MASTER'S THESIS

Optimization Model with Fairness Objective for Air Traffic Management

by T. Butler
Advisor: M. Ball

NEXTOR M.S. 98-1
(ISR M.S. 98-5)

NEXTOR

National Center of Excellence in Aviation Operations Research

The National Center of Excellence in Aviation Operations Research (NEXTOR) is a joint university, industry and Federal Aviation Administration research organization. The center is supported by FAA research grant number 96-C-001.

Web site http://www.isr.umd.edu/NEXTOR/

ABSTRACT

Title of Thesis: OPTIMIZATION MODEL WITH FAIRNESS OBJECTIVE FOR AIR TRAFFIC MANAGEMENT

Degree Candidate: Taryn Darnella Butler

Degree and year: Master of Arts, 1998

Thesis directed by: Professor Michael Ball
Robert H. Smith School of Business

With the ever increasing congestion at airports around the world, studies into ways of minimizing delay costs on the ground while meeting the goals of the airlines are necessary. When arrival capacities are reduced at major airports, the Federal Aviation Administration (FAA) issues revised departure/arrival times to prevent congestion at restricted airports. This is referred to as the National Ground Delay Program Problem. A new approach to developing ground delay programs, called Collaborative Decision Making (CDM), is being developed. CDM goals include more information exchange and greater participation on the part of the airlines in determining landing slot allocations. This thesis develops a model specifically for the CDM setting. A key element is the inclusion of a fairness criterion within the underlying optimization model. The fairness criterion seeks to "pay back" an airline for time slots that it is owed but cannot make use of due to mechanical or other difficulties. It also attempts to provide incentives to the airlines to increase the exchange of information. This thesis investigates the Ground Delay Problem relative to a single airport. Different formulations of the integer programming model are given that take into account airport capacities and airline goals and experiments are conducted with realistic data to determine the solvability of the problem. Results for this model are compared with output from the Flight Schedule Monitor (FSM), the CDM decision support tool.

OPTIMIZATION MODEL WITH FAIRNESS OBJECTIVE FOR AIR TRAFFIC MANAGEMENT

by

Taryn Darnella Butler

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland at College Park in partial fulfillment
of the requirements for the degree of
Master of Arts
1998

Advisory Committee:

Professor Michael Ball, Chairman/Advisor
Professor Frank Alt
Professor Peter Wolfe

ACKNOWLEDGEMENTS

Special thanks are extended to Thomas Vossen and Bob Hoffman for their assistance with all of the computer programming required for the completion of this thesis. I would also like to thank Professor Ball for his patience.

This work was supported in part by the Federal Aviation Administration through NEXTOR, the National Center of Excellence in Aviation Operations Research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

1. AAL - American Airlines
2. AAR - Arrival Acceptance Rate
3. AMPL - A Mathematical Programming Language (software)
4. ATCSCC - Air Traffic Control System Command Center
5. CDM - Collaborative Decision Making
6. CPLEX - linear/integer programming software
7. CTA - Controlled Time of Arrival
8. CTD - Controlled Time of Departure
9. EDCT - Estimated Departure Clearance Time (EDCT)
10. ETA - Estimated Time of Arrival
11. EWR - Newark International Airport
12. FAA - Federal Aviation Administration
13. FSM - Flight Schedule Monitor
14. GDP - Ground Delay Program
15. G-demand - Goal-demand
16. LAX – Los Angeles International Airport
17. LP - linear program
18. MAGHP - multi-airport ground-holding problem
19. NAS - National Airspace System
20. OAG - Official Airline Guide
21. RBS - Ration-By-Schedule

