, rather than forcing the flight to wait on the ground before departure. This may be an attractive option especially if the departure capacity is decreasing with time. Moreover, we could adjust the route of a flight. For instance, if there is a bad weather system that creates an airspace region of

limited capacity, we could reroute a flight along a different path to attempt to avoid the weather system.

1.2 Dynamic Flow Management Problems

In this thesis, we develop optimization techniques that encompass all aspects of the national airspace in order to minimize cost associated with delay. In particular, we address two important problems in air traffic control, the *Air Traffic Flow Management Problem (TFMP)* and the *Air Traffic Flow Management Rerouting Problem (TFMRP)*.

Besides determining release times for aircraft (ground-holding), the *Air Traffic Flow Management Problem (TFMP)* also determines the optimal airspeed adjustment of aircraft for a network of airports taking into account the capacitated airspace. Thus, the TFMP determines how to control a flight throughout its duration, not simply before its departure. This thesis makes the following contributions to this problem.

We build an integer programming formulation whose objective is the minimization of delay costs. The formulation is quite strong as some of the proposed inequalities are facet defining for the convex hull of solutions. We address the complexity of the TFMP and show that it is NP-hard. We illustrate how our models can be adjusted to account for several variations in the problem's characteristics. When modified for a special case previously addressed in the literature, called the multiple airport ground holding problem, we prove that the LP relaxation bound of our formulation is at least as strong as all others proposed in the literature. For practical problems, the solutions of the LP relaxation of the TFMP are very often integral, so there is no need to branch and bound. In essence, our formulations reduce the problem to efficiently solving large scale linear programming problems. As a result, the computation times are reasonably small for large scale, realistic size problems involving thousands of flights. Short computational times and integrality properties are particularly important, since these models are intended to be used on-line and solved repeatedly during a day.

If we add the final complication, rerouting of flights due to drastic fluctuations in the available capacity of airspace regions, we obtain the *Air Traffic Flow Management Rerouting Problem (TFMRP)*. In this problem, a flight may be rerouted through a different flight path in order to reach its destination if the current route passes through a region that has very low capacity for reasons usually related to poor weather conditions. This thesis will make the following contributions to this problem.

We present an integrated mathematical programming approach for the TFMRP, which utilizes several methodologies, for the problem of minimizing delay costs. In order to address the high dimensionality, we begin by presenting an aggregate model, in which the problem is formulated as a dynamic, multicommodity, integer network flow problem with certain side constraints. Using Lagrangian relaxation, we generate aggregate flows that are decomposed into a collection of flight paths for individual aircraft using a randomized rounding heuristic. This collection of paths is then used in a packing integer programming formulation, the solution of which generates feasible and near-optimal routes for individual flights. The overall algorithm, termed the Lagrangian Generation Algorithm, is used to solve real problems in the southwestern portion of United States. The solutions returned by our algorithm for practical problems are within 1% of the corresponding lower bounds.

Currently, the FAA implements a national ground-holding policy. This policy consists of a two-stage process. First, the FAA determines the arrival slots that each airline will receive using a first-come, first-serve rule. Next, each airline decides which of their scheduled flights to assign to each arrival slot. From this, the FAA calculates the amount of ground holding that each flight will incur. In the last decade, several models have been developed that use optimization techniques to improve upon the current practices for selecting ground holds. The FAA has investigated the possibility of implementing these techniques. In the next section, we will discuss how the TFMP and the TFMRP fit into the framework of previously addressed problems.

1.3 Literature Review

In Odoni (1987), the problem of scheduling flights in real time in order to minimize congestion costs was first conceptualized and introduced. Since then several models have been proposed for solving different versions of this problem. The first and simplest version considers a single airport and makes decisions about the ground-holds for this *Single-Airport Problem (SAGHP)*. The *Multi-Airport Ground-Holding Problem (MAGHP)* was the next problem to be addressed. It makes ground-holding decisions for an entire network of airports. Thus, the SAGHP and the MAGHP are distinguished by whether delays are assumed to propagate in the network of airports as aircraft perform consecutive flights.

As discussed above, the *Air Traffic Flow Management Problem (TFMP)* further determines the optimal airspeed adjustment of aircraft for a network of airports taking into account the capacitated airspace. For the *Air Traffic Flow Management Rerouting Problem (TFMRP)*, a flight may be rerouted through a different flight path in order to reach its destination if the current route passes through a region that has reduced capacity primarily due to poor weather conditions. In order to describe the work on these problems we consider the following modeling variations:

1. Deterministic versus stochastic models, which are distinguished by whether the capacities of the system (airports and sectors in the airspace) are assumed deterministic or probabilistic.
2. Static versus dynamic models, which are distinguished by whether or not the solutions are updated dynamically during the day.

The deterministic SAGHP (both static and dynamic) was first formulated as a network flow problem in Terrab and Odoni (1991). The stochastic SAGHP was formulated and solved as a stochastic programming problem in Richetta and Odoni (1993) (static case) and Richetta and Odoni (1994) (dynamic case). A review of optimization models for the SAGHP is given in Andreatta, Odoni and Richetta (1993).

The deterministic MAGHP was formulated as a 0-1 integer programming problem in Vranas, Bertsimas and Odoni (1994a) (static case) and in Vranas, Bertsimas and Odoni (1994b) (dynamic case). In this thesis and in Andreatta and Tidona (1994), new formulations for the MAGHP are proposed. These three models are computationally compared in Andreatta and Brunetta (1995) which concludes that our model performs the best computationally. We will further prove in this thesis that our model is the strongest formulation of MAGHP. A heuristic approach to solving the MAGHP is given in Brunetta, Guastalla, and Navazio (1996). Terrab and Paulose (1993) address the stochastic MAGHP as a stochastic programming problem. The papers by Matos and Ormerod (1995) and by Vranas (1995) discuss a problem in the European network. These papers address the “slot allocation” problem. Vranas (1995) shows that this is equivalent to the MAGHP in the case where congestion may only arise at the destination airport. Ball (1993) and Milner (1995) address the problem of banking flights and Gilbo (1993) addresses the problem of dependent arrival and departure runways.

In this thesis, we present a 0-1 integer programming model for the deterministic, multi-airport TFMP which addresses capacity restrictions on the en route airspace. Simultaneously with our work, models addressing enroute capacities were also introduced by Lindsay, Boyd and Burlingame (1993). They propose integer programming formulations for a version of TFMP that tracks a flight as it passes from fix to fix in the airspace. The fixes are points in the airspace, not regions of airspace. So the flights only experience capacity restrictions at airports and at fixes. As the linear programming relaxations of these formulations are not very strong, branch and bound is needed to generate integral solutions. However, by developing a wide array of novel formulation strengthening techniques, the dependence on “pure” branch and bound, as well as the computation times, are actually reduced. Helme (1994) has presented a method for the TFMP by designing a multicommodity minimum cost flow model over a network in space-time. This method has not as yet been fully tested, but it is expected that there will be severe dimensionality problems.

The problem of dynamically rerouting aircraft has not been addressed to the best of our knowledge in the literature. The backbone of our approach is the dynamic network flow formulation. Ford and Fulkerson (1958) first introduced a dynamic maximum flow problem as a standard network generalized to include traversal times between nodes. There are algorithms for problems that work with the dynamic network directly. Ford and Fulkerson (1958) present an algorithm that solves the dynamic maximum flow problem. Wilkinson (1971) and Minieka (1973, 1974) both present an exponential time algorithm to solve the universally maximum dynamic flow problem. In addition to these early papers, there has recently been considerable research activity on theoretical approaches to dynamic network flow problems. Hoppe and Tardos (1994, 1995a, 1995b) have described several polynomial time algorithms for discrete dynamic network problems including approximate universally maximum dynamic flows, lexicographically maximum flows and dynamic transshipment. Fleischer and Tardos (1996) furthered these algorithms by looking at the analogous continuous-time problems. For a complete review of work done on dynamic network flows see the survey papers of Aronson (1989), Bookbinder and Sethi (1980) and Powell, Jaillet and Odoni (1995). These advances are not directly relevant for our problem as our formulation is both multicommodity, integer and involves complicating side constraints.

Chapter 2

The Air Traffic Flow Management Problem

In this chapter, we discuss the air traffic flow management problem. In Section 2.1, we present an IP formulation for the traffic flow management problem and examine the size of the formulation. In Section 2.2, we make the connection with the ground-holding problem, by showing how the TFMP formulation can be reduced to model the MAGHP. Furthermore, the resulting MAGHP formulation is compared with others proposed in the literature. In Section 2.3, we provide some analysis of the polyhedral structure of the linear relaxation of our IP formulation of the TFMP. In Section 2.4, the TFMP is proven to be NP-hard. In Section 2.5, we extend the TFMP formulation to take into account several variations of the model, which incorporate the dependence between arrival and departure capacities, hub connectivity, banks of flights and rerouting of aircraft.

2.1 The 0-1 IP Formulation

The national airspace is divided into sectors. A map of the United States that displays all of the high level sector boundaries is given in Figure 2-1. Each flight passes through contiguous sectors while it is en route to its destination. There is a restriction on the

number of airplanes that may fly within a sector at a given time. This number is dependent on the number of aircraft that an air traffic controller can manage at one time, the geographic location and the weather conditions. We will refer to the restrictions on the number of aircraft in a given sector at a given time as the en route sector capacities.

Consider a set of flights, $\mathcal{F} = \{1, \dots, F\}$, a set of airports, $\mathcal{K} = \{1, \dots, K\}$, a set of time periods, $\mathcal{T} = \{1, \dots, T\}$, and a set of pairs of flights that are continued, $\mathcal{C} = \{(f', f) : f' \text{ is continued by flight } f\}$. We shall refer to any particular time period t as the “time t .” The problem input data are given as follows:

Data:

N_f = number of sectors in flight f 's path,

$$P(f, i) = \begin{cases} \text{the departure airport, if } i = 1, \\ \text{the } (i - 1)^{\text{st}} \text{ sector in flight } f\text{'s path, if } 1 < i < N_f, \\ \text{the arrival airport, if } i = N_f, \end{cases}$$

$P_f = (P(f, i) : 1 \leq i \leq N_f),$

$D_k(t)$ = departure capacity of airport k at time t ,

$A_k(t)$ = arrival capacity of airport k at time t ,

$S_j(t)$ = capacity of sector j at time t ,

d_f = scheduled departure time of flight f ,

r_f = scheduled arrival time of flight f ,

s_f = turnaround time of an airplane after flight f ,

c_f^g = cost of holding flight f on the ground for one unit of time,

c_f^a = cost of holding flight f in the air for one unit of time,

$l_{f,j}$ = number of time units that flight f must spend in sector j ,

T_f^j = set of feasible times for flight f to arrive to sector $j = [\underline{T}_f^j, \overline{T}_f^j],$

\underline{T}_f^j = first time period in the set $T_f^j,$

\overline{T}_f^j = last time period in the set $T_f^j.$

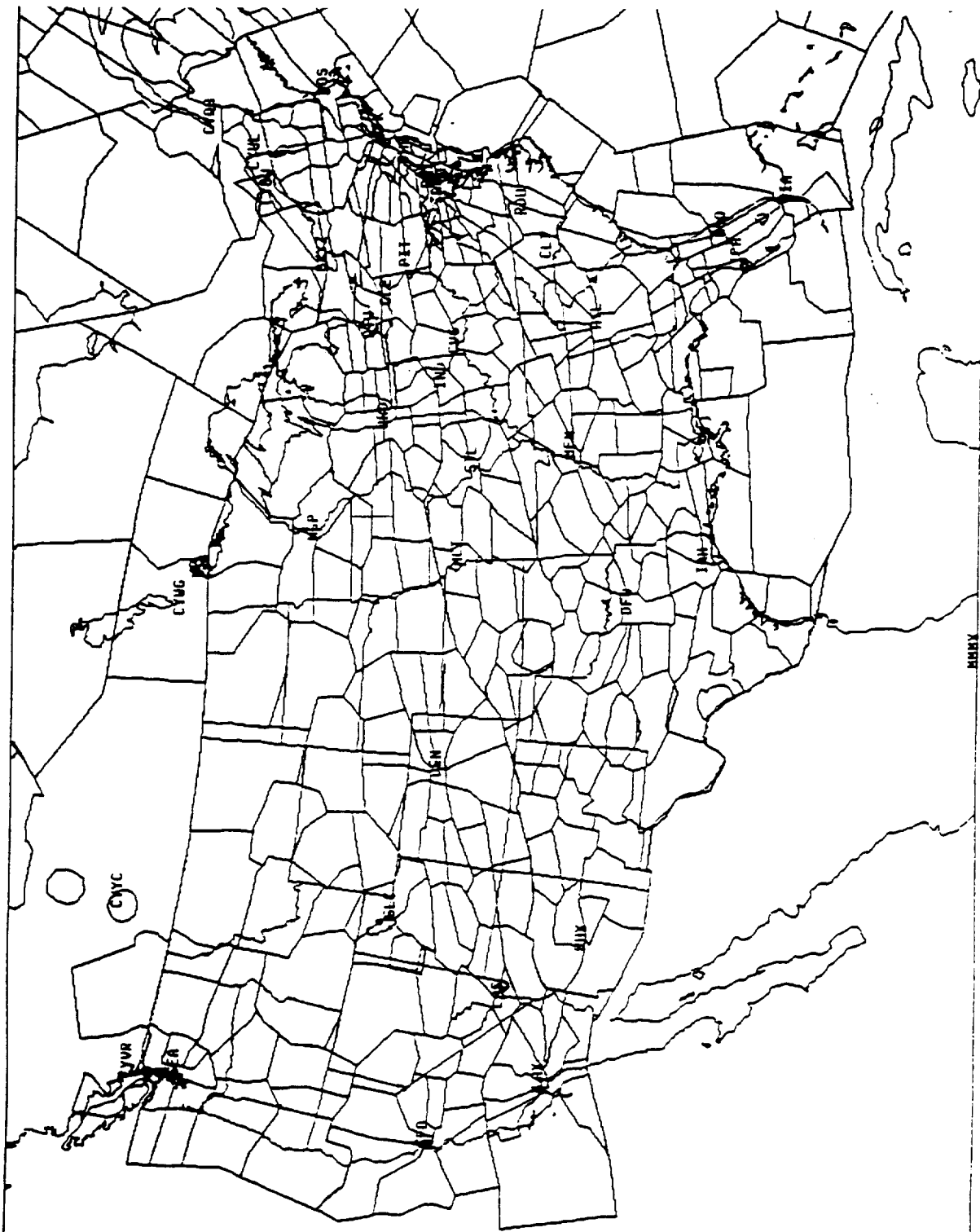


Figure 2-1: US Map with sector regions.

Note that by “flight”, we mean a flight leg between two airports. Also, flights referred to as “continued” are those flights whose aircraft relies on an aircraft that has just completed a previous flight.

Objective: The objective in the TFMP is to decide how much each flight is going to be held on the ground and in the air in order to minimize the total delay cost.

We model the problem as follows:

Decision variables:

$$w_{f,t}^j = \begin{cases} 1, & \text{if flight } f \text{ arrives at sector } j \text{ by time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Note that the $w_{f,t}^j$ are defined as being 1 if flight f arrives at sector j by time t . This definition using *by* and not *at* is critical to the understanding of the formulation. Also recall that we have also defined for each flight a list P_f including the departure airport, the pertinent sectors and the arrival airport, so that the variable $w_{f,t}^j$ will only be defined for those elements j in the list P_f . Moreover, we have defined T_f^j as the set of feasible times for flight f to arrive to sector j , so that the variable $w_{f,t}^j$ will only be defined for those times within T_f^j . Thus, in the formulation whenever the variable $w_{f,t}^j$ is used, it is assumed that this is a feasible (f, j, t) combination. Furthermore, one variable per flight-sector pair can be eliminated from the formulation by setting $w_{f, \bar{T}_f^j}^j = 1$. Since flight f has to arrive at sector j by the last possible time in its time window, we can simply set it equal to one as a parameter before solving the problem.

To ensure the clarity of the model, consider the following example pictured in Figure 2-2 which depicts two flights traversing a set of sectors. In this example, there are two flights, 1 and 2, each with the following associated data:

$$P_1 = (1, A, C, D, E, 4) \text{ and } P_2 = (2, F, E, D, B, 3).$$

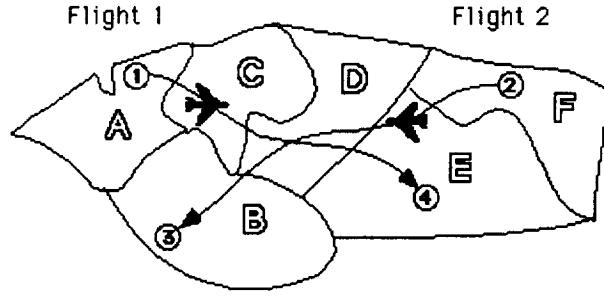


Figure 2-2: Two possible flight routes.

If we consider the current position of the aircraft at time t , indicated by the position of the airplane icons in Figure 2-2, then the decision variables for these flights at this time are given by:

$$w_{1,t}^1 = 1, w_{1,t}^A = 1, w_{1,t}^C = 1, w_{1,t}^D = 0, w_{1,t}^E = 0, w_{1,t}^4 = 0, \text{ and}$$

$$w_{2,t}^2 = 1, w_{2,t}^F = 1, w_{2,t}^E = 1, w_{2,t}^D = 0, w_{2,t}^B = 0, w_{2,t}^3 = 0.$$

Having defined the variables $w_{f,t}^j$, we can express several quantities of interest as linear functions of these variables as follows:

1. The variable $u_{f,t}^j = 1$ if flight f arrives at sector j at time t and 0 otherwise, can be expressed as follows:

$$u_{f,t}^j = w_{f,t}^j - w_{f,t-1}^j \text{ and vice versa, } w_{f,t}^j = \sum_{\{t' \leq t\}} u_{f,t'}^j. \quad (2.1)$$

As expressed earlier, the variables $w_{f,t}^j$ are only defined in the time range T_f^j , so that $w_{f,(T_f^j-1)}^j = 0$. Furthermore, the constraint that a flight must arrive at sector j at some time t , originally expressed by the restriction $\sum_{\{t \in T_f^j\}} u_{f,t}^j = 1$ can now be replaced by the simpler expression $w_{f,T_f^j}^j = 1$. As previously mentioned, this

can be handled as a parameter before the problem is solved, thus eliminating many variables and constraints. This substitution is fundamental to the performance of this model.

2. Noticing that the first sector for every flight represents the departing airport, the total number of time units that flight f is held on the ground can be expressed as the actual departure time minus the scheduled departure time, i.e.,

$$g_f = \sum_{\substack{\{t \in T_f^k, \\ k=P(f,1)\}}} tu_{f,t}^k - d_f = \sum_{\substack{\{t \in T_f^k, \\ k=P(f,1)\}}} t(w_{f,t}^k - w_{f,t-1}^k) - d_f.$$

3. Noticing that the last sector for every flight represents the destination airport, the total number of time units that flight f is held in the air can be expressed as the actual arrival time minus the scheduled arrival time minus the amount of time that the flight has been held on the ground, i.e.,

$$a_f = \sum_{\substack{\{t \in T_f^k, \\ k=P(f,N_f)\}}} tu_{f,t}^k - r_f - g_f = \sum_{\substack{\{t \in T_f^k, \\ k=P(f,N_f)\}}} t(w_{f,t}^k - w_{f,t-1}^k) - r_f - g_f.$$

The objective function

The objective of the formulation is to minimize total delay cost. Using the variables g_f and a_f for the amounts of ground and air delay respectively, as defined in items 2. and 3. above, the objective function can be expressed simply as follows:

$$\text{Min} \sum_{\{f \in \mathcal{F}\}} [c_f^g g_f + c_f^a a_f].$$

Substituting the expressions we derived in items 2 and 3 above for the variables $w_{f,t}^j$ we obtain the following expression:

$$\text{Min} \sum_{\{f \in \mathcal{F}\}} \left[c_f^g \left(\sum_{\substack{t \in T_f^k, \\ k=P(f,1)}} t (w_{f,t}^k - w_{f,t-1}^k) - d_f \right) + c_f^a \left(\sum_{\substack{t \in T_f^k, \\ k=P(f,N_f)}} t (w_{f,t}^k - w_{f,t-1}^k) - \right. \right. \\ \left. \left. r_f - \left(\sum_{\substack{t \in T_f^k, \\ k=P(f,1)}} t (w_{f,t}^k - w_{f,t-1}^k) - d_f \right) \right) \right]$$

Rearranging variables, we can now present the objective function along with the complete formulation.

(TFMP)

$$IZ_{TFMP} = \text{Min} \sum_{\{f \in \mathcal{F}\}} \left[(c_f^g - c_f^a) \sum_{\substack{t \in T_f^k, \\ k=P(f,1)}} t (w_{f,t}^k - w_{f,t-1}^k) + \right. \\ \left. c_f^a \sum_{\substack{t \in T_f^k, \\ k=P(f,N_f)}} t (w_{f,t}^k - w_{f,t-1}^k) + (c_f^a - c_f^g) d_f - c_f^a r_f \right]$$

subject to

$$\sum_{\{f: P(f,1)=k\}} (w_{f,t}^k - w_{f,t-1}^k) \leq D_k(t), \quad \forall k \in \mathcal{K}, t \in \mathcal{T}, \quad (2.2)$$

$$\sum_{\{f: P(f,N_f)=k\}} (w_{f,t}^k - w_{f,t-1}^k) \leq A_k(t), \quad \forall k \in \mathcal{K}, t \in \mathcal{T}, \quad (2.3)$$

$$\sum_{\substack{\{f: P(f,i)=j, \\ P(f,i+1)=j', i < N_f\}}} (w_{f,t}^j - w_{f,t}^{j'}) \leq S_j(t), \quad \forall j \in \mathcal{J}, t \in \mathcal{T}, \quad (2.4)$$

$$w_{f,t+l_f,j}^{j'} - w_{f,t}^j \leq 0, \quad \begin{cases} \forall f \in \mathcal{F}, t \in T_f^j, j = P(f, i), \\ j' = P(f, i + 1), i < N_f, \end{cases} \quad (2.5)$$

$$w_{f,t}^k - w_{f',t-s_{f'}}^k \leq 0, \quad \begin{cases} \forall (f', f) \in \mathcal{C}, t \in T_f^k, \\ k = P(f, 1) = P(f', N_f), \end{cases} \quad (2.6)$$

$$w_{f,t}^j - w_{f,t-1}^j \geq 0, \quad \forall f \in \mathcal{F}, j \in P_f, t \in T_f^j, \quad (2.7)$$

$$w_{f,t}^j \in \{0, 1\}, \quad \forall f \in \mathcal{F}, j \in P_f, t \in T_f^j.$$

The first three constraints take into account the capacities of various aspects of the system. The first constraint ensures that the number of flights which may take off from airport k at time t , will not exceed the departure capacity of airport k at time t . Likewise, the second constraint ensures that the number of flights which may arrive at airport k at time t , will not exceed the arrival capacity of airport k at time t . In each case, the difference will be equal to one only when the first term is one and the second term is zero. Thus, the differences capture the time at which a flight uses a given airport. The third constraint ensures that the sum of all flights which may feasibly be in sector j at time t will not exceed the capacity of sector j at time t . This difference gives the flights which are in sector j at time t , since the first term will be 1 if flight f has arrived in sector j by time t and the second term will be 1 if flight f has arrived at the next sector by time t . So, the only flights which will contribute a value of 1 to this sum are those flights that have arrived at j and have not yet departed from j by time t .

Constraints (2.5) represent connectivity between sectors. They stipulate that if a flight arrives at sector j' by time $t + l_{f,j}$, then it must have arrived at sector j by time t where j and j' are contiguous sectors in flight f 's path. In other words, a flight cannot enter the next sector on its path until it has spent $l_{f,j}$ time units (the minimum possible) traveling through sector j , the current sector in its path.

Constraints (2.6) represent connectivity between airports. They handle the cases in which a flight is continued, i.e., the flight's aircraft is scheduled to perform a later flight within some time interval. We will call the first flight f' and the following flight f .

Constraints (2.6) state that if flight f departs from airport k by time t , then flight f' must have arrived at airport k by time $t - s_{f'}$. The turnaround time, $s_{f'}$, takes into account the time that is needed to clean, refuel, unload and load, and further prepare the aircraft for the next flight. In other words, flight f cannot depart from airport k , until flight f' has arrived and spent at least $s_{f'}$ time units at airport k .

Constraints (2.7) represent connectivity in time. Thus, if a flight has arrived by time t , then $w_{f,t}^j$ has to have a value of 1 for all later time periods, $t' \geq t$.

Important Remark

The major reason we used the variables $w_{f,t}^j$, as opposed to the variables $u_{f,t}^j$ is that the former variables nicely capture the three types of connectivity in TFMP: connectivity between sectors, connectivity between airports and connectivity in time. Of course, given that the two sets of variables are linearly related, the same constraints can be captured using the $u_{f,t}^j$ variables. We feel, however, that the variables $w_{f,t}^j$ not only take connectivity naturally into account, but also they define connectivity constraints that are facets of the convex hull of solutions (see Section 2.3). As we report in Section 3.2, the LP relaxation of (TFMP) is almost always integral, i.e., the given formulation is a particularly strong one. We believe that the key for this is the use of the decision variables $w_{f,t}^j$ in the formulation.

Size of the Formulation

Let

$$D = \max_{\{f \in \mathcal{F}, j \in P_f\}} |T_f^j|,$$

be the maximum cardinality of the set of feasible times for flight f to be in sector j taken over all f and j , and let

$$X = \max_{\{f \in \mathcal{F}\}} N_f$$

be the maximum number of sectors that a flight passes through along its route, taken over all flights. Note that $X \geq 2$, since the departure and arrival airports are always counted as sectors on a flight's path. Let $|\mathcal{F}|$ be the total number of flights, $|\mathcal{T}|$ be

the total number of time periods, $|\mathcal{K}|$ be the total number of airports, $|\mathcal{J}|$ be the total number of sectors, and $|\mathcal{C}|$ be the total number of flights that are continued.

The actual number of variables $w_{f,t}^j$ is

$$\sum_f \sum_{\{j \in P_f\}} |T_f^j|$$

since each flight has a different number of sectors and number of feasible time intervals associated with it. An upper bound on the number of variables $w_{f,t}^j$ will be

$$|\mathcal{F}| D X.$$

The exact number of constraints is

$$2|\mathcal{K}||\mathcal{T}| + |\mathcal{J}||\mathcal{T}| + 2 \sum_{\{f \in \mathcal{F}\}} \sum_{\{j \in P_f\}} |T_f^j| + \sum_{\substack{\{(f',f) \in \mathcal{C}, \\ a=P(f,1), \\ a=P(f',N_{f'})\}}} \min\{|T_f^a|, |T_{f'}^a|\}$$

An upper bound on the number of constraints can then be calculated as

$$2|\mathcal{K}||\mathcal{T}| + |\mathcal{J}||\mathcal{T}| + 2|\mathcal{F}|DX + |\mathcal{C}|D.$$

In order to get a feeling of the size of the formulation let us consider an example that adequately represents the U.S. network:

1. $\mathcal{K} = 20$ representing the most congested airports in the U.S.
2. $|\mathcal{T}| = 14 * 12 = 168$, representing a 14 hour day with 5 minute intervals.
3. $|\mathcal{J}| = 200$, representing 200 sectors.
4. $|\mathcal{F}| = 10000$, representing approximately half of the number of daily flights of major carriers.

5. $|\mathcal{C}| = 8000$, representing an 80% connectivity among flights.
6. $D = 6$, representing an upper bound of half an hour that a flight can be late to any given sector.
7. $X = 5$, representing an upper bound of at most 5 sectors in a flight's path.

For this example the number of variables is at most 300,000 and the number of constraints is at most 688,320. The critical quantities that significantly affect the number of variables and constraints are D , X , and $|\mathcal{F}|$. If for example any of these parameters doubles, the number of variables doubles and the number of constraints nearly doubles.

2.2 The Multi-Airport Ground-Holding Problem as a Special Case

As mentioned in Chapter 1, the ground-holding problem is a special case of the TFMP. If we remove the sector capacity constraints and the variables associated with the sectors, we obtain a new formulation of the MAGHP which, as we demonstrate in Chapter 3.2, leads to significant computational advantages compared to alternative formulations that have previously been proposed (see Section 1.3). Notice that $N_f = 2$ for all $f \in \mathcal{F}$, since a flight's path consists solely of the departure and arrival airports.

Let us redefine the variables as:

$$y_{f,t} = w_{f,t}^k, \text{ for the departure airport, } k = P(f, 1),$$

$$z_{f,t} = w_{f,t}^k, \text{ for the arrival airport, } k = P(f, 2).$$

Also, let T_f^d be the set of feasible departure times for flight f and let T_f^a be the set of feasible arrival times for flight f .

Using the new variables, the formulation (*TFMP*) specializes to the following new formulation of (*MAGHP*):

(*MAGHP*)

$$\begin{aligned}
IZ_{MAGHP} = \text{Min} \sum_{\{f \in \mathcal{F}\}} & \left[(c_f^g - c_f^a) \sum_{\{t \in T_f^d\}} t(y_{f,t} - y_{f,t-1}) \right. \\
& \left. + c_f^a \sum_{\{t \in T_f^a\}} t(z_{f,t} - z_{f,t-1}) + (c_f^a - c_f^g)d_f - c_f^a r_f \right]
\end{aligned}$$

$$\text{subject to} \quad \sum_{\{f: t \in T_f^d\}} (y_{f,t} - y_{f,t-1}) \leq D_k(t), \quad \forall k \in \mathcal{K}, t \in \mathcal{T}, \quad (2.8)$$

$$\sum_{\{f: t \in T_f^a\}} (z_{f,t} - z_{f,t-1}) \leq A_k(t), \quad \forall k \in \mathcal{K}, t \in \mathcal{T}, \quad (2.9)$$

$$z_{f,t} - y_{f,t-(r_f-d_f)} \leq 0, \quad \forall f \in \mathcal{F}, t \in T_f^a, \quad (2.10)$$

$$y_{f,t} - z_{f',t-s_{f'}} \leq 0, \quad \forall (f', f) \in \mathcal{C}, t \in T_f^d, \quad (2.11)$$

$$y_{f,t} - y_{f,t-1} \geq 0, \quad \forall f \in \mathcal{F}, t \in T_f^d, \quad (2.12)$$

$$z_{f,t} - z_{f,t-1} \geq 0, \quad \forall f \in \mathcal{F}, t \in T_f^a, \quad (2.13)$$

$$y_{f,t}, z_{f,t} \in \{0, 1\}, \quad \forall f \in \mathcal{F}, t \in \mathcal{T}.$$

The first two constraints incorporate the capacity restrictions of the departure and arrival airports. The next constraint is the sector connectivity constraint, which is equivalent to constraint (2.5) in the TFMP formulation. However, for the ground holding problem the only elements in the path are the departure airport and the arrival airport. So this constraint connects these two elements by making sure that flight f cannot arrive at time t unless it has departed by at least t minus the minimum flight time. The next constraint is the flight connectivity constraint, which is equivalent to constraint (2.6)

in the TFMP formulation. The last two constraints are time connectivity constraints, which are equivalent to constraint (2.7) in the formulation (*TFMP*).

Using the previous definitions, an upper bound on the number of variables is $2|\mathcal{F}|D$ and an upper bound on the number of constraints is $2|\mathcal{K}||\mathcal{T}| + 3|\mathcal{F}|D + |\mathcal{C}|D$. For the same example as in the end of the previous subsection, an upper bound on the number of variables in the above formulation is 120,000 and an upper bound on the number of constraints is 234,720.

If we remove the constraint (2.11) and consider the set \mathcal{K} to be the singleton set, then we have a valid formulation for SAGHP, which we will call (*SAGHP*). We define the feasible regions for the formulations (*TFMP*), (*MAGHP*), and (*SAGHP*) as IP_{TFMP} , IP_{MAGHP} and IP_{SAGHP} respectively.

The variables used in the formulation in Vranas et al. (1994a) are defined differently: $u_{f,t} = 1$ if flight f takes off at time t and $v_{f,t} = 1$ if flight f arrives at time t . These are linearly related to variables $y_{f,t}$ and $z_{f,t}$ as per the relationship given by (2.1). As already mentioned, the ground-holding delays can be expressed in terms of these variables in the following manner:

$$g_f = \sum_{\{t \in T_f^d\}} t u_{f,t} - d_f,$$

as can the airholding delay,

$$a_f = \sum_{\{t \in T_f^a\}} t v_{f,t} - r_f - g_f.$$

In Vranas et al. (1994a), it is assumed that when the departure capacity is large, without loss of generality, $a_f = 0$, thus implying that all of the delay would be taken on the ground before departure. This gives an equivalent expression for g_f as, $g_f = \sum_{t \in T_f^d} t v_{f,t} - r_f$, which contains no departure information, thus eliminating the variables $u_{f,t}$ from the formulation. Moreover, instead of the flight connectivity constraints (2.11), the following constraints,

$$g_{f'} - (d_f - s_{f'} - r_{f'}) \leq g_f, \tag{2.14}$$

establish connectivity between the arriving flight f' and the departing flight f by forcing the amount of ground-hold for flight f to be at least the amount that flight f' arrives late, $g_{f'}$, minus the amount of slack time, $d_f - s_{f'} - r_{f'}$. The description of the feasible space in Vranas et al. (1994a) expressed in the $z_{f,t}$ space as per the relationship (2.1) is as follows:

$$IP_{VBO} = \left\{ z_{f,t} \in \{0, 1\} \mid \sum_f (z_{f,t} - z_{f,t-1}) \leq A_k(t), \sum_{\{t \in T_f^a\}} (z_{f,t} - z_{f,t-1}) = 1, \right. \\ \left. g_f = \sum_{\{t \in T_f^a\}} t(z_{f,t} - z_{f,t-1}) - r_f, g_{f'} - (d_f - s_{f'} - r_{f'}) \leq g_f, z_{f,t} - z_{f,t-1} \geq 0 \right\}.$$

Terrab and Paulose (1993) use the same variables, v_{ft} as in Vranas et al. (1994a). However, they express the flight connectivity constraints as follows:

$$\sum_{\{t \in T_f^a, t \leq \tau\}} v_{f,t} - \sum_{\substack{\{t' \in T_{f'}^a, \\ t' \leq \tau - s_{f'} - (r_f - d_f)\}}} v_{f',t'} \leq 0. \quad (2.15)$$

Constraint (2.15) forces connectivity, since if the second sum is zero then flight f' has not landed by time $\tau - s_{f'} - (r_f - d_f)$, which is time period τ minus the turnaround time, minus the flight time of f . This forces the first sum to be zero so that flight f can not land before time τ . The description of their formulation expressed in the $z_{f,t}$ space as per the relationship (2.1) is:

$$IP_{TP} = \left\{ z_{f,t} \in \{0, 1\} \mid \sum_f (z_{f,t} - z_{f,t-1}) \leq A_k(t), \sum_{\{t \in T_f^a\}} (z_{f,t} - z_{f,t-1}) = 1, \right. \\ \left. \sum_{\{t \in T_f^a, t \leq \tau\}} (z_{f,t} - z_{f,t-1}) - \sum_{\substack{\{t' \in T_{f'}^a, \\ t' \leq \tau - s_{f'} - (r_f - d_f)\}}} (z_{f',t'} - z_{f',t'-1}) \leq 0, z_{f,t} - z_{f,t-1} \geq 0 \right\}.$$

If we specialize our formulation for the case of large departure capacities and use only the variables, $z_{f,t}$ ($y_{f,t} = z_{f,t+(r_f-d_f)}$), we obtain:

$$IP'_{MAGHP} = \left\{ z_{f,t} \in \{0, 1\} \left| \begin{array}{l} \sum_{\{f:t \in T_f^a\}} (z_{f,t} - z_{f,t-1}) \leq A_k(t), \sum_{\{t \in T_f^a\}} (z_{f,t} - z_{f,t-1}) = 1, \\ z_{f,t+(\tau_f-d_f)} - z_{f',t-s_{f'}} \leq 0, z_{f,t} - z_{f,t-1} \geq 0 \end{array} \right. \right\}.$$

In all of these formulations, the expression $\sum_{t \in T_f^a} (z_{f,t} - z_{f,t-1}) = 1$ reduces to the expression $z_{f, \bar{T}_f} = 1$. This telescoping property is due to the unique definition of the decision variables as flights arriving by some time t rather than at time t .

If we denote the polyhedra corresponding to the linear programming relaxations of IP'_{MAGHP} , IP_{VBO} , and IP_{TP} as P'_{MAGHP} , P_{VBO} , and P_{TP} and denote their corresponding values as Z'_{MAGHP} , Z_{VBO} , and Z_{TP} , then we can state the following proposition whose proof is included in Appendix A.

Proposition 1 $IP_{TP} = IP_{VBO} = IP'_{MAGHP} \subseteq P'_{MAGHP} \subseteq P_{TP} \subseteq P_{VBO}$ and correspondingly, $Z_{VBO} \leq Z_{TP} \leq Z'_{MAGHP} \leq IZ'_{MAGHP} = IZ_{VBO} = IZ_{TP}$.

Therefore, the LP relaxation of (*MAGHP*) gives bounds that are at least as strong as those from the LP relaxations of either Vranas et al. (1994a) or Terrab and Paulose (1993).

2.3 Insights from the Polyhedral Structure

In Section 3.2, we report computational results for the TFMP based on the formulation (*TFMP*). Even for large scale problems and for a variety of problem parameters the solutions of the LP relaxation of both (*TFMP*) and (*MAGHP*) were integral. In the tradition of polyhedral combinatorics in mathematical programming, we examine the polyhedral structure of P_{TFMP} and P_{MAGHP} in order to obtain a deeper understanding of why this formulation performs so well computationally. Given a set S we denote with $conv(S)$ the convex hull of solutions in S . In particular, we address the following questions:

1. Are the polyhedra P_{TFMP} and P_{MAGHP} integral? If not, is the optimal solution to the optimization problem integral if we impose the simplification that $c_g = c_f^g$ and $c_a = c_f^a$ for all $f \in \mathcal{F}$?
2. Are the constraints in $(TFMP)$ and $(MAGHP)$ facets of $\text{conv}(IP_{TFMP})$ and $\text{conv}(IP_{MAGHP})$ respectively?

We summarize our findings in the following theorem:

Theorem 2

1. *The polyhedra P_{TFMP} and P_{MAGHP} are not integral. Even with the simplification that $c_g = c_f^g$ and $c_a = c_f^a$ for all $f \in \mathcal{F}$, integral solutions are not obtained.*
2. *Inequalities (11), (12), (13) and (14) are facet defining for $\text{conv}(IP_{MAGHP})$, while the constraints (9) and (10) need not be. Inequalities (5), (6) and (7) are facet defining for $\text{conv}(IP_{TFMP})$, while the constraints (2), (3) and (4) need not be.*

As the proofs of the theorem are somewhat technical, we have placed them in appendices B, and C respectively.

The previous theorem gives some partial insight on the usefulness of the new variables we introduced, which make it easy to express sharply the various types of connectivity in the problem. While the formulations are not integral, the inequalities that the three types of connectivity impose are indeed facet defining. As the solutions obtained were integral for a wide spectrum of examples and parameters, we did not investigate further the determination of other facets.

2.4 Complexity of the TFMP

In this section we show that the TFMP is an NP-hard problem.

Theorem 3 *The TFMP with all capacities equal to 1 is NP-hard.*

Proof. We show that job-shop scheduling (see Garey and Johnson (1979)) reduces to TFMP.

JOB SHOP SCHEDULING PROBLEM (JSP)

INSTANCE: Number $m \in Z^+$ of processors, set J of jobs, each $j \in J$ consisting of an ordered collection of tasks $t_k[j]$, $1 \leq k \leq n_j$, for each task t a length $l(t) \in Z_0^+$ and a processor $p(t) \in \{1, 2, \dots, m\}$, where $p(t_k[j]) \neq p(t_{k+1}[j])$ for all $j \in J$ and $1 \leq k < n_j$, and a deadline $D \in Z^+$.

QUESTION: Is there a job-shop schedule for J that meets the overall deadline, i.e., a collection of one-processor schedules σ_i mapping $\{t : p(t) = i\}$ into Z_0^+ , $1 \leq i \leq m$, such that

$$\begin{aligned} \sigma_i(t) > \sigma_i(t') & \quad \text{implies } \sigma_i(t) \geq \sigma_i(t') + l(t), \\ \sigma_{i'}(t_{k+1}[j]) \geq \sigma_i(t_k[j]) + l(t_k[j]), & \quad \text{where } i' = p(t_{k+1}[j]), i = p(t_k[j]) \\ & \quad \text{for all } j \in J, k \in \{1, \dots, n_j\}, \\ \text{and } \sigma_i(t_{n_j}[j]) + l(t_{n_j}[j]) \leq D, & \quad \text{where } i = p(t_{n_j}[j]) \text{ for all } j \in J. \end{aligned}$$

For each job we create an aircraft. For each processor we associate an airport or sector. Task $t_k[j]$ of job j corresponds to a flight segment, $f_k[j]$ of aircraft j . Given a collection of tasks, $t_k[j]$ of job j , we associate a list of airports and sectors to be visited by aircraft j . Furthermore, the processing time of task $t_k[j]$ corresponds to the time required to perform the flight segment, $f_k[j]$.

We obtain a list of airports and sectors,

$$(A_j^1, S_j^2, \dots, A_j^k, S_j^{(k+1)}, \dots, A_j^{n_j}),$$

and a list of the flight segment times,

$$(t_{A_j}^1, t_{S_j}^2, \dots, t_{A_j}^k, t_{S_j}^{k+1}, \dots, t_{A_j}^{n_j}),$$

for each aircraft j by the relationships:

$$\begin{aligned}
A_j^1 &= p(t_j[1]), & t_{A_j}^1 &= l(t_j[1]), \\
S_j^2 &= p(t_j[2]), & t_{S_j}^2 &= l(t_j[2]), \\
S_j^3 &= p(t_j[3]), & t_{S_j}^3 &= l(t_j[3]), \\
& & \vdots & \\
A_j^{n_j} &= p(t_j[n_j]), & t_{A_j}^{n_j} &= l(t_j[n_j]).
\end{aligned}$$

So by finding a job-shop schedule that satisfies the given conditions, we will find a solution to the transformed problem such that all flights are performed by the deadline D . Also, according to the relationship

$$\sigma_{i'}(t_{k+1}[j]) \geq \sigma_i(t_k[j]) + l(t_k[j]) \text{ where } i' = p(t_{k+1}[j]), i = p(t_k[j]),$$

no two tasks will ever be performed simultaneously on the same processor, which is equivalent to limiting the capacities of airports and sectors to one. Moreover, the relationship,

$$\sigma_i(t) > \sigma_i(t') \text{ implies } \sigma_i(t) \geq \sigma_i(t') + l(t),$$

dictates that a task can not be processed unless the previous task has completed.