22. RTA - Revised Time of Arrival

23. UAL - United Airlines

24. USA – US Airline

Chapter 1

Introduction

The Federal Aviation Administration (FAA) is sometimes forced to respond to congestion at airports by putting restrictions on users of the National Airspace System (NAS). Limited airport capacity, specifically the maximum number of arrivals that can be performed during a fixed time interval at a given airport, is the major cause of congestion. This capacity, called the airport acceptance rate, fluctuates due to weather conditions, runway configuration, and other factors. Air traffic flow management strives to reduce congestion delay effects while maintaining an efficient and safe utilization of the NAS. Until 1981 [1], aircraft were routinely allowed by the FAA to take off whenever they were ready; if there was congestion at the destination terminal, they were placed in holding patterns until they were able to land (or until they ran low on fuel and were directed to an alternative airport). Flow management attempts to reduce the congestion and allocates necessary delays elsewhere in the NAS by using a combination of techniques. This in turn reduces the number of airborne queues. Glockner, in [2], defines flow management as an efficient use of congested airspace and airports, which minimizes the number of aircraft waiting at any single facility. Ground delays, enroute speeding, and enroute slowing (vectoring) are just some of the techniques used.

Solutions to these congestion problems depend on the time horizon. Long term approaches include construction of additional airports and additional runways at existing airports, improved air traffic control technologies and procedures, and use of larger aircraft. The new Denver International Airport is an example of the construction of an additional airport and airlines are currently using larger aircraft in some cases. Medium-term approaches include modification of the temporal pattern of aircraft flow to eliminate periods of “peak” demand as defined by [3]. Short-term approaches have a planning horizon of 6-12 hours and include ground delay programs, which are the focus of this thesis.

A short- term solution for a single airport is given in this thesis. The goal is to develop a model specifically for the newly developed Collaborative Decision Making (CDM) setting.

The remainder of this thesis is organized as follows. In Chapter 2, background on Ground Delay Programs (GDP) and CDM is given and an illustration of a GDP is provided. Grover Jack, the procedure currently in use by the FAA, is explained. We next describe the CDM procedure compression and ration-by-schedule (RBS), which are developed by the users to address issues of fairness. We extract from these procedures a definition of fairness. Subsequently, an example is given of these procedures.

Mathematical models are presented in Chapter 3. We initially describe the OPTIFLOW model which was developed in the mid-90’s and which serves as the basis for our new models. The next section presents two formulations that include considerations of fairness. The first formulation is the integer programming model and the second is a multicommodity flow formulation.

Chapter 4 documents the results of multiple experiments done to evaluate the models. The software used to conduct the experiments was AMPL, CPLEX and FSM. A brief explanation of each software package is given in this chapter and simple examples are provided. Historical data was acquired from outside sources to run the tests. This allowed a comparison of output from our model with output from FSM.

Chapter 2

Background

2.1 Ground Delay Program and Collaborative Decision Making

After the air traffic controllers' strike in 1981, the FAA was forced to introduce the Ground Delay Program (GDP) concept which responds to reductions in the arrival capacity at one or more major airports. When large delays are forecasted, the Air Traffic Control System Command Center (ATCSCC) imposes ground delays on particular flights prior to departure. This ensures that planes are not allowed to take off until there is a high probability that they can complete their flight without significant delays. ATCSCC monitors airports throughout the U.S. for capacity-demand imbalances. A GDP is motivated by the fact that, as long as delay at the airport of destination is unavoidable, it is both less costly and safer to absorb the delay on the ground before take-off, rather than in the air. GDPs are executed when factors, such as inclement weather or the closing of a runway, cause congestion.

The current GDP process has come under scrutiny and is currently being revamped by a cooperative effort known as Collaborative Decision Making (CDM). This program is a joint FAA/industry initiative aimed at improving Traffic Flow Management through increased information exchange and improved collaboration. The proposed set of CDM procedures is explained in [12] and [15]. Hoffman states that flights will initially be assigned to time slots on a first-scheduled, first-assigned basis. The FAA generates an initial allocation using a procedure called ration-by-schedule. Then, in an iterative exchange between the airlines and the ATCSCC, each airline will have the opportunity to reassign some of its flights to the arrival slots it has been allocated, thus giving the airlines greater control over the economic impacts of a GDP. As part of the "cancellation and substitution" process, the airlines may both cancel flights and rearrange the assignment of flights to time slots. Lastly, the FAA eliminates any "holes" in the schedule using the compression algorithm.