This stipulation guarantees connectivity between flights, and sectors, as specified by the set of tasks for each aircraft. Thus, all of the constraints of the TFMP will be satisfied if and only if there exists a feasible job-shop schedule. ■

2.5 Modeling Variations

Our goal in this section is to demonstrate that the formulation (*TFMP*) can be easily extended in many directions to take into account several variations of the model.

2.5.1 Dependence Between Arrival and Departure Capacities

The interdependence between the arrival and departure capacities of airports results from the fact that the same runways are often used for both arrivals and departures. Thus, the runway allocation will determine how an airport's available capacity is allocated between the arrivals and departures at a given time. By operating under a specific runway configuration, arrival and departure capacities can be adjusted. This will significantly influence airport efficiency.

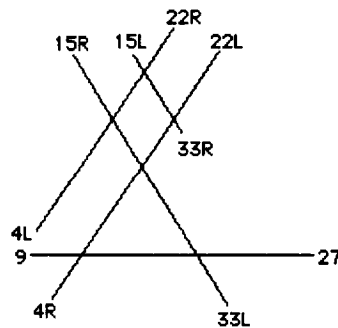


Figure 2-3: Complete Runway Configuration for Logan Airport.

By choosing a particular configuration of runways for a given time, the capacity allocation will be fixed. The complete set of runways for Logan Airport is given in Figure 2-3. A common configuration used at Logan Airport is to use runways 4L and 4R for arriving flights and use runways 9 and 4R for departing flights. Notice that since runway 4R is the longest runway and certain types of aircraft require a long runway, it is used for both arrivals and departures.

In Figure 2-4, we have shown a hypothetical runway allocation. Since it takes longer for an aircraft to arrive than to depart, if the airport is operating under the allocation shown in Figure 2-4 and all the capacity at Logan Airport is allocated to arrivals then 52 flights could arrive and if all the capacity is allocated to departures then 62 flights could

depart within an hour. If only less than 50 flights are allowed the depart within an hour, then the capacity allocation is given by the equation, $3 A_k(t) + D_k(t) = 3 * 52$. Whereas, if more than 50 flights are allowed to depart within an hour, the capacity allocation is given by the equation, $A_k(t) + 3 D_k(t) = 3 * 62$.

We review briefly ideas introduced in Gilbo (1993) and Vranas et al. (1994a). We represent the runway allocation by a set of linear constraints indexed by i for airport k at time t of the type

$$\alpha_{kt}^i D_k(t) + \beta_{kt}^i A_k(t) \leq \gamma_{kt}^i, \quad \forall k \in \mathcal{K}, t \in \mathcal{T}, i \in I_{kt}, \quad (2.16)$$

where $\alpha_{kt}^i, \beta_{kt}^i$, and γ_{kt}^i are given constants. In the example of Figure 2-4, there were two linear constraints. The set I_{kt} will determine the number of linear constraints for each airport k at time t .

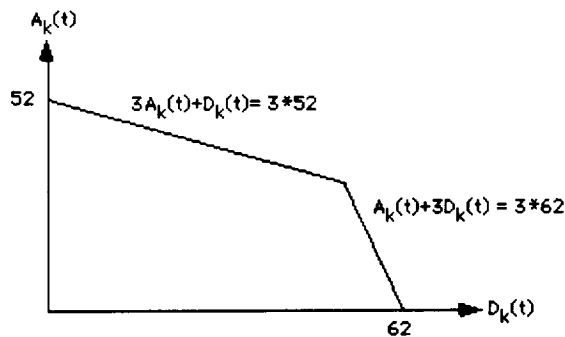


Figure 2-4: Runway Departure/Arrival Allocation for a Specific Configuration.

The region formed by the above constraints gives a complete depiction of all the possible runway allocations at a given time, and likewise, all possible departure and arrival capacity assignments.

In order to solve this variation, we treat $D_k(t)$ and $A_k(t)$ as variables that satisfy constraints (2.16) and add them to $(TFMP)$. We can further reduce the size of the

resulting formulation by eliminating the variables $D_k(t)$ and $A_k(t)$ by incorporating constraints (2.2) and (2.3) taken at equality into (2.16) as follows:

$$\alpha_{kt}^i \sum_{\{f:t \in T_f^k, k=P(f,1)\}} (w_{f,t}^k - w_{f,t-1}^k) + \beta_{kt}^i \sum_{\{f:t \in T_f^k, k=P(f,N_f)\}} (w_{f,t}^k - w_{f,t-1}^k) \leq \gamma_{kt}^i.$$

The addition of this constraint to $(TFMP)$ incorporates the dependence between the arrival and departure capacity assignments without the addition of any new variables.

2.5.2 Hub Connectivity with Multiple Connections

Given that many airlines now control key hub airports through which most of their flights are directed, it is no longer obvious which aircraft will fly a subsequent flight. At these hubs, many airplanes are capable of performing any one of multiple consecutive flights. We refer to the issue of assigning aircraft to continuing flights as hub connectivity. This can be achieved by extending the model as follows:

For each arriving flight f' that is continued there is a set of flights $R_{f'}$ that can continue flight f' . Introducing the 0 – 1 variables $x_{f',f}$, which take on the value 1 if flight f' is continued by flight $f \in R_{f'}$ and 0 otherwise, we alter constraint (2.6) as follows

$$\begin{aligned} w_{f,t}^k - w_{f',t-s_{f'}}^k &\leq 1 - x_{f',f}, & \forall (f', f) \in \mathcal{C}, t \in T_f^k, \\ & & k = P(f, 1) = P(f', N_{f'}), \end{aligned}$$

and add the constraint that each continued flight f' has to be assigned to a flight in $R_{f'}$:

$$\sum_{\{f \in R_{f'}\}} x_{f',f} = 1.$$

2.5.3 Banks of Flights

With the evolution of the hub and spoke system, airlines have a set of flights (banks) that are scheduled to arrive at a hub airport and another set scheduled to depart within a

small time window of the arrival bank. Each arriving aircraft will be assigned to perform at most one of the departing flights. This situation is similar to hub connectivity, except that the objective of hub connectivity is to minimize delay costs, while the objective of the airlines who rely heavily on banking flights is to minimize the spread between the arrival times of the first and the last flight in the bank. Let B be the set of flights in a bank. We define the decision variables

$$y_{B,t} = \begin{cases} 1 & \text{if the first flight } f \text{ in } B \text{ arrives by time } t \\ 0 & \text{otherwise} \end{cases}$$

$$z_{B,t} = \begin{cases} 1 & \text{if the last flight } f \text{ in } B \text{ arrives by time } t \\ 0 & \text{otherwise} \end{cases}$$

These definitions require the constraints:

$$\begin{aligned} y_{B,t} - w_{f,t}^k &\geq 0, & \forall f \in B, t \in T_f^k, k = P(f, N_f) \\ z_{B,t} - w_{f,t}^k &\leq 0, & \forall f \in B, t \in T_f^k, k = P(f, N_f). \end{aligned}$$

We also need the additional time connectivity constraints for these variables

$$\begin{aligned} y_{B,t} - y_{B,t-1} &\geq 0, & \forall t \in \mathcal{T} \\ z_{B,t} - z_{B,t-1} &\geq 0, & \forall t \in \mathcal{T}. \end{aligned}$$

The objective function of minimizing the “spread” in the arrival times for the flights in the bank B can be modeled as follows:

$$\min \sum_{\{t \in \mathcal{T}\}} t(z_{B,t} - z_{B,t-1}) - \sum_{\{t \in \mathcal{T}\}} t(y_{B,t} - y_{B,t-1}).$$

This is equivalent to determining

$$\min \sum_{\{t \in \mathcal{T}\}} \left(\max_{\substack{\{f \in B, \\ k = P(f, N_f)\}}} t(w_{f,t}^k - w_{f,t-1}^k) - \min_{\substack{\{f \in B, \\ k = P(f, N_f)\}}} t(w_{f,t}^k - w_{f,t-1}^k) \right).$$

With the addition of these new variables, new constraints, and new objective function, banking can be incorporated into the formulation. Alternative approaches to handle banking constraints are proposed in Ball (1993) and in Milner (1995).

2.5.4 Rerouting of Aircraft

As we already mentioned, very often extreme weather conditions force the capacities of some sectors (and airports) in the national airspace system to drop significantly or even to become zero. Air traffic controllers are then forced to use alternative routes for aircraft passing through these sectors to accommodate these changes in capacities (see Figure 4-1 for an example). Currently, these rerouting decisions are handled through the experience of the air traffic controllers and not through a formal optimization model. Rerouting will be discussed extensively in Chapters 4 and 5. We also note that the formulation (*TFMP*) can be extended to accommodate dynamic rerouting decisions. However, these extensions quickly lead to very large formulations, especially when considering realistic size problems.

There are two approaches that we may take in attempting to formulate the rerouting problem by extending the *TFMP* formulation. We have called these the path and sector approaches. The path approach assumes that each flight has a set of routes to choose from in deciding which is optimal. The sector approach makes no assumption about the routes, but rather selects that path as a series of possible sectors.

The path approach first defines Q_f as a set of possible routes that flight f may fly. In the formulation (*TFMP*) we have assumed that Q_f only contains one route, which we have denoted as P_f . In order for the formulation to be of manageable (but still large)

size we need to restrict the size of Q_f . We extend the *TFMP* variables in the following manner:

$$w_{f,t}^{j,r} = \begin{cases} 1, & \text{if flight } f \text{ arrives at sector } j \text{ by time } t \text{ along route } r, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, the variables $w_{f,t}^j$ defined in Section 2.1 can be written as:

$$w_{f,t}^j = \sum_{\{r \in Q_f\}} w_{f,t}^{j,r}.$$

Moreover, since the departure and arrival airports will remain the same for a given flight over all routes, $P(f, 1)$ and $P(f, N_f)$ will be independent of the particular route. Using the newly defined variables we can modify the *TFMP* to include rerouting with the extra restriction that only one path may be used per flight. The size of the resulting formulation will be at most a factor $\max_f |Q_f|$ larger than the *TFMP* formulation. This implies that we may be able to handle problems with a relatively small number of alternative paths.

The sector approach decides at each sector in its route which sector to enter next. We need to define $N(f, j)$, the set of sectors that flight f can enter immediately after exiting sector j , as well as $P(f, j)$, the set of sectors that flight f can enter immediately before entering sector j . We extend the *TFMP* variables in the following manner:

$$w_{f,t}^{j,j'} = \begin{cases} 1, & \text{if flight } f \text{ arrives at sector } j' \text{ from sector } j \text{ by time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, the variables $w_{f,t}^j$ defined in Section 2.1 can be written as:

$$w_{f,t}^j = \sum_{\{j' \in N(f,j)\}} w_{f,t}^{j',j}.$$

As before, the departure and arrival airports will remain the same for a given flight over

all routes; $P(f, 1)$ and $P(f, N_f)$ will be independent of the particular choice of sectors. Furthermore, by using the newly defined variables, we can modify the *TFMP* to include rerouting with the necessary constraints that each flight can only travel from one sector to one other sector.

Chapter 3

Computational Results for the Traffic Flow Management Problem

In this chapter we report the results of a series of computational experiments that we conducted. In Section 3.1, we provide computational results comparing our formulation of *MAGHP* with others in the literature using data from Vranas et al. (1994a). In Section 3.2, we report computational results of the traffic flow management problem using several datasets including some real data provided by the FAA.

In performing the computational experiments, we aim to address the following questions:

1. How frequently are the solutions obtained by solving the LP relaxations of (*TFMP*) and (*MAGHP*) integral?
2. How is the integrality of solutions affected by the various problem parameters and the size of the problem?
3. How is the computational time required to obtain an optimal solution affected by the various problem parameters and the size of the problem?
4. How does the present approach compare with other approaches in the literature?

5. Given that the TFMP needs to be solved on line for controlling air traffic in the US, perhaps the **most important question** to ask is: Can large problems of realistic size be solved in reasonable computational times? In other words, is the present approach a realistic method to control air traffic in the US?

3.1 Ground-Holding Problem Computations

We performed computational experiments on datasets used in Vranas et al. (1994a) on the Ground-Holding Problem. Specifically, we looked at the datasets consisting of 2 and 6 airports with 500 flights per airport, totalling 1000 and 3000 flights respectively. Some adjustments in the data were necessary in order to accommodate the differences between the two models. In particular, the previous model did not include of any departure data, as all of the optimization was done with respect to arrivals. Thus, we generated departure data (times and capacities) that were compatible with the existing arrival data.

As in Vranas et al. (1994a), for each of these cases, four levels of flight connectivity were considered. These levels give the ratios of continued flight to total flights, $|C|/|\mathcal{F}|$, as 0.20, 0.40, 0.60, and 0.80.

We considered 15 minute time intervals taken over a 16 hour day. All experiments were performed on a Sun SPARCstation 10 model 41. GAMS was used as the modeling tool and CPLEXMIP 2.1 was used as the solver. The results that we obtained using the above datasets and our (*MAGHP*) formulation are summarized in Tables 3.1 and 3.2.

$ \mathcal{F} $	$ C / \mathcal{F} $	Dep Capacity	Arr Capacity	Time	% Nonint
1000	0.20	32	15	262	0
1000	0.40	17	10	741	4
1000	0.60	20	14	359	0
1000	0.80	20	20	283	0

Table 3.1: Results at the infeasibility border for 1000 flights.

$ \mathcal{F} $	$ \mathcal{C} / \mathcal{F} $	Dep Capacity	Arr Capacity	Time	% Nonint
3000	0.20	20	20	5475	0
3000	0.40	20	20	4703	0
3000	0.60	20	20	5407	0
3000	0.80	20	20	9411	0

Table 3.2: Results at the infeasibility border for 3000 flights.

Tables 3.1 and 3.2 give results at the infeasibility border for each case. The infeasibility border is the set of critical values for the departure and arrival capacities, in units of flights per time interval, under which the problem becomes infeasible. We expect that it is in this region that the problem is very relevant practically and is harder to solve. The critical capacity levels were found by a series of trial and error tests. The times reported are in CPU seconds and the % Nonint column is the percentage of total flights whose solution was noninteger. If we compare these results with the results from Vranas et al. (1994a) (see Tables 3.3 and 3.4), we can see that the largest amount of improvement occurred in the integrality of the solutions. The computational times for solving our LP for 1000 flights (see Table 3.1; column Time) are comparable to the time required to solve their LP (see Table 3.3; column LP Time), while for the 3000 flights the LP in Vranas et al. (1994a) was solved faster. However, our solutions are for the most part already integral (the only instance where the solution was not integral was the 40% connectivity instance of the 1000 flight example). The total amount of time required to find an integral solution from the LP in Vranas et al. (1994a), found in the total time column, includes the time required to solve the LP relaxation, found in the LP Time column, plus the time required to perform a Branch & Bound heuristic. If we compare the amount of time required to find an integral solution, we see a significant improvement in computational time.

Tables 3.5 and 3.6 were constructed to demonstrate how computational time and in-

$ \mathcal{F} $	$ \mathcal{C} / \mathcal{F} $	Dep Capacity	Arr Capacity	LP Time	Total Time	% Nonint
1000	0.20	∞	(12,14)	258	374	6.3
1000	0.40	∞	10	327	894	8.4
1000	0.60	∞	11	377	6958	12.8
1000	0.80	∞	10	453	9512	16.8

Table 3.3: Previous Results at the infeasibility border for 1000 flights.

$ \mathcal{F} $	$ \mathcal{C} / \mathcal{F} $	Dep Capacity	Arr Capacity	LP Time	Total Time	% Nonint
3000	0.20	∞	12	1453	11360	not given
3000	0.40	∞	18	1808	13291	not given
3000	0.60	∞	17	2547	17980	not given
3000	0.80	∞	18	3072	25021	not given

Table 3.4: Previous Results at the infeasibility border for 3000 flights.

tegrality are affected by changes in the capacities, i.e., how well does the model perform when the capacities are not at the infeasibility border? These results suggest that the computational time did not change significantly at different capacity levels. For the one case in which the solution was not completely integral, (1000 flights at 40% connectivity), increasing the capacity resulted in integral solutions.

$ \mathcal{F} $	$ C / \mathcal{F} $	Dep Capacity	Arr Capacity	Obj Value	Time	% Nonint
1000	0.20	32	17	50750	342	0
1000	0.20	32	16	55450	227	0
1000	0.20	32	15	63525	262	0
1000	0.20	32	14	inf	-	-
1000	0.40	18	12	47000	290	0
1000	0.40	18	10	79916	521	2.2
1000	0.40	17	10	88241	741	4
1000	0.40	16	10	inf	-	-
1000	0.60	20	18	22316	369	0
1000	0.60	20	15	33292	376	0
1000	0.60	20	14	39266	359	0
1000	0.60	20	13	inf	-	-
1000	0.80	30	30	17000	183	0
1000	0.80	20	20	28250	283	0
1000	0.80	19	19	inf	-	-

Table 3.5: Results for varying capacity levels for 1000 flights.

3.2 Air Traffic Flow Management Problem Computations

We next performed experiments on a connected network of four airports: Boston Logan (BOS), NY LaGuardia (LGA), Washington National (DCA) and a node representing all other airports (X). Three hypothetical sectors that flights must use before landing at LaGuardia, were also introduced into the model. Flights coming from DCA would traverse two of these sectors before reaching LGA. Flights coming from BOS would traverse one of these sectors before reaching LGA. A certain fraction of the remaining flights would use the two sectors approach, while the other flights would enter through the one sector. See Figure 3-1.

$ \mathcal{F} $	$ C / \mathcal{F} $	Dep Capacity	Arr Capacity	Obj Value	Time	% Nonint
3000	0.20	30	30	42000	4537	0
3000	0.20	20	20	228000	5475	0
3000	0.20	19	19	inf	-	-
3000	0.40	30	30	42000	5062	0
3000	0.40	20	20	234000	4703	0
3000	0.40	19	19	inf	-	-
3000	0.60	30	30	42000	5629	0
3000	0.60	20	20	234000	5407	0
3000	0.60	19	19	inf	-	-
3000	0.80	30	30	42000	6021	0
3000	0.80	20	20	252000	9411	0
3000	0.80	19	19	inf	-	-

Table 3.6: Results for varying capacity levels for 3000 flights.

The three airports (BOS, LGA, DCA) and the three sectors were the only capacitated elements in the system. The other sectors were allocated unlimited capacity. We performed one set of experiments for 200 flights over a 24 hour time period and another set for 1000 flights over a 24 hour time period. The 200 flight dataset was obtained from the January 1993 Official Airline Guide (OAG). For the larger set of 1000 flights, the data was generated by the Pseudo-OAG Generator (POAGG) which is flight generation software developed at Draper Laboratories that realistically mimics the flight schedules of the OAG. All models were programmed in GAMS, run on a Sun SPARCstation 10 model 41 and solved with the solver CPLEXMIP 2.1. For most of the test cases the time interval was 5 minutes long. Since some of the sectors could be crossed in under 15 minutes, we tried to select a time interval that would capture as many sectors as possible without becoming prohibitively large. With this in mind, we decided to use a 5 minute interval whenever possible.

For the set of 200 flights, the time frame was 24 hours divided into discrete time units of 5 minutes each. To solve the problem CPLEX requires 234 seconds CPU time.

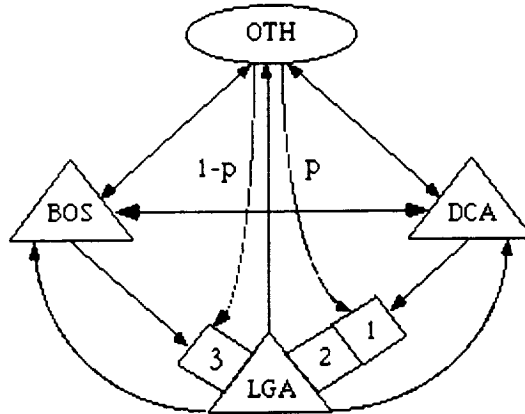


Figure 3-1: Sector flow model for test case.

Moreover, the resulting optimal solution was integral.

We were able to solve the 1000 flights problem at the infeasibility border over a 24 hour time period considering 15 minute intervals in 436 seconds CPU time. The optimal solution was once again integral. For the complete set of results see Table 3.7. Notice that the computation time in CPU seconds varies very little with the capacity restrictions in flights per time interval and that the solutions were completely integral.

Lastly, we obtained two realistic size datasets obtained directly from the OAG flight guide. This dataset has also been used to solve similar problems at the MITRE Corporation. The first dataset consists of 278 flights, 10 airports and 178 sectors, tested over a 7 hour time frame with 5 minute intervals. The second of these datasets consists of 1002 flights, 18 airports, and 305 sectors tested over an 8 hour time frame with 5 minute intervals.

The sector crossing times, sector and airport capacities, and required turnaround times were all provided by the FAA. Nothing used in these datasets was generated or hypothesized. We believe that these datasets are very comparable to the problem being solved everyday by the FAA.

For the first problem, consisting of 43226 constraints and 18733 variables, we found

Sector Capacity	Dep Capacity	Arr Capacity	Obj Value	Time	% Nonint
50	20	20	31975	425	0
20	20	20	31975	427	0
20	10	10	inf	-	-
15	15	15	68725	427	0
12	12	12	244225	450	0
11	11	11	inf	-	-
10	12	12	24225	456	0
5	12	12	24350	432	0
4	12	12	250975	466	0
3	12	12	295225	459	0
2	12	12	inf	-	-

Table 3.7: Results for varying capacity levels for 1000 flights.

an optimal solution in 1141 seconds. Furthermore, the solution obtained was completely integral. The second and larger dataset consisting of 151662 constraints and 69497 variables, was solved to optimality in 29920 seconds, again achieving completely integral solutions.

In summary, to address the questions we raised in the beginning of this chapter we remark:

1. In all but one instance in *MAGHP* and all instances of *TFMP* the relaxations of (*MAGHP*) and (*TFMP*) were integral.
2. The integrality of solutions was not affected by problem parameters, nor the size of the problem, except for the one instance in which the solution was non-integral.
3. The computational time required to obtain an optimal solution increases with the degree of connectivity as well as with the size of the problem.
4. Our approach improves upon earlier work particularly in obtaining integral solutions. This is evident from the fact that our formulation was almost always integral

especially in realistic instances. In contrast, for similar test cases (they used fixes while we used sectors) the formulation of Lindsay et al. (1993) was not integral, so they needed to improve their formulation by using automatic constraint generation techniques and also to use branch and bound.

5. We are able to solve large, realistic size problems in a reasonable amount of time. In addition, because we were able to solve the two instances of the TFMP with real data, we are very optimistic that our approach can effectively address the TFMP. Indeed, the reason we did not solve bigger problems is the difficulty of obtaining real data and memory restrictions of the SPARCstation.

Chapter 4

The Air Traffic Flow Management Rerouting Problem

This chapter addresses the traffic flow management rerouting problem (TFMRP), i.e., the problem of dynamically rerouting aircraft in the air traffic control system in order to avoid airspace regions that have reduced capacities primarily due to bad weather. The overall objective being the minimization of delay costs. In Section 4.1, we formulate the TFMRP as a integer multicommodity dynamic network flow problem. In Section 4.2, we discuss the multiple airline problem. In Section 4.3, we present the Lagrangian Generation Algorithm for solving this problem. In Section 4.4, we model the TFMP and the MAGHP as dynamic network flow problems.

In the United States, the control of air traffic is centered on 22 regional control centers. These centers receive information from aircraft and ground-based radars on location, altitude, and speed of aircraft, as well as weather information. When the weather conditions are poor, the capacities of some airports and sectors in the national airspace are forced to drop significantly or even to become zero. In this situation, the Air Traffic Command Center (ATCC) initiates a process to reschedule and reroute flights so that the delay costs caused by the weather conditions are kept to a minimum. Aircraft must then fly alternative routes if they were scheduled to pass through airspace regions of

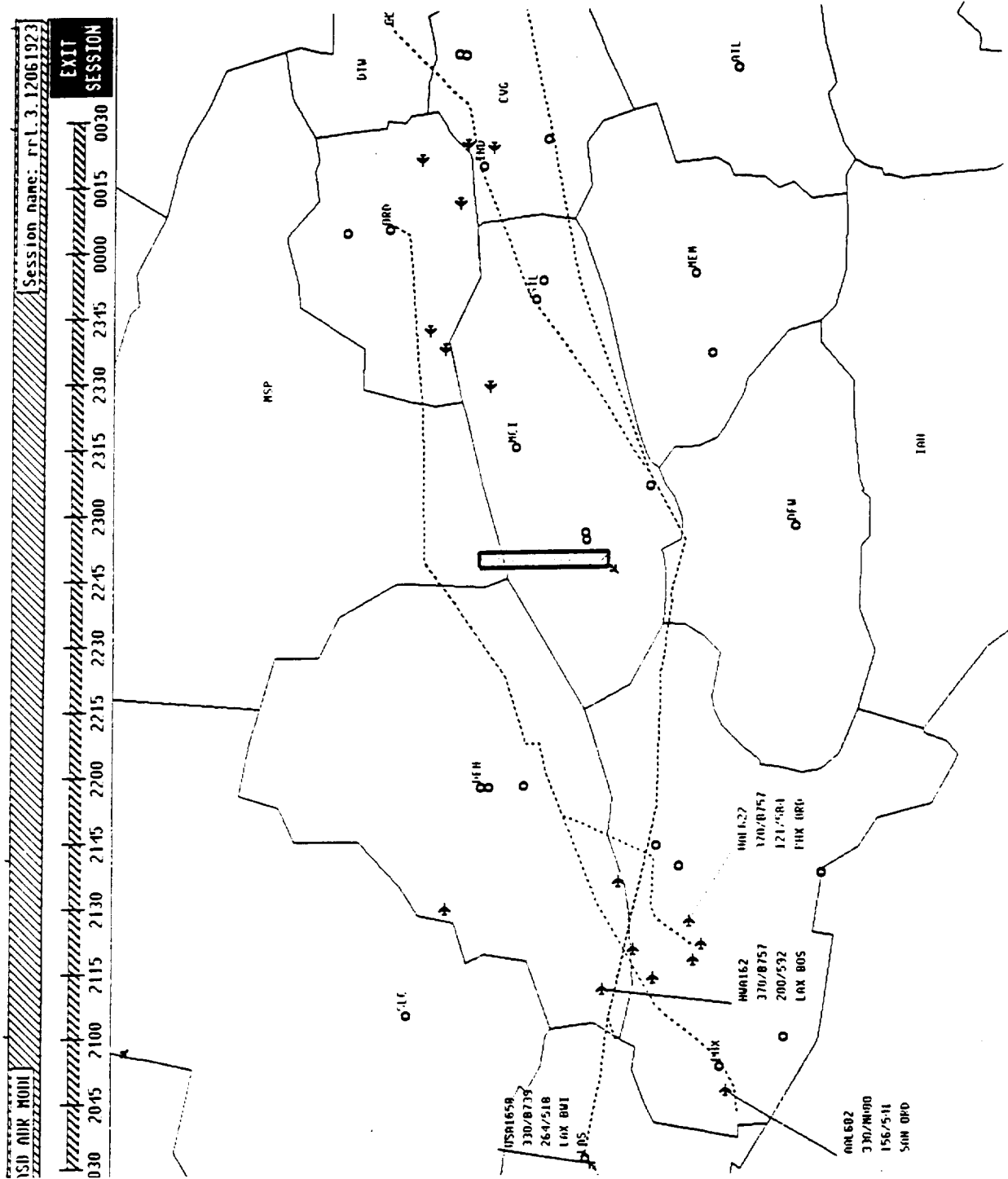


Figure 4-1: Alternative Routes Taken as Flights Avoid a Low Capacity Region.

reduced capacity (see Figure 4-1 for an example). Currently, these rerouting decisions are determined through an iterative process between the ATCC and the Airline Operations Centers (AOC). The ATCC contacts each airline's AOC concerning the necessity of rerouting. Each AOC then determines a set of new flight paths that they would like to use to complete their scheduled flights given the new limited capacity scenario information. This problem that each AOC faces when poor weather conditions limit the capacity of the national airspace, is the topic of this chapter.

4.1 The Integer Multicommodity Dynamic Network Flow Formulation

In this section we present an integer, multicommodity dynamic network flow model of the TFMRP. There are several components to the model. These include the dynamic network, the aggregated flow variables, the non-aggregated variables, and the capacity constraints. We will describe each of these in detail below.

We first describe the dynamic network that models the air traffic system. We create a graph which represents the actual geographical picture of the airport/airspace system. The nodes of the graph represent both airports as well as sectors. The example in Figure 4-2 of four airports and six sectors demonstrates how the nodes and arcs of the network are constructed.

The outlined regions are the sectors and the stars are the entrance and exit points for the sectors. We define one node for each of these sector crossing points. The circles are the airports. Each airport is represented by four nodes as described below. The arcs connect the entrance and exit points of a sector as well as these sector crossing points and the airports. Each arc (i, j) has a corresponding travel time, $t_{i,j}$. We assume that each sector has a limited number of entrance and exit points. In order to represent delay in the network we define self-loops with a travel time of one.

The commodities in the network are defined as origin-destination pairs of airports.

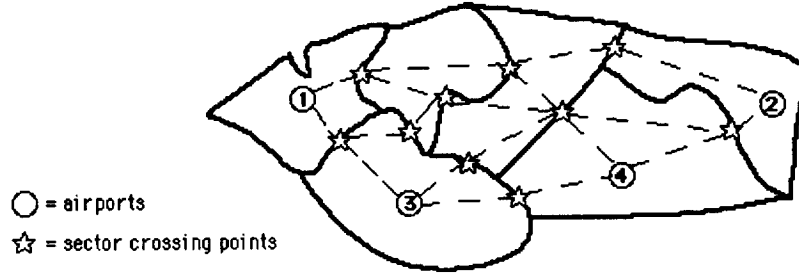


Figure 4-2: A network corresponding to four airports and six sectors.

So, if there are A airports and flights between all airports are flown, then there are exactly $A(A - 1)$ commodities. However, if we wanted to distinguish between certain characteristics of flights such as aircraft type, we could do this by breaking the commodities down even further. We will discuss the multiple airline problem in Section 4.2.

In order to model airport i we use four nodes i_A , i_B , i_C , and i_D (see Figure 4-3).

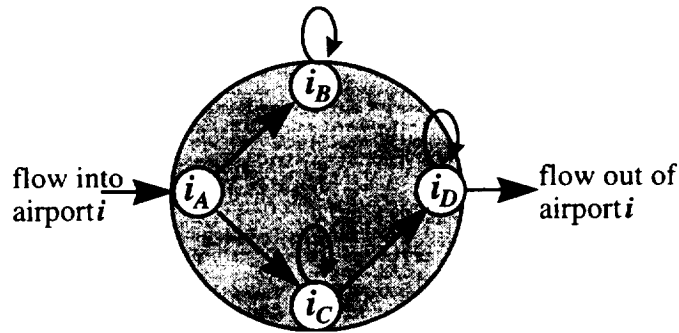


Figure 4-3: An airport is modeled with four nodes.

Node i_A is used to track all the incoming flights to airport i . Once a flight has landed at the airport, it is initially at node i_A . It can either proceed to node i_B or to node i_C . Node i_B represents the situation in which an aircraft has completed all of its required

flights for the day. Consequently, the flow into node i_B is removed from the network for the remaining time. Node i_C represents the situation in which an arriving aircraft must perform at least one more flight during the day. We use k to index the commodities of incoming flights and by k' the commodities of outgoing flights. At node i_C flow is exchanged from commodity k to commodity k' . The delay arc at node i_C models the situation in which an aircraft arrives before the continued flight is scheduled to depart. Flow then proceeds from node i_C to node i_D . At node i_D , new aircraft are introduced to the network and all the flights departing from the airport leave from this node. The delay arc at this node represents ground holding of flights.

Let $\mathcal{N} = (\mathcal{S}, \mathcal{E})$ be the network formed from airports and sectors, as described above. The set of commodities is denoted by $\{1, \dots, A(A - 1)\}$, where A is the number of airports. We discretize the time horizon into a set of time periods, $\mathcal{T} = \{1, \dots, T\}$. We refer to any particular time period t as the “time t .” Note that by “flight”, we mean a flight leg between two airports. A flight that is “continued” relies on an aircraft that has just completed a previous flight. The problem input data are given as follows.

Data:

- $t_{i,j}$ = minimum travel time along arc (i, j) ,
- $C_i(t)$ = capacity of sector i at time t ,
- $r(i)$ = turnaround time required to refuel, reload and clean an aircraft at airport i ,
- $orig(k)$ = airport of origin for a commodity k flight,
- $dest(k)$ = airport of destination for a commodity k flight,
- $N(k)$ = set of arcs that a commodity k flight can use,
- \mathcal{F} = set of flights,
- d_f = scheduled departure time of flight $f \in \mathcal{F}$,
- c^g = cost of holding a flight on the ground for one unit of time,
- c^a = average cost of flying an aircraft for one unit of time,

- H = total amount of scheduled flying time for all flights,
- \mathcal{C} = set of flights that are continued,
- T_f = set of feasible departure times for flight f where $f \in \mathcal{C}$,
- $k(f)$ = commodity of flight f where $f \in \mathcal{F}$,
- $k'(f)$ = commodity of the flight which precedes flight f where $f \in \mathcal{C}$,
- $Sup_k(t)$ = number of flights of commodity k that are scheduled to depart at time t ,
that are not continued,
- $Dem_k(t)$ = number of flights of commodity k that are scheduled to land at time t ,
at the latest, and do not continue to a later flight this day.

In order to reduce the dimensionality of the problem, we aggregate some of the variables over flights. By doing this, we lose the ability to distinguish some of the data over the flights. In particular, for the TFMP in Section 2.1, the costs of ground and air delay depended upon the flight, c_f^g and c_f^a . Since we are aggregating over flights in this formulation, this is no longer the case. The delay cost variables given above are simply c^g and c^a which contain no flight information. Moreover, the turnaround times for the TFMP also depended upon the flight, s_f , whereas the turnaround times in this aggregated formulation can no longer depend upon the flight. Instead, the turnaround time depends upon the airport, $r(i)$.

The most important data that is affected by the aggregation concern the continued flights. For the TFMP, each continued flight is assigned a unique flight to precede it. In the following formulation, each continued flight is assigned a unique commodity to precede it. Thus, for each continued flight, $f \in \mathcal{C}$, we define the commodity of flight f as $k(f)$ and we define the commodity of the corresponding flight that precedes flight $f \in \mathcal{C}$ as $k'(f)$. So, instead of forcing flight $f \in \mathcal{C}$ to be a continuation of flight f' , we ensure that for every continued flight there must be aircraft of commodity $k'(f)$ available for flight $f \in \mathcal{C}$ to use.

The aggregated variables are defined as:

$$x_{i,j}^k(t) = \text{number of flights of commodity } k \text{ that depart from} \\ \text{node } i \text{ at time } t \text{ and arrive at node } j \text{ at time } t + t_{i,j}.$$

Note that these variables are flow variables and not flight variables. So, in order to recommend flight paths we must have a method for disaggregating, i.e. for converting these flow variables to flight variables. In Section 4.3.2, we propose a method for performing the disaggregation, namely, the randomized rounding heuristic.

We also introduce flight variables for continued flights as follows:

$$y_f(t) = \begin{cases} 1, & \text{if the aircraft performing flight } f \in \mathcal{C} \text{ is ready for departure at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

By defining these non-aggregated variables, it becomes necessary to create the additional constraints that specify that there must be an aircraft available for each flight $f \in \mathcal{C}$ at some time. The reason we need these non-aggregated variables is to ensure that the necessary transfers of commodities occur at the flight level for continued flights. There are other methods of ensuring this transfer, but they involve adding a large number of side constraints to the formulation. Whereas, this method only requires the addition of $|\mathcal{C}|$ constraints.

The objective of the TFMRP is to minimize the total delay cost of flying all the required flights. Any flight may experience delay resulting from ground holding, decreasing speed while in the air, and selecting a route that is longer than the scheduled route. Moreover, a continued flight may also experience delay if there is no aircraft available for use at its departure time.

The objective function can be written in the following form:

$$\sum_{\{k,t,i=orig(k)\}} c^g x_{i_D,i_D}^k(t) + \sum_{\substack{\{(k,t,f):f \in C, \\ k(f)=k\}}} (t - d_f) y_f(t) + \sum_{\{k,i,t\}} c^a x_{i,i}^k(t) + \sum_{\{k,t,(i,j) \in N(k)\}} c^a t_{i,j} x_{i,j}^k(t) - c^a H. \quad (4.1)$$

The first term represents the cost of ground holding delay. The second term represents the cost of delay incurred by continued flights that were unable to depart on time as there were no available aircraft. The third term represents the cost of air delay due to speed reduction. Finally, the cost of delay caused by rerouting is obtained when the total cost of all scheduled air travel is subtracted from the total actual cost of air travel. The fourth term gives the total actual cost of all air travel and the fifth term is simply a constant representing the total cost of all scheduled air travel.

The constraints are given below.

(TFMRP)

$$\sum_{\{j:(i,j) \in N(k)\}} x_{i,j}^k(t) - \sum_{\{j:(j,i) \in N(k)\}} x_{j,i}^k(t - t_{j,i}) = 0, \quad \forall i \in \mathcal{S}, k, t, \quad (4.2)$$

$$\sum_{\{j:(j,i_A) \in N(k)\}} x_{j,i_A}^k(t - t_{j,i_A}) - x_{i_A,i_B}^k(t) - x_{i_A,i_C}^k(t) = 0, \quad \forall k, t, i = dest(k), \quad (4.3)$$

$$x_{i_A,i_B}^k(t) + x_{i_B,i_B}^k(t-1) - x_{i_B,i_B}^k(t) = Dem_k(t), \quad \forall k, t, i = dest(k), \quad (4.4)$$

$$\sum_{\{f \in C: k=k'(f)\}} y_f(t) + x_{i_C,i_C}^k(t) - x_{i_C,i_C}^k(t-1) - x_{i_A,i_C}^k(t-r(i)) = 0, \quad \forall k, t, i = dest(k), \quad (4.5)$$

$$\sum_{\{j:(i_D,j) \in N(k)\}} x_{i_D,j}^k(t) - \sum_{\{f \in C: k=k(f)\}} y_f(t) - x_{i_D,i_D}^k(t-1) = Sup_k(t), \quad \forall k, t, i = orig(k), \quad (4.6)$$

$$\sum_k \sum_{\{j:(i,j) \in N(k)\}} \sum_{\{t': t-t_{i,j} < t' \leq t\}} x_{i,j}^k(t') \leq C_i(t), \quad \forall i, t, \quad (4.7)$$

$$\sum_{\{t \in T_f\}} y_f(t) = 1, \quad \forall f \in \mathcal{C}, \quad (4.8)$$

$$x_{i,j}^k(t) \geq 0, \quad \forall i, j, k, t, \quad (4.9)$$

$$y_f(t) \in \{0, 1\}, \quad \forall f, t. \quad (4.10)$$

Constraint (4.2) represents dynamic flow conservation for the sectors. There is a constraint for each sector node, $i \in \mathcal{S}$, commodity k , and time t .

The next four constraints represent flow conservation at each of the airport nodes i_A , i_B , i_C , and i_D . Constraint (4.3) forces flow conservation at node i_A of airport i . At node i_A , we sum over all the nodes that can arrive at airport i from some sector j of commodity k , $\sum_{\{j:(j,i_A) \in N(k)\}} x_{j,i_A}^k(t - t_{j,i_A})$. The time index is $t - t_{j,i_A}$ since this is the time that the flow leaves j if it is to arrive at i at time t . This flow must be equal to the flow out of node i_A at time t . This flow goes to either node i_B (where it will be removed from the network) or to node i_C (where it will be transferred to another commodity).

Constraint (4.4) forces flow conservation at node i_B of airport i . The flow into node i_B at time t equals the flow from node i_A , $x_{i_A,i_B}^k(t)$ plus any flow that is held on the ground from the previous time period, $x_{i_B,i_B}^k(t - 1)$. This must equal the flow out which is $Dem_k(t)$ plus $x_{i_B,i_B}^k(t)$. In other words, if flow arrives at node i_B from node i_A prior to the latest time that it is scheduled to arrive, it lands and then is held on the ground at no cost until the time that the flow can be removed with a nonzero value of $Dem_k(t)$. Thus, every aircraft is removed from the network at its scheduled latest arrival time.

Constraint (4.5) forces flow conservation at node i_C . The flow into this node is of commodity k and the flow out of this node is a different commodity. The flow into node i_C only comes from node i_A , giving the term, $\sum_{\{k:i=dest(k)\}} x_{i_A,i_C}^k(t - r(i))$. The term $r(i)$ is the turnaround time at airport i . This is the time necessary in order to refuel and otherwise prepare the aircraft for the next flight. There is a delay arc at node i_C which captures those aircraft that arrive at airport i and are cleaned and refueled before they

are needed to fly the next flight. There is no cost for using this delay arc, it simply represents an aircraft waiting at an airport for its next flight. Lastly, the flow out of node i_C is given by $\sum_{\{f:k=k'(f)\}} y_f(t)$. This captures all the flights that are continued from commodity k flights that may depart at time t .

Constraint (4.6) forces flow conservation at node i_D . At this node, all the flow leaving i_D to some node j minus the flow into i_D must equal the supply at airport i at time t for commodity k such that $i = orig(k)$, which is denoted by $Sup_k(t)$. The flow leaving node i_D is simply given by $\sum_{\{j:(i_D,j) \in N(k)\}} x_{i_D,j}^k(t)$. This summation includes ground holding when $j = i_D$. The flow into node i_D is either from node i_C or from ground holding in the previous time period. In the former case, the flow from i_C passes through non-aggregated flight arcs. So this flow into node i_D comes from all the flights of commodity $k(f) = k$ that are continuations of previous flights. The ground holding value from the previous time period is given by $x_{i_D,i_D}^k(t-1)$.

Constraint (4.7) captures the capacity restrictions. There is a capacity on the number of aircraft that can be within sector i at time t given by $C_i(t)$. To represent this in terms of the flow variables, we need to sum over all commodities and all arcs that represent travel in sector i of commodity k at time t .

Constraint (4.8) forces the flights that are continued to occur. It states that for every flight that is a continuation of a previous flight, that flight must occur at some time, t , within a time window, T_f .