The CDM web page [5] provides the two central tenets to the CDM. They are: (1) better information will lead to better decision making, and (2) tools and procedures need to be in place to enable the ATCSCC and the NAS users to more easily respond to the changing conditions. The near-term CDM program focuses on airport arrival demand and those instances that usually require some type of ground holding strategy. Longer-term objectives include using CDM to make route allocation decisions and distribute information on the status of the NAS.

2.2 GDP Processes

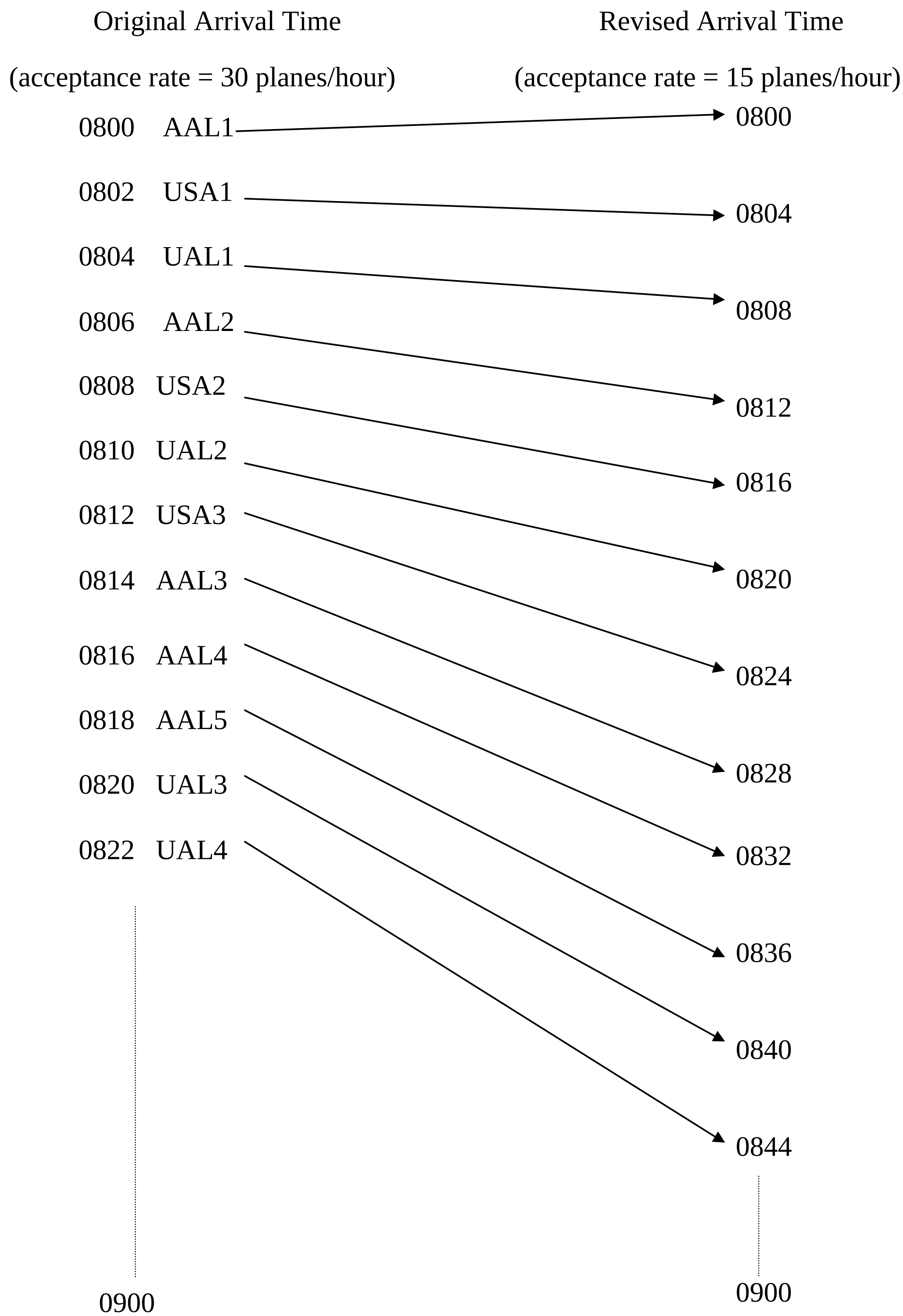
GDPs essentially place NAS users into a state of irregular operations. Airlines respond by rescheduling, canceling, or substituting flights. The cancellation and substitution processes allow scheduled airlines to mitigate the adverse effects of ground delays.