The remaining two constraint sets specify that the $x_{i,j}^k(t)$ variables are nonnegative, and that the $y_f(t)$ variables are binary. Note that if $y_f(t)$ is binary, then $x_{i,j}^k(t)$ would be integer.

The above formulation of the TFMRP is a multicommodity, integer variation of the minimum cost dynamic network flow problem. There are some important differences. First, the capacity constraint (4.7) is bundled over commodities, arcs and time periods, not just over commodities. Second, the disaggregated variables $y_f(t)$, are not flow variables, and finally there are additional side constraints (4.8).

4.2 Modeling the Multiple Airline Problem

The model proposed in the previous section can be used to solve the rerouting problem for a single airline. However, if we took the viewpoint of the FAA in which several airlines are occupying the airspace at the same time, then we need to modify the formulation slightly. The reason that the multiple airline problem is not the same concerns the continued flights.

In the formulation of Section 4.1 we stipulate that a continued flight does not rely on a unique flight, rather the formulation guarantees that an aircraft of the correct commodity will be available before the continued flight can depart. Once we introduce multiple airlines, we need to make sure that not only is the incoming aircraft of the correct commodity, but also that it belongs to the correct airline. For instance, obviously an American Airlines aircraft could not continue a Delta Airlines flight.

To do this we will need to redefine the commodities such that there is a unique commodity distinguished by each origin-destination pair and by each airline. By redefining the commodities in this manner, we can now ensure that a continued flight will rely on an aircraft whose commodity corresponds to the correct airline. This would increase the number of commodities by a multiple equal to the number of airlines.

We further need to consider the issue of fairness. It may be globally optimal to assign all the delay to a single airline, but of course, this solution is not acceptable. Thus, we would need to ensure that the delay is allocated in a fair manner across the airlines. This could be accomplished though modifying the Packing Formulation discussed in Section 4.3.3 by adding constraints that guarantee that each airline receive no more than a given percentage of the total delay.

4.3 The Lagrangian Generation Algorithm

In this section we use Lagrangian relaxation of the formulation outlined in Section 4.1, randomized rounding, and a packing formulation to propose near-optimal solutions for

the TFMRP. The overall algorithm is outlined below. We then explain each step in detail.

The Lagrangian Generation Algorithm

1. **(Lagrangian relaxation of the LP)** Starting with the formulation of TFMRP, i.e., the problem of minimizing (4.1) subject to the constraints (4.2)–(4.10), we relax the capacity constraints (4.7) into the objective function with multipliers, λ . We further relax the integrality constraints (4.10), i.e., we solve the relaxed problem as a linear program. We initialize the lower bound by solving the linear programming relaxation of the formulation for the TFMRP. The initial upper bound is infinity.
2. **(Solution of the relaxed problem)** We solve the relaxed problem and obtain a potentially fractional solution $y_f(t)$, $x_{i,j}^k(t)$.
3. **(Randomized Rounding)** We randomly round the variables $y_f(t)$ to zero-one solutions, and randomly decompose the flow into routes for flights. These routes are then added to a list of paths.
4. **(Packing formulation)** We formulate and solve an integer packing problem, in which we are attempting to pack the elements of the list of paths into the capacitated airspace system. If a new solution is found, we update the upper bound.
5. **(Stopping criterion)** If the upper and lower bound are within a desired accuracy ϵ , we stop.
6. **(Update of multipliers)** We update the multipliers λ and go to Step 2.

In the following subsections we describe each of the steps of the algorithm in detail.

4.3.1 Lagrangian Techniques

During Step 2 of the algorithm, we solve an uncapacitated multicommodity dynamic network flow problem as a linear program. Using the network flow solver of CPLEX, we solve the problem quickly (see Chapter 5). The main motivation of this step is to generate attractive routes for flights.

We update the multipliers using the iterative approach of Everett (1963) as follows. We represent the capacity constraints (4.7) as $Ax \leq b$. Let a_j be the j th row of the matrix A , and let b_j be the right hand side value for this row. Let x^k be the vector of solutions at the k th iteration. Let λ_j^k be the value of the lagrange multiplier for the j th constraint at iteration k :

$$\text{If } a'_j x^k > b_j \text{ then } \lambda_j^{k+1} = (1 + \delta_j^k) \lambda_j^k.$$

$$\text{If } a'_j x^k \leq b_j \text{ then } \lambda_j^{k+1} = (1 - \delta_j^k) \lambda_j^k,$$

where the parameters δ_j^k are updated by the rule:

$$\text{If } (a'_j x^k - b_j)(a'_j x^{k-1} - b_j) > 0, \text{ then } \delta_j^{k+1} = \epsilon_1 \delta_j^k.$$

$$\text{If } (a'_j x^k - b_j)(a'_j x^{k-1} - b_j) < 0, \text{ then } \delta_j^{k+1} = \epsilon_2 \delta_j^k.$$

$$\text{If } (a'_j x^k - b_j)(a'_j x^{k-1} - b_j) = 0, \text{ then } \delta_j^{k+1} = \delta_j^k.$$

The values of ϵ_1 and ϵ_2 are fixed parameters where $\epsilon_1 > 1$ and $\epsilon_2 < 1$.

The motivation for this method is as follows. If $a'_j x > b_j$ then the solution x uses too much of the available amount of the j th resource; thus, we increase the lagrange multiplier in order to penalize the violation more. In this method, the lagrange multiplier would be increased by a factor of $(1 + \delta_j^k)$. Likewise, if $a'_j x \leq b_j$ then the solution x uses a feasible amount of the j th resource; thus, we decrease the lagrange multiplier. It is decreased by the factor $(1 - \delta_j^k)$ as shown above. The amount of increase or decrease at each iteration

is determined by δ_j^k , called the step size, which is controlled at each iteration.

The values of δ_j^k are updated in the following manner. If a constraint is not satisfied iteration after iteration, then the step size is gradually increased based on the assumption that the value of λ_j^k may still be quite far from its optimal value. If the constraint fluctuates between feasibility and infeasibility, then the step size is reduced substantially based on the assumption that λ_j^k has come close to its optimal value. It is interesting to note that updating the lagrange multipliers depends only upon whether or not the constraint was satisfied, not on the magnitude of the difference.

Everett's Method does not guarantee convergence. However, it has been shown to perform very well computationally; see Pugh (1993).

4.3.2 Randomized Rounding Heuristic

The objective of this step is to generate a rich set of paths for individual flights from the aggregated flow solutions. The motivation for using randomization is to generate a broader set of solutions. After completing the Lagrangian relaxation step of the Lagrangian Generation Algorithm, we have a potentially fractional solution $y_f(t)$, $x_{i,j}^k(t)$.

Basically, this heuristic randomly walks through the network for every flight looking for a positive flow path. Starting at the departure airport of flight f and at the departure time of flight f , the heuristic randomly picks the next arc that has a positive flow on it. If this turns out to be a self-loop, then it remains at the airport for another time period and then makes another random decision about where to move in the next time period. Perhaps the next step places it at a new sector after it has completed the travel time for that arc. It then, once again, will randomly pick another arc that has some positive flow on it. It will walk through the network in this manner until it reaches its destination. With this in mind, we will now describe the process in full detail.

We will create a list of paths as follows:

$$\mathbf{P} = \{p_1, \dots, p_N\},$$

where $p_i = \{(s_i(0), t_i(0)), \dots, (s_i(n_i), t_i(n_i))\}$,

where $s_i(m)$ is the m th element of path p_i , and $t_i(m)$ is the time that the flight arrives at $s_i(m)$, $m = 0, \dots, n_i$. We define N as the total number of paths in the list and $n_i + 1$ as the total number of elements in path p_i .

We select the first element in each path $s_i(0)$ in a deterministic manner. For every noncontinued flight, we can create a path in which $s_i(0)$ is equal to node i_D of the departure airport and $t_i(0)$ is equal to the scheduled departure time. For every continued flight, we select the earliest time such that $y_f(t)$ is non-zero and set this equal to one. Now we can create a path for each continued flight where $s_i(0)$ is equal to node i_D of the departure airport and $t_i(0)$ is equal to the time at which $y_f(t)$ is equal to one.

To build the rest of each path, we step through the network for every flight beginning at the node given by $s_i(0)$ at the time $t_i(0)$. We will refer to the commodity of our flight as commodity k . Next, we randomly select from all arcs emanating from $s_i(0)$ that have a positive flow value, i.e. $x_{s_i(0),j}^k(t_i(0)) > 0$. This may include the possibility of selecting the arc that represents ground delay. Let the arc that we randomly select be denoted by $(s_i(0), \hat{j})$. We set

$$s_i(1) = \hat{j} \text{ and } t_i(1) = t_i(0) + t_{s_i(0),\hat{j}}.$$

We then need to decrease the flow value on the variable, $x_{s_i(0),\hat{j}}^k(t_i(0))$, by one.

We continue in this manner until we reach the node representing node i_A of the destination airport. Since these flows respect the flow conservation constraints in the Lagrangian relaxation, there will always be flow out of a node that the heuristic reaches.

There are a number of ways that we could set the probabilities used to select paths. Currently, we simply assign an equal probability to each node that has a positive flow. The rationale for this is simply to place a higher probability on obtaining alternative paths. Another possible method of randomizing would be to assign each arc a probability based on the flow on the arc. In particular, we could assign a probability P_j to arc (a, j)

defined by

$$P_j = \frac{x_{a,j}^k(t)}{\sum_{\{j:(a,j) \in N(k)\}} x_{a,j}^k(t)}.$$

We could then use these probabilities to determine which arc to select at each step.

After this heuristic is completed we have a set of path and time specifications that are added to a list of paths and used as input data for the Integer Packing Problem. Note that the heuristic only produces paths that satisfy flow conservation constraints. In particular, the capacity constraints (4.7), and as well as the airport constraints that handle continued flights (4.5), may not be satisfied. The path and time specifications will satisfy constraints (4.2), (4.3), (4.4), (4.6), (4.8), (4.9), and (4.10). By not forcing these paths to satisfy all the constraints in *(TFMRP)*, we create, after a few iterations, a list of paths that has more flexibility. The Integer Packing Formulation will then select from this list a combination of paths that will satisfy all the constraints in *(TFMRP)*.

4.3.3 The Integer Packing Formulation

The goal of the packing problem is to pack the paths generated by the randomized rounding heuristic into the air traffic system, so that all the necessary flights occur at correct (though not necessarily on time) times, and so that the capacity constraints are satisfied.

A path, $p_i = (s_i(0), t_i(0)), \dots, (s_i(n_i), t_i(n_i))$, specifies the elements and the times of a given route. The elements are both airports and sectors. Obviously the first element, $s_i(0)$ is the departure airport and the last element $s_i(n_i)$ is the arrival airport. However, ground holding is represented in these paths. So, if path p_i includes g units of ground holding then the elements $s_i(m)$, $m = 0, \dots, g - 1$, all represent the departure airport and the times $t_i(m)$, $m = 0, \dots, g - 1$, are each separated by one time unit. Thus, time $t_i(0)$ is not the actual departure time, rather it is the time at which an aircraft becomes available to perform the flight represented by path p_i . If this is a flight that is not continued, then $t_i(0)$ will be the same as the scheduled departure time. If the path represents a flight

that is continued, then $t_i(0)$ may be later than the scheduled departure time, since it is possible that there was no aircraft available for a continued flight to use at its departure time.

The decision variables in the packing formulation are as follows:

$$z_{f,i} = \begin{cases} 1, & \text{if path/time pair } p_i \text{ is used to fly flight } f \in \mathcal{F}, \\ 0, & \text{otherwise.} \end{cases}$$

Let Z be the set of feasible combinations of path/time pairs and flights.

$$Z = \{(f, i) : f \in \mathcal{C}, p_i(0) = \text{orig}(k(f)), p_i(n_i) = \text{dest}(k(f)), t_i(0) \geq d_f\} \cup \\ \{(f, i) : f \in \mathcal{F} \setminus \mathcal{C}, p_i(0) = \text{orig}(k(f)), p_i(n_i) = \text{dest}(k(f)), t_i(0) = d_f\}.$$

The objective of the packing problem is to minimize the cost of delay in the air as well as the ground holding cost of departing after the scheduled departure time. Let g once again be the number of units ground holding associated with path p_i . Then the delay cost of path p_i is given by

$$c_i = c^a[t_i(n_i) - t_i(g)] + c^g[t_i(g-1) - t_i(0)],$$

which includes both the amount of time spent ground holding and flying.

The objective function is as follows.

$$\sum_{i=1}^N c_i \sum_{\{f:(f,i) \in Z\}} z_{f,i} + c^g \left[\sum_{i=1}^N t_i(0) \sum_{\{f:(f,i) \in Z\}} z_{f,i} - \sum_{f \in \mathcal{F}} d_f \right] - c^a H.$$

The first term captures the cost resulting from ground holding delay and from air travel. When combined with the final term, which is a fixed cost of scheduled air travel, we capture the cost of delay resulting from ground holding, decreasing speed while in the air, and selecting a route that is longer than the scheduled route. The only remaining delay occurs if there is no aircraft available for a continued flight to use at its departure

time. To capture this delay, we sum over all the times at which aircraft become available to perform the flight represented by path p_i and subtract the sum of all the scheduled departure times.

The constraint set is given by the following equations.

$$(PP) \quad \sum_{\substack{\{(f,i) \in Z: \exists m | j = s_i(m), \\ t_i(m) \leq t < t_i(m+1)\}}} z_{f,i} \leq C_j(t), \quad \forall j \in \mathcal{S}, t, \quad (4.11)$$

$$\sum_{\{i: (f,i) \in Z\}} z_{f,i} = 1, \quad \forall f \in \mathcal{F}, \quad (4.12)$$

$$\sum_{\substack{\{(f,i) \in Z: f \in \mathcal{C}, \\ k'(f) = k, t_i(0) \leq t\}}} z_{f,i} \leq \sum_{\substack{\{(f',i') \in Z: k(f') = k, \\ t_{i'}(n_{i'}) + r(s_{i'}(n_{i'})) \leq t\}}} z_{f',i'}, \quad \forall k, t, \quad (4.13)$$

$$z_{f,i} \in \{0, 1\}, \quad \forall i = 1, \dots, N.$$

Constraints (4.11) represent the capacity constraints. They stipulate that for every sector j and every time t , the sum over all the flights that are in this sector at time t must be less than or equal to the sector capacity. A flight is within a sector at time t if it entered the sector before time t , $t_i(m) \leq t$, and has not yet entered the next sector in its path before time t , $t < t_i(m+1)$.

Constraints (4.12) ensure that each flight will be assigned to exactly one route.

Constraints (4.13) guarantee that a continued flight will not depart before a suitable aircraft has arrived for it to use. The left hand side of constraints (4.13) represent the number of continued flights whose preceding flight is of commodity k , $k'(f) = k$, and whose possible departure time is less than or equal to t . This number must be less than the number of flights of commodity k , $k(f') = k$, that arrive before time t minus the turnaround time, $r(s_{i'}(n_{i'}))$.

The final constraint set simply forces these variables to be binary.

4.4 TFMP and MAGHP as Special Cases

The air traffic flow management problem (TFMP) and the multi-airport ground-holding problem (MAGHP) can be seen as special cases of the rerouting problem. For instance, if we consider the TFMRP in which the choice of route through the network of sectors is fixed, then we have the TFMP. If we consider the TFMP in which the route consists of only a departure airport and an arrival airport, then we have the MAGHP. As both of these problems are of importance, it is reasonable to ask whether the Lagrangian Generation Algorithm could be used to efficiently solve these problems.

4.4.1 The TFMP Dynamic Network Flow Formulation

We shall start by describing how to model the TFMP as a multicommodity dynamic network. For TFMRP, we had variables of the following form.

$$x_{i,j}^k(t) = \text{number of flights of commodity } k \text{ that depart from} \\ \text{node } i \text{ at time } t \text{ and arrive at node } j \text{ at time } t + t_{i,j}.$$

The TFMP does not have as many choices for node j . In fact, for the TFMP, when the flow leaves node i there will be two options. It could be delayed at node i for one more time unit or it could proceed along its predefined path towards the next element in its path. If node i is one of the four airport nodes, then we have the exact same situation as we had for the rerouting case. If we define a path for each commodity as $R_k = \{r_{k,i_D}, r_{k,0}, \dots, r_{k,n_k}, r_{k,i_A}\}$, where n_k is the number of sectors, then we can modify the variables in the following manner.

$$\begin{aligned}
x_{i,0}^k(t) &= \text{number of flights of commodity } k \text{ that are held at node } r_{k,i} \\
&\quad \text{from time } t \text{ to time } t + 1, \text{ for } i = 0, \dots, n_k, \\
x_{i,1}^k(t) &= \text{number of flights of commodity } k \text{ that leave node } r_{k,i} \\
&\quad \text{at time } t \text{ and arrive at } r_{k,i+1} \text{ at time } t + t_i^k, \text{ for } i = 0, \dots, n_k.
\end{aligned}$$

The remaining variables are identical to the definition used for the rerouting problem.

$$y_f(t) = \begin{cases} 1, & \text{if an aircraft scheduled to perform flight } f \text{ arrives at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

By using these variables in (TFMRP), we obtain the following objective function for the traffic flow management problem.

$$MIN \sum_{\{k,t,i=orig(k)\}} c^g x_{i_D,i_D}^k(t) + \sum_{\{t,f \in C\}} (t - d_f) y_f(t) + \sum_{\{k,t,0 \leq i \leq n_k\}} c^a x_{i,0}^k(t).$$

The constraints are given by:

$$x_{i,0}^k(t) + x_{i,1}^k(t) - x_{i,0}^k(t-1) - x_{i-1,1}^k(t - t_{i-1}^k) = 0, \quad \forall k, t, i = 1, \dots, n_k,$$

$$x_{n_k,1}^k(t - t_{n_k}^k) - x_{i_A,i_B}^k(t) - x_{i_A,i_C}^k(t) = 0, \quad \forall k, t, i = dest(k),$$

$$x_{i_A,i_B}^k(t) + x_{i_B,i_B}^k(t-1) - x_{i_B,i_B}^k(t) = Dem_k(t), \quad \forall k, t, i = dest(k),$$

$$\sum_{\{f:k=k'(f)\}} y_f(t) + x_{i_C,i_C}^k(t) - x_{i_C,i_C}^k(t-1) - x_{i_A,i_C}^k(t-r(i)) = 0, \quad \forall k, t, i = dest(k),$$

$$x_{i_D, i_D}^k(t) + x_{i_D, 1}^k(t) - \sum_{\{f: k=k(f)\}} y_f(t) - x_{i_D, i_D}^k(t-1) = Sup_k(t), \quad \forall k, t, i = orig(k),$$

$$\sum_{\{k: \exists i | j=r_{i,k}\}} \sum_{\{t': t-t_i^k < t' \leq t\}} (x_{i,0}^k(t') + x_{i,1}^k(t')) \leq C_j(t), \quad \forall j, t,$$

$$\sum_{\{t \in T_f\}} y_f(t) = 1, \quad \forall f \in \mathcal{C},$$

$$x_{i,j}^k(t) \geq 0, \quad \forall i, j, k, t,$$

$$y_f(t) \in \{0, 1\}, \quad \forall f, t.$$

Note that many of the constraints are exactly the same as in (*TFMRP*). However, the majority of the constraints in (*TFMRP*) are due to the flow conservation constraints at sectors, (4.2) and from the capacity constraints (4.7). Both of these constraints are simplified for the TFMP due to the specification of the routes. We could now apply the Lagrangian Generation Algorithm to this dynamic formulation by relaxing the capacity constraints. The Integer Packing Formulation would be exactly the same as in Section 4.3.3.

4.4.2 The MAGHP Dynamic Network Flow Formulation

For the ground holding problem we disregard the sectors and only consider the airports. So for the formulation we simply need the four node depiction of the situation at an airport as well as the corresponding constraints. We obtain the following objective.

$$MIN \quad \sum_{\{k, t, i = orig(k)\}} c^g x_{i_D, i_D}^k(t) + \sum_{\{t, f \in \mathcal{C}\}} (t - d_f) y_f(t).$$

The constraint set is given by the following.

$$\begin{aligned}
x_{i_D, i_A}^k(t - t^k) - x_{i_A, i_B}^k(t) - x_{i_A, i_C}^k(t) &= 0, & \forall k, t, i = \text{dest}(k), \\
x_{i_A, i_B}^k(t) + x_{i_B, i_B}^k(t - 1) - x_{i_B, i_B}^k(t) &= \text{Dem}_k(t), & \forall k, t, i = \text{dest}(k), \\
\sum_{\{f: k=k'(f)\}} y_f(t) + x_{i_C, i_C}^k(t) - x_{i_C, i_C}^k(t - 1) - x_{i_A, i_C}^k(t - t_a) &= 0, & \forall k, t, i = \text{dest}(k), \\
x_{i_D, i_D}^k(t) + x_{i_D, i_A}^k(t) - \sum_{\{f: k=k(f)\}} y_f(t) - x_{i_D, i_D}^k(t - 1) &= \text{Sup}_k(t), & \forall k, t, i = \text{orig}(k), \\
\sum_{\{k: i_D = \text{orig}(k)\}} (x_{i_D, i_A}^k(t) &\leq C_{i_D}(t), & \forall i_D, t, \\
\sum_{\{k: i_A = \text{dest}(k)\}} (x_{i_A, i_B}^k(t) + x_{i_A, i_C}^k(t)) &\leq C_{i_A}(t), & \forall i_A, t, \\
\sum_{\{t \in T_f\}} y_f(t) &= 1, & \forall f \in \mathcal{C}, \\
x_{i, j}^k(t) &\geq 0, & \forall k, t, (i, j) \in N(k), \\
y_f(t) &\in \{0, 1\}, & \forall f, t.
\end{aligned}$$

This formulation differs from the last in that there are no sector flow conservation constraints and the capacity constraints have been separated into only departure and arrival capacity, given above as $C_{i_D}(t)$ and $C_{i_A}(t)$, respectively. Once again, we can now apply the Lagrangian Generation Algorithm to this formulation by relaxing the capacity constraints.

Chapter 5

Computational Results for the Lagrangian Generation Algorithm

In this chapter, we report on the computational performance of the Lagrangian Generation Algorithm. In Sections 5.1, 5.2, and 5.3, the Lagrangian Generation Algorithm is applied to solve three instances of the air traffic flow management rerouting problem. These instances model a bad weather front passing through a portion of southwestern United States. This region consists of four airports, located at Denver (DEN), Phoenix (PHX), Las Vegas (LAS), and Salt Lake City (SLC). There are 42 sectors that lie in the vicinity of these four airports as shown in Figure 5-1. This data, that was provided by the FAA, includes the travel times for the sectors as well as the necessary turnaround times. Each of the three instances has a different capacity scenario determined by the movement of a weather front through the region.

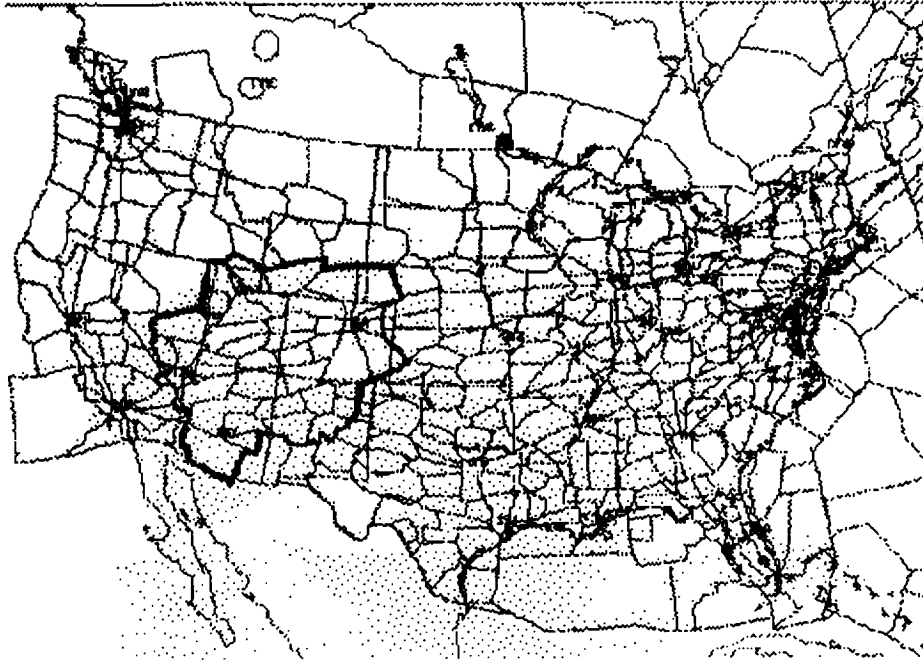


Figure 5-1: Sector map of the US with southwest region shown.

5.1 Computations for Weather Scenario I

For this instance, flight schedules for 71 flights between the 4 airports shown in Figure 5-1 are extracted from a dataset provided by the FAA that covers an 8 hour time frame with 5 minute time intervals. We simulated a weather front passing from the northeast corner of this region to the southwest corner. We set the sector capacities according to a uniform distribution.

During normal weather conditions, the sector capacities were generated using a uniform distribution with a mean determined by the size of the sector and a standard deviation of one. Figure 5-2 shows the sector capacities during normal weather conditions. At the cusp of the storm, the capacities were generated using a uniform distribution with a mean of zero. All of the resulting negative values were set to zero. As the weather

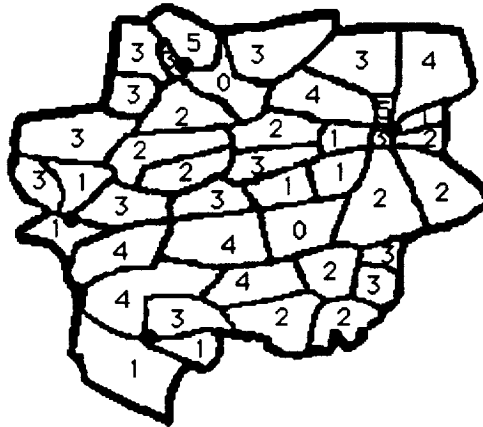


Figure 5-2: Weather Scenario I at 8am representing normal operating conditions before the weather front has hit this region.

front gradually passes through, the mean slowly increases until it reached the level during normal weather conditions. Figure 5-3 shows the different weather scenarios that evolve over time for this first capacity scenario as the weather front passes through the region. The shaded areas show the cusp of weather front with the numbers being the corresponding sector capacities.

To achieve a lower bound on the solution, we solve the LP relaxation of the multi-commodity dynamic network formulation, which consists of 24,509 constraints and 61,912 variables. Finding a solution to the LP requires 181 seconds. The solution is highly non-integral with an objective value of 2498.5.

We ran the Lagrangian Generation Algorithm setting the parameter values as $\epsilon_1 = 2$ and $\epsilon_2 = 0.33$. The starting values for λ_j^0 and δ_j^0 were set to 10 and 0.8 respectively. We did not perform extensive trials to determine the best starting values for λ and δ , however, a few different settings were tried. The results presented reflect the stated starting conditions, which converged in the shortest number of iterations. The Lagrangian relaxation solves a network problem with 15,279 nodes, 54,427 arcs and 26 of the side

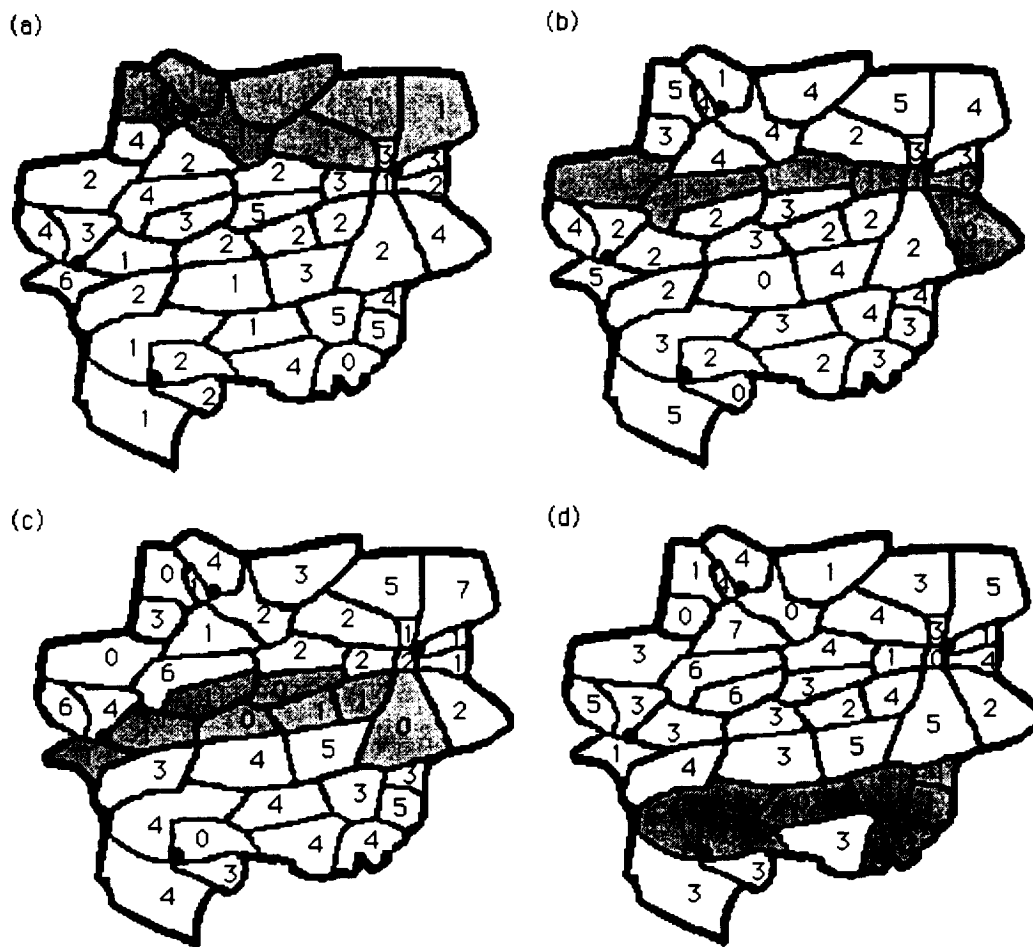


Figure 5-3: Weather Scenario I at: (a) 9:45am (b) 10:35am (c) 11:25am (d) 12:15pm.

constraints (4.8). The size of the integer packing problem grows at each iteration as the size of the list of paths increases. At the final iteration, the formulation consists of 2209 constraints and 145 variables, which reduces to 424 constraints and 125 variables by using some presolving routines in CPLEX. Table 5.1 tracks the performance of the Lagrangian Generation Algorithm as it steps through the algorithm. All of times are given in seconds.

The total amount of time needed to solve this problem, including the subproblem times for the Lagrangian relaxation and the integer packing problem given in Table 5.1,

Iteration	Lagrangian			Packing		
	Time	Obj Value	No. Infeas.	Presolve Time	IP Time	Obj Value
0	25.01	-113062	89	0.03	-	Inf.
1	24.22	-12572	65	0.03	-	Inf.
2	24.70	1321.46	77	0.07	-	Inf.
3	24.42	2073.94	54	0.12	-	Inf.
4	24.58	2248.13	64	0.18	-	Inf.
5	24.40	2324.27	68	0.18	-	Inf.
6	24.65	2286.42	61	0.18	-	Inf.
7	24.67	2377.25	66	0.18	15.26	2509

Table 5.1: Computational Results for weather scenario I.

was 330 seconds. The solution value found, 2509, is within 0.4% of the lower bound. The total amount of ground holding imposed upon this solution was 810 minutes and the total amount of airholding was 15 minutes.

Numerous routes were used for each particular commodity. As an example, we will consider one of these commodities, Las Vegas to Phoenix, in detail. Three different routes were used over the course of the day to fly flights of this commodity. Two routes are shown and labeled in Figure 5-4. Most of the time when this commodity is flown, route



Figure 5-4: Two routes used at different times to fly from Las Vegas to Phoenix.

1 is used and ground delay alone is assigned to avoid any anticipated capacity problems

while in the air. However, we will examine two flights that use route 2. The important times for these two flights are depicted in Figure 5-5 and explained below.

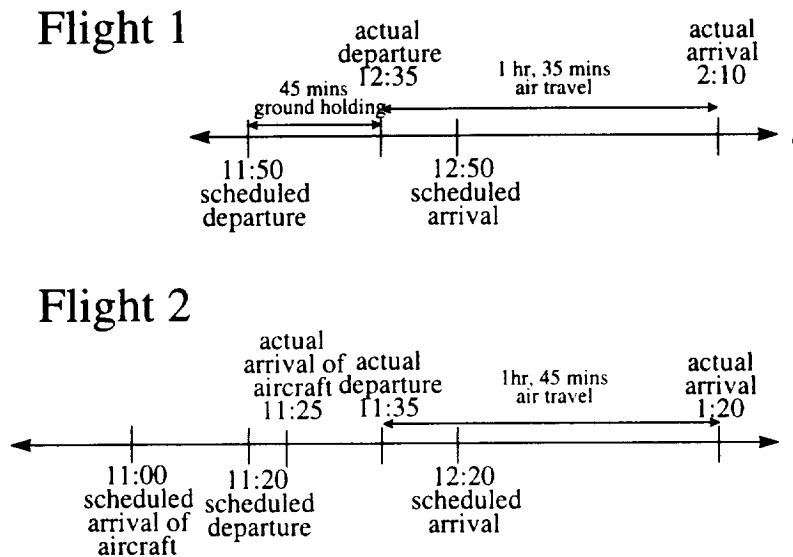


Figure 5-5: Time lines of two flights that use route 2 on Figure 5-4.

At 11:50am a flight is scheduled to depart from Las Vegas. This flight is scheduled to arrive at Phoenix one hour later. The flight is held on the ground for 45 minutes and actually departs at 12:35pm. It reaches Phoenix at 2:10pm making the total amount of time spent traveling equal to 1 hour and 35 minutes, which is 35 minutes more than scheduled. The total delay of 80 minutes resulted from 45 minutes of ground holding and 35 minutes of rerouting delay.

At 11:20am, a continued flight is scheduled to depart from Las Vegas. It is relying on an aircraft that is scheduled to arrive at Las Vegas at 11:00am. However, this incoming flight experiences delay and does not arrive at Las Vegas until 11:25am. The aircraft is immediately refueled and boarded, and the continued flight actually departs at time 11:35am, 15 minutes after the scheduled departure time. There is no ground holding assigned to this flight. The flight travels along route 2 and is further delayed in terms of airspeed reduction. When the flight reaches the shaded sector in Figure 5-4, its speed

is reduced such that the travel time through this sector is increased by 10 minutes. It reaches Phoenix at 1:20pm, a total of 1 hour after the scheduled arrival time. This total delay of 60 minutes resulted from 10 minutes of airspeed reduction, 35 minutes of rerouting delay and 15 minutes of departure delay due to the late incoming aircraft.

5.2 Computations for Weather Scenario II

For this instance, the same flight schedules for the 71 flights between the 4 airports shown in Figure 5-1 were used. However, to simulate the weather front passing from the northeast corner of this region to the southwest corner, the sector capacities were set deterministically.

During normal weather conditions, we fixed the capacities according to the size of the sector. At the cusp of the storm, the sector capacities were set to zero. One hour later, as the storm front moves along, those sectors that had zero capacity during the last hour, now have a slightly increased capacity of one. The sector capacities would continue to increase hourly until they have resumed the level of normal weather conditions. Figure 5-6 shows the different sector capacities that evolve over time for the second capacity scenario as the weather front passes through the region. Figure 5-6 (a) through Figure 5-6 (f) show the sector capacities between the times 8:20 to 9:20am, 9:25 to 10:25am, 10:30 to 11:30am, 11:35am to 12:35pm, 12:40 to 1:40pm, and 1:45 to 2:45pm, respectively. The shaded areas show the cusp of the weather front in which the sector capacities are zero. The front appears to move along gradually spending one hour before each progression.

To achieve a lower bound on the solution, we solve the LP relaxation of the multicommodity dynamic network formulation. The size of the formulation is not affected by the change in the weather capacity scenario. Thus, the number of constraints and variables is the same as specified for weather scenario I. Solving the LP requires 59 seconds and gives a solution that is highly non-integral with an objective value of 2387.

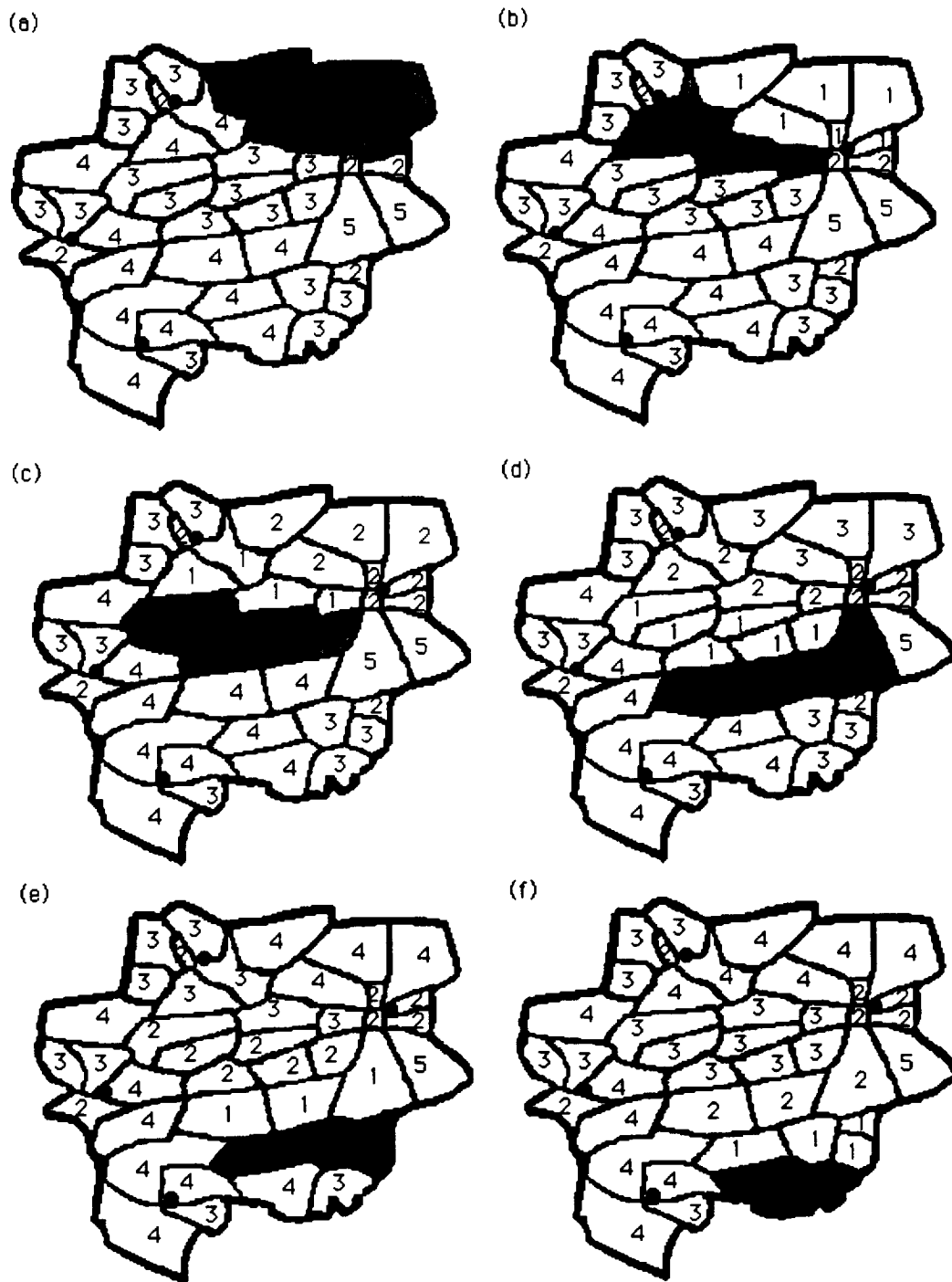


Figure 5-6: Weather Scenario II: (a) 8:20 to 9:20 am, (b) 9:25 to 10:25am, (c) 10:30 to 11:30am, (d) 11:35am to 12:35pm, (e) 12:40 to 1:40pm, (f) 1:45 to 2:45pm.

Iteration	Lagrangian			Packing		
	Time	Obj Value	No. Infeas.	Presolve Time	IP Time	Obj Value
0	24.72	-107072	72	0.12	-	Inf.
1	24.63	-11148	60	0.10	-	Inf.
2	24.72	1618.38	20	0.42	17.51	2418

Table 5.2: Computational Results for weather scenario II.

We ran the Lagrangian Generation Algorithm setting the parameter values as $\epsilon_1 = 2$ and $\epsilon_2 = 0.33$. The starting values for λ_j^0 and δ_j^0 were set to 10 and 0.8 respectively. Table 5.2 tracks the performance of the Lagrangian Generation Algorithm as it steps through the algorithm. The problem sizes are the same as for weather scenario I. All of times are given in seconds.

The total amount of time needed to solve this problem, including the subproblem times for the Lagrangian relaxation and the integer packing problem given in Table 5.2, was 116 seconds. The solution value found, 2418, is within 1.2% of the lower bound. The total amount of ground holding imposed upon this solution was 480 minutes and no airholding was used.

Iteration	Lagrangian			Packing		
	Time	Obj Value	No. Infeas.	Presolve Time	IP Time	Obj Value
0	24.72	-107072	72	0.12	-	Inf.
1	24.63	-11148	60	0.10	-	Inf.
2	24.72	1618.38	20	0.42	17.51	2418
3	26.18	2104.20	25	0.48	17.42	2408
4	25.98	2231.75	32	0.53	18.02	2396
5	25.98	2295.00	18	0.53	19.10	2393
6	25.98	2317.74	19	0.57	19.35	2392
7	25.43	2345.69	21	0.62	18.46	2389
8	25.88	2327.27	21	0.65	19.56	2389
9	25.82	2360.46	22	0.65	19.89	2389
10	25.88	2304.59	22	0.68	19.99	2389
11	25.78	2367.44	10	0.72	20.03	2389
12	26.55	2375.27	12	0.68	20.36	2389
13	25.93	2378.53	14	0.72	21.58	2389
14	26.45	2379.18	21	0.67	21.60	2389

Table 5.3: Computational Results for weather scenario II for 15 iterations.

We ran the algorithm for one hundred iterations, without the ϵ stopping condition, to see if we could find a solution which is even better than the one found above. The results from the first fifteen iterations are given in the Table 5.3. The remaining eighty-five iterations did not find a solution with a lower solution value less than the value at the fifteenth iteration, and were thus not including in the table. Within 7 iterations we generate the best solution that we were able to find. This value is 2389, which is within 0.08 % of the lower bound of 2387.



Figure 5-7: Three routes used to fly between Salt Lake City and Phoenix.

Numerous routes were used for each particular commodity. In Figure 5-7 we look at a few of the routes used to fly between Phoenix and Salt Lake City. Route 1 travels from Salt Lake City heading towards Phoenix. This flight is scheduled to depart at 9:30am, but is held on the ground until 11:30am, incurring a two hour ground delay. At that point, the flight departs from Salt Lake City and follows a reasonably direct route, route 1, to Phoenix, basically trailing the storm front.

Routes 2 and 3 go north from Phoenix to Salt Lake City. The flight that travels along route 2 is scheduled to depart at 9:25am, but is held on the ground for 1 hour. It then departs and follows a circuitous route due to the limited sector capacities. The flight

that travels along route 3 is scheduled to depart at 11:30am and only suffers a 20 minute ground delay. This flight passes through its route immediately after the weather front. The capacities are still quite limited, however, forcing this flight to significantly deviate from the shortest path. In order to appreciate how this affects travel time, we note that route 1 requires 1 hour and 55 minutes of flying time, route 2 requires 2 hours and 25 minutes of flying time and route 3 requires 2 hours and 35 minutes of flying time.

5.3 Computations for Weather Scenario III

The final weather scenario that we tested uses the same geographic area as shown in Figure 5-1. However, we increased the number of flights to 200 flights and scaled the normal weather sector capacities accordingly. To simulate the weather front passing from the northeast corner of this region to the southwest corner, we set the sector capacities deterministically.

During normal weather conditions, we fixed the sector capacities according to the size of the sector. At the cusp of the storm, the sector capacities were set to zero. The front once again gradually moves along spending forty minutes, before each progression. As the cusp of the storm front moves through the region, the available sector capacity increases by two each forty minutes. So this weather front moves more quickly and does not leave such bad conditions behind it, as does the previous weather scenarios. Had we kept the capacity levels at that of either the two previous scenarios, then the problem would have been infeasible, meaning that there would have been no way to complete all of the 200 flights during the time frame without cancelling some flights. Figure 5-8 shows the different weather scenarios that evolve over time for the third capacity scenario as the weather front passes through the region. Figure 5-8 (a) through Figure 5-8 (f) show the sector capacities between the times 8:20 to 9:00am, 9:05 to 9:45am, 9:50 to 10:30am, 10:35 to 11:15am, 11:20am to 12:00pm, and 12:05 to 12:45pm, respectively. The shaded areas show the cusp of the weather front in which the sector capacities are set to zero.

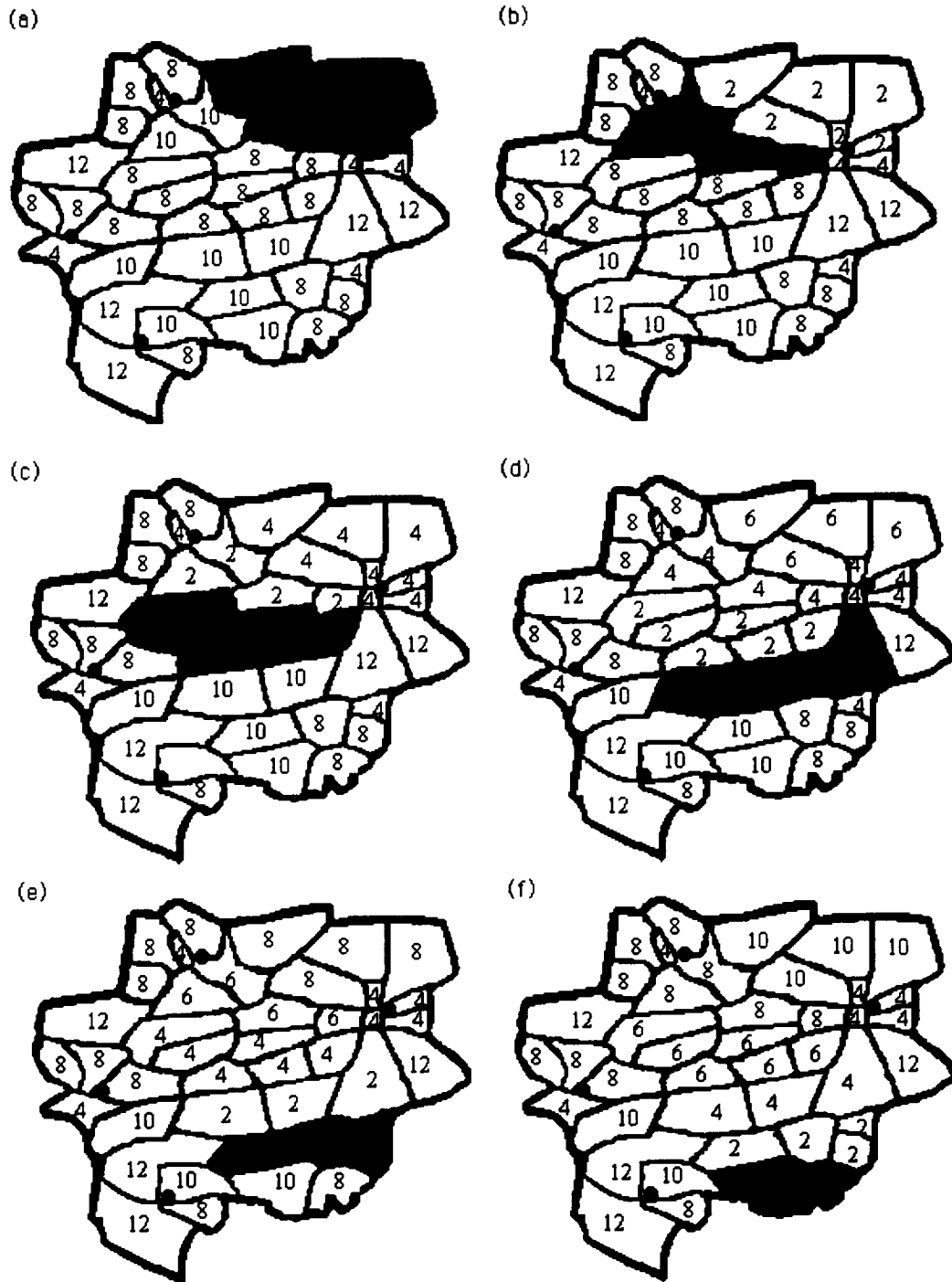


Figure 5-8: Weather Scenario III for 200 flights: (a) 8:20 to 9:00am, (b) 9:05 to 9:45am, (c) 9:50 to 10:30am, (d) 10:35 to 11:15am, (e) 11:20am to 12:00pm, (f) 12:05 to 12:45pm.

To achieve a lower bound on the solution, we solve the LP relaxation of the multi-commodity dynamic network formulation, which consists of 25,881 constraints and 66,489 variables. Solving the LP requires 86 seconds and gives a solution that is highly non-integral with an objective value of 6513.5.

We again ran the Lagrangian Generation Algorithm with the same starting values as before. Table 5.4 tracks the performance of the Lagrangian Generation Algorithm as it steps through the algorithm. All of the times given in Table 5.4 are in seconds. The Lagrangian relaxation solves a network problem with 16,219 nodes, 57,294 arcs and 138 of the side constraints (4.8). The size of the integer packing problem grows at each iteration as the size of the list of paths increases. At the final iteration, the formulation consists of 2394 constraints and 1197 variables, which reduces to 879 constraints and 1078 variables by using some presolving routines in CPLEX.

Iteration	Lagrangian			Packing		
	Time	Obj Value	No. Infeas.	Presolve Time	IP Time	Obj Value
0	36.52	-283953	91	0.19	-	Inf.
1	36.49	-26652	54	0.26	-	Inf.
2	36.31	5772.76	39	0.81	-	Inf.
3	36.56	6152.06	44	0.83	18.11	6574

Table 5.4: Computational Results for 200 flight dataset.

The total amount of time needed to solve this problem, including the subproblem times for the Lagrangian relaxation and the integer packing problem given in Table 5.4, was 169 seconds. The solution value found, 6574, is within 0.92% of the lower bound. The total amount of ground holding imposed upon this solution was 700 minutes and no airholding was used.

Once again, we ran this problem for more iterations to see if we could obtain a better solution, even though we already found a solution that is well within the tolerance. Table 5.5 gives results that show all the solutions that we generated. Of these, the best solution, 6520, is within 0.09% of the lower bound, 6513.5.

Again, we can see how well the Lagrangian Generation Algorithm performs in terms

Iteration	Lagrangian			Packing		
	Time	Obj Value	No. Infeas.	Presolve Time	IP Time	Obj Value
0	36.10	-283953	91	0.19	-	Inf.
1	36.16	-26652	54	0.26	-	Inf.
2	36.14	5772.76	39	0.81	-	Inf.
3	36.26	6152.06	44	0.83	18.11	6574
4	36.08	6291.80	42	0.92	18.78	6545
5	36.03	6327.60	39	0.94	18.63	6529
6	35.97	6307.54	42	1.00	19.00	6527
7	35.83	6370.08	37	1.04	19.12	6523
8	36.51	6453.82	36	1.08	19.58	6520
9	36.26	6458.95	47	1.09	19.64	6520
10	35.90	6319.08	41	1.10	20.01	6520
11	36.22	6380.18	36	1.10	20.82	6520

Table 5.5: Computational Results for 200 flight dataset for 12 iterations.

of speed and accuracy of solutions. This last dataset is quite large, but the size of the multicommodity dynamic network flow formulation is not drastically increased by an increase in the number of flights. Only the number of non-aggregated variables is increased; the number of aggregated variables remains the same. The only set of constraints that is affected by this increase is constraint (4.8), which comprises a very small amount of the total number of constraints.

Chapter 6

Conclusions

In this thesis, we have presented what we believe to be interesting and practical approaches for solving the Air Traffic Flow Management Problem and the Air Traffic Flow Management Rerouting Problem. In Section 6.1, we summarize the contributions of the research presented in this thesis. In Section 6.2, we discuss some remaining issues of concern to the FAA.

6.1 Contributions

The Air Traffic Flow Management Problem determines how to allocate ground holds to flights as well as how to control the enroute speed of an aircraft while taking into account all the capacitated elements in the system (arrival, departure and sector capacities). We built an integer programming formulation to minimize delay costs. The formulation is quite strong as some of the proposed inequalities are facet defining for the convex hull of solutions. We addressed the complexity of the TFMP and showed that it is NP-hard. The formulation easily extends to incorporate the dependence of airport runway capacity of departures and arrivals, hub connectivity, and banking of flights. When specialized for the multiple airport ground holding problem, we proved that the LP relaxation bound of our formulation is at least as strong as others proposed in the literature. The solutions

of the LP relaxation of the TFMP were almost always integral, so there was no need to branch and bound. In essence, our formulations reduced the problem to efficiently solving large scale linear programming problems. As a result, the computation times were reasonably small for large scale, realistic size problems involving thousands of flights. Short computational times and integrality properties are particularly important, since these models are intended to be used on-line and solved repeatedly during a day.

The Air Traffic Flow Management Rerouting Problem (TFMRP) determines how to reroute flights through different flight paths in order to reach their destinations if the current routes pass through a region that is unusable as a result of poor weather conditions. This is the first research that has taken a global look at rerouting. We not only determined the best routes for the aircraft to follow, but also the amount of ground holding, and the amount of speed adjustment while taking into consideration that the entire national airspace system is operating under a strict capacity restriction. To address this problem, we presented an integrated mathematical programming approach, which utilizes several methodologies, for the problem of minimizing delay costs. In order to address the high dimensionality, we began by presenting an aggregate model, in which the problem is formulated as a dynamic, multicommodity, integer network flow problem with certain side constraints. Using Lagrangian relaxation, we generated aggregate flows that are decomposed into a collection of flight paths for individual aircraft using a randomized rounding heuristic. This collection of paths was then used in a packing integer programming formulation, the solution of which generates feasible and near-optimal routes for individual flights. The overall algorithm, termed the Lagrangian Generation Algorithm, was used to solve real problems in the southwestern portion of United States. The solutions returned by our algorithm were within 1% of the corresponding lower bounds. However, there are several remaining issues that need to be addressed before this approach can be implemented. In Section 6.2, we discuss some of these issues.

In the course of this research, we obtained some general insights that may have wider applicability. First, we have shown through our formulation of the TFMP, that

by redefining traditional time assignment variables as “by time t ” rather than “at time t ”, some of the proposed inequalities are likely to be facet defining for the convex hull of solutions, which in turn leads to considerable computational improvement in terms of achieving integrality. We believe that this redefinition may potentially be beneficial for other scheduling problems where we are assigning items to specific time intervals.

Second, the algorithmic design of the Lagrangian Generation Algorithm could be used in other problem contexts. The idea of extracting solutions using randomization and combining these solutions using integer programming may be useful in other problems as well.

Although we have presented our formulations in the context of air traffic control, we envision other applications of our models in any area in which goods are dynamically flowing through a system with several types of capacitated elements such as manufacturing, telecommunications and ground transportation systems.

6.2 Other FAA Issues

For several years, the FAA has been operating an Air Traffic Control System Command Center (ATCSCC) in Washington, D.C. This center is equipped with outstanding information-gathering capabilities that dynamically keeps track of all the information about capacities, flight information, weather, etc. As we have mentioned earlier, the FAA uses a computerized procedure to allocate ground holding delays based on a first-come-first-serve rule. We believe that the optimization-based approaches that are described in this thesis are well suited to be the optimization “brain” for this system. However, there are important issues that need to be addressed before applying an optimization based approach in a real world environment.

1 Interaction with airlines.

After the ground delays are issued, the airlines have the opportunity to propose modifications to these delays through a cancellation and substitution process. It

would be important to elucidate the effects of this interaction.

2 Passenger and flight crew connections.

There are more connectivity requirements than just the aircraft connectivity. In particular, there are the connections of the passengers and the flight crew. On occasion delay will be incurred if many of the passengers have not yet arrived due to delay encountered during the first leg of their journey. Moreover, if the delay involves the flight crew, either an alternative flight crew must be reassigned or the flight will experience a delay in waiting for the crew to arrive.

3 Mechanical Difficulties.

Flights are often further delayed by mechanical failures of the aircraft. Research could be done that allows for the possibility of randomly occurring delay from equipment failure.

4 Flight cancellations.

When weather conditions become bad enough to warrant extreme measures, flights are often cancelled or forced to land at alternative destinations. This situation causes many difficulties throughout the air traffic system. However, it is a viable option that has not yet been considered.

5 Dynamic updating of decisions.

Both ground and enroute delays are determined simultaneously several hours before a flight leaves. In practice, however, enroute delays are not known until after the aircraft is in the air. Clearly more research is needed to clarify the implications of considering enroute delays on a much shorter time scale and to demonstrate how to effectively update the previous solution to incorporate any new information.

6 Stochastic modeling.

The model presented in this paper assumes a deterministic environment. More research is needed to account for stochasticities inherent in a system that is strongly

dependent upon weather conditions.

Appendix A

On the Polyhedral Relationships Between Ground Holding Formulations

We intend to establish Proposition 1. Since IP'_{MAGHP} , IP_{VBO} and IP_{TP} are valid integer programming formulations, it is clear that $IP'_{MAGHP} = IP_{VBO} = IP_{TP}$. Moreover, since the IP is more restrictive than its relaxation, $IP'_{MAGHP} \subseteq P'_{MAGHP}$.

To show the relationship $P'_{MAGHP} \subseteq P_{TP}$ we will start with a feasible point in P'_{MAGHP} , \bar{z}_{ft} , and show that this is indeed feasible to P_{TP} . The first two constraints and the last constraint are identical in the two models. So what remains to be shown is that any point, \bar{z}_{ft} , that satisfies the third constraint of P'_{MAGHP} will also satisfy the third constraint of P_{TP} . So

$$\begin{aligned} \sum_{t \in T_f^a, t \leq \tau} (\bar{z}_{ft} - \bar{z}_{f,t-1}) - \sum_{t' \in T_{f'}^a, t' \leq \tau - s_{f'} - (r_f - d_f)} (\bar{z}_{f't'} - \bar{z}_{f',t'-1}) \\ = \bar{z}_{f\tau} - \bar{z}_{f',\tau - s_{f'} - (r_f - d_f)} = \bar{z}_{f,t+(r_f - d_f)} - \bar{z}_{f,t-s_{f'}} \leq 0 \end{aligned}$$

So the point, \bar{z}_{ft} , satisfies the third constraint of P_{TP} and all of the constraints of

P_{TP} hold. Thus, the point \bar{z}_{ft} does indeed lie in the polyhedron P_{TP} . This establishes the relationship $P'_{MAGHP} \subseteq P_{TP}$.

Now we need to prove the relationship $P_{TP} \subseteq P_{VBO}$. To show this we will start with a feasible point in P_{TP} , \bar{z}_{ft} , and show that this is indeed feasible to P_{VBO} . Once again, the first two constraints and the last constraint are identical in the two models. So what remains to be shown is that any point, \bar{z}_{ft} , that satisfies the third constraint of P_{TP} will also satisfy the third and fourth constraints of P_{VBO} . So

$$g_f = \sum_{t \in T_f^a} t(\bar{z}_{ft} - \bar{z}_{f,t-1}) - r_f$$

$$\begin{aligned} g_{f'} &= d_f + s_{f'} + r_{f'} - g_f \\ &= \sum_{t \in T_{f'}^a} t(\bar{z}_{f't} - \bar{z}_{f',t-1}) - r_{f'} - d_f + s_{f'} + r_{f'} - \sum_{t \in T_f^a} t(\bar{z}_{ft} - \bar{z}_{f,t-1}) - r_f \\ &= -\bar{z}_{f',r'_{f'}} - \dots - \bar{z}_{f',r'_{f'} + \bar{T}_{f'} - 1} + r_{f'} + \bar{T}_{f'} - r_{f'} - d_f + s_{f'} + r_{f'} \\ &\quad + \bar{z}_{f,r_f} + \dots + \bar{z}_{f,r_f + \bar{T}_f - 1} - r_f - \bar{T}_f + r_f \\ &= -\bar{z}_{f',r'_{f'}} - \dots - \bar{z}_{f',r'_{f'} + \bar{T}_{f'} - 1} + \bar{T}_{f'} - d_f + s_{f'} + r_{f'} + \bar{z}_{f,r_f} + \dots + \bar{z}_{f,r_f + \bar{T}_f - 1} - \bar{T}_f \\ &\leq -(r_{f'} + \bar{T}_{f'} - 1 - (r_f - s_{f'} - (r_f - d_f)) + 1) + \bar{T}_{f'} - d_f + s_{f'} + r_{f'} \\ &\quad + (\bar{T}_f + r_f - 1 - r_f + 1) - \bar{T}_f \\ &= -(r_{f'} + \bar{T}_{f'} - r_f + s_{f'} + r_f - d_f) + \bar{T}_{f'} - d_f + s_{f'} + r_{f'} + 0 \\ &= -r_{f'} - s_{f'} + d_f - d_f + s_{f'} + r_{f'} \\ &= 0 \end{aligned}$$

where \bar{T}_f is maximum amount of time that flight f may arrive late. So all of the constraints hold and the point \bar{z}_{ft} does indeed lie in the polyhedron P_{VBO} . This establishes the relationship $P_{TP} \subseteq P_{VBO}$. ■

Appendix B

On the Non-integrality of the Polyhedron P_{MAGHP}

In this section we prove Theorem 1a, i.e., the polyhedron P_{MAGHP} is not integral, by providing the following example which has a fractional extreme point. Consider the case in which there are two flights arriving and being continued by two flights departing from a given airport during a restricted time window. The data of the problem is as follows:

$$|\mathcal{K}| = 1, \mathcal{T} = \{1, 2, 3, 4\}, \mathcal{C} = \{(1, 1), (2, 2)\},$$

i.e., the arriving flight i is continued by departing flight i . The turnaround times are $s_1 = 0, s_2 = 1$. The time windows are:

$$T_1^a = \{1, 2\}, T_2^a = \{1, 2\}, T_1^d = \{1, 2\}, T_2^d = \{2, 3\}.$$

Notice that flight 2 can only depart during time slots 2 and 3 since the turnaround time for the second flight is 1. The decision variables are:

$$y_{11}, y_{12}, y_{22}, y_{23}, z_{11}, z_{12}, z_{21}, z_{22},$$

with the interpretation that $y_{ij} = 1$ if flight i departs by time j and $z_{ij} = 1$ if flight i arrives by time j . Because of the time windows, we know that

$$y_{13} = 1, y_{24} = 1, z_{13} = 1, z_{23} = 1.$$

The capacities are:

$$D(1) = D(2) = D(3) = 1, A(1) = A(2) = A(3) = 1.$$

The resulting formulation (*MAGHP*) is:

$$\begin{aligned} y_{11} &\leq 1, & z_{11} + z_{21} &\leq 1, & y_{12} - y_{11} &\geq 0, & y_{23} - y_{22} &\geq 0, \\ y_{12} - y_{11} + y_{22} &\leq 1, & z_{12} - z_{11} + z_{22} - z_{21} &\leq 1, & z_{12} - z_{11} &\geq 0, & z_{22} - z_{21} &\geq 0, \\ 1 - y_{12} + y_{23} - y_{22} &\leq 1, & 1 - z_{12} + 1 - z_{22} &\leq 1, & y_{11} - z_{11} &\leq 0, & y_{22} - z_{21} &\leq 0, \\ & & & & y_{12} - z_{12} &\leq 0, & y_{23} - z_{22} &\leq 0 \end{aligned}$$

Letting

$$\begin{aligned} x &= (y_{11}, y_{12}, y_{22}, y_{23}, z_{11}, z_{12}, z_{21}, z_{22})' \\ b &= (1, 1, 0, 1, 1, -1, 0, 0, 0, 0, 0, 0, 0, 0)' \end{aligned}$$

and

$$A = \begin{pmatrix} y_{11} & y_{12} & y_{22} & y_{23} & z_{11} & z_{12} & z_{21} & z_{22} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{pmatrix}$$

the feasible space can be written as $Ax \leq b$.

Notice that matrix A is not totally unimodular since the submatrix consisting of the columns corresponding to the variables y_{12} , y_{22} , z_{12} , and z_{21} and the third, fifth, twelfth and thirteenth rows:

$$\begin{pmatrix} y_{12} & y_{22} & z_{12} & z_{21} \\ -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix}$$

has determinant of 2. The objective function

$$\text{Min} \quad 2 y_{11} - 4 y_{12} + 2 y_{22} - 6 y_{23} - 3 z_{11} + 6 z_{12} - 3 z_{21} + 6 z_{22}$$

gives an optimal solution of

$$\begin{aligned} y_{11} = 0 \quad y_{22} = 0 \quad z_{11} = \frac{1}{2} \quad z_{21} = \frac{1}{2} \\ y_{12} = \frac{1}{2} \quad y_{23} = \frac{1}{2} \quad z_{12} = \frac{1}{2} \quad z_{22} = \frac{1}{2} \end{aligned}$$

that shows that the polyhedron P_{MAGHP} is not integral. Furthermore, this is the objective function that is obtained when we let $c_f^g = 1$, $c_f^a = 3$ for all $f \in \mathcal{F}$. So, even with the restriction that $c_g = c_f^g$ and $c_a = c_f^a$ for all $f \in \mathcal{F}$, the polyhedron P_{MAGHP} is not integral.

Appendix C

Facet Defining Constraint Proofs

In this section we analyze the polyhedral structure of the $\text{conv}(IP_{MAGHP})$ and provide the proof of the first half of Theorem 1b that establishes which constraints are facets of $\text{conv}(IP_{MAGHP})$. The proof of the second half of Theorem 1b concerning problem (*TFMP*) is similar, but more algebraically involved. We first show that the constraint

$$\sum_{\{f:t \in T_f^d\}} (y_{ft} - y_{f,t-1}) \leq D_k(t), \quad \forall k \in \mathcal{K}, t \in \mathcal{F}$$

is not a facet of $\text{conv}(IP_{MAGHP})$ by constructing a counterexample with two flights, one arriving at airport k and one departing from airport k , three time periods and $D(t) = 1$, $A(t) = 1$. Then only the variables $y_{11}, y_{12}, y_{13}, z_{11}, z_{12}$, and z_{13} are defined. The complete

set of feasible solutions to IP_{MAGHP} is given by:

$$\begin{array}{cccccc} & y_{11} & y_{12} & y_{13} & z_{11} & z_{12} & z_{13} \\ \left(\begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right) \end{array}$$

In this case, $\dim(IP_{MAGHP}) = 5$ which can be determined by checking the rank of the matrix of solutions. We define the set

$$H_t = \{(y, z) \in IP_{MAGHP} : \sum_{\{f:t \in T_f^2\}} (y_{ft} - y_{f,t-1}) = 1\}, \text{ for some } t \in \mathcal{T}.$$

Then,

$$H_3 = \{(0, 0, 1, 0, 0, 1), (0, 0, 1, 0, 1, 1), (0, 0, 1, 1, 1, 1)\}.$$

In this case, the maximum number of affinely independent points in H_3 is less than the $\dim(IP_{MAGHP}) - 1$. We conclude that the constraint $\sum_{\{f:t \in T_f^2\}} (y_{ft} - y_{f,t-1}) \leq D_k(t), \forall k \in \mathcal{K}, t \in \mathcal{F}$ is not a facet. The same result can be checked in a similar manner for the constraint $\sum_{\{f:t \in T_f^2\}} (z_{ft} - z_{f,t-1}) \leq A_k(t), \forall k, t. \square$

For ease of exposition we consider instances of $(MAGHP)$ such that

- $|T_f|$ is that same for all f and therefore $D = \max_f |T_f| = |T_f|$,
- $s_f = 0, \forall f \in \mathcal{F}$,

- $A_k(t), D_k(t) \geq 1, \forall k, t.$

We consider an instance of $(MAGHP)$ with $|\mathcal{F}|$ flights in which $|\mathcal{C}|$ ($< |\mathcal{F}|$) of these flights are continued. These flights were arranged such that the first $|\mathcal{C}|$ flights are continued by flights $|\mathcal{C}| + 1, \dots, 2|\mathcal{C}| \leq |\mathcal{F}|$, with flight 1 being followed by flight $|\mathcal{C}| + 1$, flight 2 being followed by flight $|\mathcal{C}| + 2$, and so on.

We first determine $\dim(IP_{MAGHP})$ by constructing the following matrices of solutions, in which each row represents a solution to $(MAGHP)$, (see Figures C-1 and C-2). The rows of these matrices are affinely independent and there are $2|\mathcal{F}|D + 1$ such rows. So, we have exhibited $2|\mathcal{F}|D + 1$ affinely independent points in IP_{MAGHP} and thus, $\dim(IP_{MAGHP}) = 2|\mathcal{F}|D$.

We next consider the set

$$G_{ft} = \{(y, z) \in IP_{MAGHP} : y_{ft} - y_{f,t-1} = 0\}, \text{ for some } f \in \mathcal{F}, t \in \mathcal{T}.$$

If $f \in \{1, \dots, |\mathcal{C}|\}$ then there are four distinct solutions from the matrices of Figures C-1 and C-2 which do not belong to G_{ft} . For each of these rows, replace the 0 in the $y_{f,t-1}$ column with an 1.

If $f \in \{|\mathcal{C}| + 1, \dots, 2|\mathcal{C}|\}$ then there are two distinct solutions from Figures C-1 and C-2 which do not belong to G_{ft} . For each of these rows, replace the 1 in the $y_{f,t}$ column with a 0.

If $f \in \{2|\mathcal{C}| + 1, \dots, |\mathcal{F}|\}$ then there are two unique solutions from Figures C-1 and C-2 which do not belong to G_{ft} . For each of these rows, replace the 0 in the $y_{f,t-1}$ column with a 1.

For all of these cases, we have constructed a matrix with $|\mathcal{F}|D$ affinely independent rows, proving that $\dim(G_{ft}) \geq |\mathcal{F}|D - 1$. Since G_{ft} is a proper face of IP_{MAGHP} , we know that $\dim(G_{ft}) < \dim(IP_{MAGHP})$. So, $\dim(G_{ft}) = |\mathcal{F}|D - 1$ and thus, G_{ft} is a

facet of IP_{MAGHP} . \square

We next consider the set

$$K_{ft} = \{(y, z) \in IP_{MAGHP} : z_{ft} - z_{f,t-1} = 0\}, \text{ for some } f \in \mathcal{F}, t \in \mathcal{T}.$$

If $f \in \{1, \dots, |C|\}$ then there are three distinct solutions from the matrices of Figures C-1 and C-2 which do not belong to G_{ft} . For each of these rows, replace the 1 in the $y_{f,t}$ column with a 0.

If $f \in \{|C|+1, \dots, |F|\}$ then there is only one distinct solution from Figures C-1 and C-2 which does not belong to G_{ft} , so remove this row.

For each of these cases, we have constructed a matrix with $|\mathcal{F}|D$ affinely independent rows, proving that $\dim(K_{ft}) \geq |\mathcal{F}|D - 1$. Since K_{ft} is a proper face of IP_{MAGHP} , we know that $\dim(K_{ft}) < \dim(IP_{MAGHP})$. So, $\dim(K_{ft}) = |\mathcal{F}|D - 1$ and thus, K_{ft} is a facet of IP_{MAGHP} . \square

We next consider the set

$$M_{ft} = \{(y, z) \in IP_{MAGHP} : z_{ft} - y_{f,t-(r_f-d_f)} = 0\}, \text{ for some } f \in \mathcal{F}, t \in \mathcal{T}.$$

For all $f \in \{1, \dots, |\mathcal{F}|\}$ there are $t - \underline{T}_f + 1$ distinct solutions from the matrices of Figures C-1 and C-2 which do not belong to M_{ft} . For each of these rows replace the 0's in the columns corresponding to $z_{ft'}$, $t \leq t' \leq \bar{T}_f$ with 1's. \bar{T}_f and \underline{T}_f are the last possible and the earliest possible times that flight f could arrive, respectively.

The remaining matrix will have $|\mathcal{F}|D$ affinely independent rows, proving that $\dim(M_{ft}) \geq |\mathcal{F}|D - 1$. Since M_{ft} is a proper face of IP_{MAGHP} , we know that $\dim(M_{ft}) < \dim(IP_{MAGHP})$. So, $\dim(M_{ft}) = |\mathcal{F}|D - 1$ and thus, M_{ft} is a facet of IP_{MAGHP} . \square

Finally, we consider the set

$$N_{f'ft} = \{(y, z) \in IP_{MAGHP} : y_{ft} - z_{f't} = 0\}, \text{ for some } (f', f) \in \mathcal{C}, t \in \mathcal{T}.$$

For all $f \in \{1, \dots, |\mathcal{F}|\}$ there are $t - \underline{T}_f + 1$ distinct solutions from the matrices of Figures C-1 and C-2 which do not belong to $N_{f'ft}$. For each of these rows replace the 0's in the columns corresponding to $y_{ft'}, t \leq t' \leq \overline{T}_{ft}$ with 1's.

The remaining matrix will have $|\mathcal{F}|D$ affinely independent rows, proving that $\dim(N_{f'ft}) \geq |\mathcal{F}|D - 1$. Since $N_{f'ft}$ is a proper face of IP_{MAGHP} , we know that $\dim(N_{f'ft}) < \dim(IP_{MAGHP})$. So, $\dim(N_{f'ft}) = |\mathcal{F}|D - 1$ and thus, $N_{f'ft}$ is a facet of IP_{MAGHP} .

■

$\overbrace{0\dots 0}^{y_{1,t}}$	0	$\overbrace{0\dots 0}^{y_{C,t}}$	$\overbrace{0\dots 0}^{y_{C+1,t}}$	0	$\overbrace{0\dots 0}^{y_{2C,t}}$	0	$\overbrace{0\dots 0}^{y_{F,t}}$	$\overbrace{0\dots 0}^{z_{1,t}}$	0	$\overbrace{0\dots 0}^{z_{C,t}}$	$\overbrace{0\dots 0}^{z_{C+1,t}}$	0	$\overbrace{0\dots 0}^{z_{2C,t}}$	0	$\overbrace{0\dots 0}^{z_{F,t}}$
1...1	0	0...0	0...0	0	0...0	0	0...0	0...0	0	0...0	0...0	0	0...0	0	0...0
0...1	:	:	...	:	:	:	...	:	...
0...1	0	0...0	0...0	0	0...0	0	0...0	0...0	0	0...0	0...0	0	0...0	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	1...1	0...0	0	0...0	0	0...0	0...0	0	0...0	0...0	0	0...0	0	0...0
...	:	0...1	...	:	...	:	:	:	...	:	...
0...0	0	0...1	0	0	0...0	0	0...0	0...0	0	0...0	0...0	0	0...0	0	0...0
1...1	0	0...0	1...1	0	0...0	0	0...0	1...1	0	0...0	0...0	0	0...0	0	0...0
0...1	:	...	0...1	:	...	:	...	0...1	:	:	...	:	...
0...1	0	0...0	0...1	0	0...0	0	0...0	0...1	0	0...0	0...0	0	0...0	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	1...1	0...0	0	1...1	0	0...0	0...0	0	1...1	0...0	0	0...0	0	0...0
...	:	0...1	...	:	0...1	:	:	0...1	...	:	...	:	...
0...0	0	0...1	0...0	0	0...1	0	0...0	0...0	0	0...1	0...0	0	0...0	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	0...0	0...0	0	0...0	0	1...1	0...0	0	0...0	0...0	0	0...0	0	0...0
...	:	:	...	:	0...1	...	:	:	...	:	...
0...0	0	0...0	0...0	0	0...0	0	0...1	0...0	0	0...0	0...0	0	0...0	0	0...0

Figure C-1: First Half of Matrix of Solutions to IP_{MAGHP} .

$y_{1,t}$	$y_{C,t}$	$y_{C+1,t}$	$y_{2C,t}$	$y_{F,t}$	$z_{1,t}$	$z_{C,t}$	$z_{C+1,t}$	$z_{2C,t}$	$z_{F,t}$						
1...1	0	0...0	0...0	0	0...0	0	0...0	1...1	0	0...0	0...0	0	0...0	0	0...0
0...1	:	:	...	:	...	0...1	:	:	...	:	...
0...1	0	0...0	0...0	0	0...0	0	0...0	0...1	0	0...0	0...0	0	0...0	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	1...1	0...0	0	0...0	0	0...0	0...0	0	1...1	0...0	0	0...0	0	0...0
...	:	0...1	...	:	...	:	:	0...1	...	:	...	:	...
0...0	0	0...1	0	0	0...0	0	0...0	0...0	0	0...1	0...0	0	0...0	0	0...0
1...1	0	0...0	1...1	0	0...0	0	0...0	1...1	0	0...0	1...1	0	0...0	0	0...0
0...1	:	...	0...1	:	...	:	...	0...1	:	...	0...1	:	...	:	...
0...1	0	0...0	0...1	0	0...0	0	0...0	0...1	0	0...0	0...1	0	0...0	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	1...1	0...0	0	1...1	0	0...0	0...0	0	1...1	0...0	0	1...1	0	0...0
...	:	0...1	...	:	0...1	:	:	0...1	...	:	0...1	:	...
0...0	0	0...1	0...0	0	0...1	0	0...0	0...0	0	0...1	0...0	0	0...1	0	0...0
0...0	:	0...0	0...0	:	0...0	:	0...0	0...0	:	0...0	0...0	:	0...0	:	0...0
0...0	0	0...0	0...0	0	0...0	0	1...1	0...0	0	0...0	0...0	0	0...0	0	1...1
...	:	:	...	:	0...1	...	:	:	...	:	0...1
0...0	0	0...0	0...0	0	0...0	0	0...1	0...0	0	0...0	0...0	0	0...0	0	0...1

Figure C-2: Second Half of Matrix of Solutions to IP_{MAGHP} .

Bibliography

- [1] Ahuja, R.K., Orlin, J.B. and Magnanti, T.L. (1993), **Network Flows**, Prentice Hall, Englewood Cliffs, New Jersey.
- [2] Andreatta, G., and Brunetta, L. (1995), "Multi-Airport Ground Holding Problem: A Computational Evaluation Of Exact Algorithms", to appear in **Operations Research**.
- [3] Andreatta, G., Odoni, A.R., and Richetta, O. (1993), "Models for the Ground-Holding Problem", chapter in **Large-Scale Computation and Information Processing in Air Traffic Control**, L. Bianco and A. R. Odoni, editors, Springer-Verlag, Berlin, 1993, pp. 125-168.
- [4] Aronson, J.E. (1989), "A Survey of Dynamic Network Flows", **Annals of Operations Research**, 20, pp.1-66.
- [5] Ball, Michael (1993), personal communication.
- [6] Andreatta, G., and Tidona, G. (1994), "A new formulation for the Multi-Airport Ground-Holding Problem", Internal Report No. 3, Dip. di matematica Pura e Applicata, University of Padova, Italy.
- [7] Bertsimas, D. and Stock, S. (1994), "The air traffic flow management problem with enroute capacities", to appear in **Operations Research**.

- [8] Bookbinder, J.H. and Sethi, S.P. (1980) "The Dynamic Transportation Problem: A Survey," **Naval Research Logistics Quarterly**, 27, pp. 65-88.
- [9] Brunetta, L., Guastalla, G., and Navazio, L., (1996), "A new approach for solving the Multi-Airport Ground Holding Problem", to appear in **Annals of Operations Research**.
- [10] Everett, H., (1963) "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," **Operations Research**,3,pp. 399-417.
- [11] Fleischer, L. and Tardos, É. (1996), "Efficient Continuous-Time Dynamic Network Flow Algorithms," Technical Report TR1166, Cornell University, Department of Operations Research and Industrial Engineering.
- [12] Ford, L.R., and Fulkerson, D.R. (1958), **Flows in Networks**, Princeton University Press, Princeton, New Jersey.
- [13] Gale, D. (1959), "Transient Flows in Networks," **Michigan Mathematical Journal**, 6, pp.59-63.
- [14] Garey, Michael R. and Johnson, David S. (1979), **Computers and Intractability: A Guide to the Theory of NP-Completeness**, W.H. Freeman & Company, New York.
- [15] Gilbo, Eugene P. (1993), " Airport Capacity: Representation, Estimation, Optimization", **IEEE Transactions on Control Systems Technology**, Vol. 1, No. 3, September 1993, pp. 144-154.
- [16] Helme, Marcia (1992), "Reducing Air Traffic Delay in a Space-Time Network", **Proceedings of the 1992 IEEE International Conference on Systems, Man and Cybernetics**, Chicago, pp. 236-242.
- [17] Hoppe, Bruce (1995), " Efficient Dynamic Network Flow Algorithms," Ph.D. Thesis, Dept. of Computer Science, Cornell University, Ithaca, New York.

- [18] Hoppe, B. and Tardos, É. (1994), "Polynomial time algorithms for some evacuation problems," **Proceedings of the 5th Annual ACM-SIAM Symposium on Discrete Algorithms**, pp. 433-441.
- [19] Hoppe, B. and Tardos, É. (1995), "The quickest transshipment problem," **Proceedings of the 6th Annual ACM-SIAM Symposium on Discrete Algorithms**, pp. 512-521.
- [20] Lindsay, Kenneth S., Boyd, E. Andrew, and Burlingame, Rusty (1993), "Traffic Flow Management Modeling with the Time Assignment Model," **Air Traffic Control Quarterly**, vol.1, no. 3.
- [21] Matos, P., and Ormerod, R., (1995), "The Application of Operational Research to European Air Traffic Flow Management - Understanding the Context", Research Paper No. 200, Operational Research and Systems Group, Warwick Business School, Warwick, U.K.
- [22] Milner, Joseph (1995), "Dynamic Slot Allocation with Airline Participation," Ph.D. Thesis, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts.
- [23] Minieka, E. (1973), "Maximal, Lexicographic, and Dynamic Network Flows," **Operations Research**, 21, pp. 517-527.
- [24] Minieka, E. (1974), "Dynamic Network Flows with arc changes," **Networks**, 4, pp. 255-265.
- [25] Nemhauser, George and L. Wolsey (1988), **Integer and Combinatorial Optimization**, John Wiley & Sons, New York, pp. 535-572.
- [26] Odoni, A. R. (1987), "The Flow Management Problem in Air Traffic Control", Flow Control of Congested Networks, A.R. Odoni and G. Szego, editors, Springer-Verlag, Berlin (pp. 269-288).

- [27] Powell, W.B., Jaillet, P. and Odoni, A. "Stochastic and dynamic networks and routing." **handbooks in Operations Research and Management Science: Networks**. Ball, M.O., Magnanti, T.L., Monma, C.L. and Nemhauser, G.L., editors. Elsevier Science Publishers B.V., 1995.
- [28] Pugh, G.E. (1993), **Value-Driven Methods of Decision and Control**, draft of a textbook, p. 3-5.
- [29] Richetta, O. and A. R. Odoni (1993), "Solving Optimally the Static Ground-Holding Policy Problem in Air Traffic Control", **Transportation Science**, 27, pp. 228-238.
- [30] Richetta, O. and A. R. Odoni (1994), "Dynamic Solution to the Ground-Holding Policy Problem in Air Traffic Control", to appear in **Transportation Research**.
- [31] Terrab, M. and A. R. Odoni (1993), "Strategic Flow Control on an Air Traffic Network", **Operations Research**, pp. 138-152.
- [32] Terrab, M. and Paulose, S. (1993), "Dynamic Strategic and Tactical Air Traffic Flow Control", RPI Technical Report.
- [33] Vancheieri, A.R., (1997), "Aviation Dream Team", **ORMS Today**, 24, No. 2, pp.38-41.
- [34] Vranas, P., (1995), "Optimal Slot Allocation for European Air Traffic Flow Management", Working Paper, German Aerospace Research Establishment, Bonn, Germany.
- [35] Vranas, P., D. Bertsimas and A. R. Odoni (1994a), "The Multi-Airport Ground-Holding Problem in Air Traffic Control", **Operations Research**, pp. 249-261.
- [36] Vranas, P., D. Bertsimas and A. R. Odoni (1994b), "Dynamic Ground-Holding Policies for a Network of Airports", to appear in **Transportation Science**.

- [37] Wald, Matthew L., "Austere Future Looms for F.A.A.", **New York Times**, October 15, 1995, p. A1.
- [38] Wilkinson, W.L. (1971), "An Algorithm for Universal Maximal Dynamic Flows in a Network," **Operations Research**, 19, pp. 1602-1612.