Cancellation and substitution are specific GDP processes. We now illustrate current GDP procedures with a simple example. Suppose a GDP is invoked. That is, the arrival airport's capacity was reduced resulting in the delay of flights arriving there. The delayed flights are held on the ground at their current airport. This is the process currently in use by the ATCSCC. It is known as Grover Jack. Grover Jack is the process of delaying flights while preserving their order.

Suppose at Reagan National Airport (DCA), United Airlines (UA), for example, had 10 arrival time slots in the first hour, 0800, with an arrival acceptance rate (AAR) of 30 flights per hour. Now suppose, due to bad weather, this rate was cut in half to 15 flights per hour (Figure 1).

Grover Jack solves this problem by considering the estimated time of arrival (ETA) for each flight. These are converted to departure times by subtracting en-route times. We stretch the number of arrivals out over a period of time, thus preserving the order of arrivals. The Grover Jack process, currently in use by the FAA, is based on this concept. Then, in the example above, United should receive 10 arrival slots in the first 2-hour period. The list of flights are given controlled times of arrival (CTA) such that, for example, the first 4 flights are assigned to the first 4 slots, the next 4 flights to the next 4 slots, and so on, stretching the flights out over time.

Figure 1: Grover Jack Solution



Cancellation and substitution processes and the impact of "other delays" may actually make it impossible for airlines to utilize the time slots allocated. After cancellations, substitutions, and other delays, some slots will be left open since a flight cannot be assigned to a time slot earlier than its ETA. Once the FAA issues revised departure/arrival times, the airlines can propose changes by canceling flights and then substituting a flight into an open slot created by the cancelled flight.

This process is complicated by the fact that flights are cancelled or delayed due to other reasons. If mechanical delays occur prior to a GDP being run, then with the current approach, additional delays will be incurred. This is the so-called double penalty issue. If United must delay one of their flights, say *f*, by 1 hour due to mechanical problems and if the FAA is informed of this, then *f* is moved down the list prior to running Grover Jack. For example, its original 12:00 ETA is updated to 1:00, 1 hour later. If a GDP is issued with a 30 minute delay assigned to *f*, then *f* would be given a 1:30 CTA, 30 minutes later than the updated ETA. Flight *f* would receive 60 min + 30 min =1 ½ hours of delay. Thus, it appears they are being penalized twice. There is a consensus among airlines that the ETA should not be used, but rather the original time of arrival.

Grover Jack is simply an order preserving schedule. Thus, it is necessary to incorporate the issue of fairness into the model. The G-Demand model introduced in this thesis will consist of an integer programming model that will:

- "pay back" an airline for time slots that it is owed but cannot make use of due to mechanical delays or other difficulties;
- make use of other objectives and constraints, designated as OPTIFLOW.

It is hoped that this will produce a solution that is fairer and provide incentives for the airlines to provide the FAA with current flight information.

2.2.1 Example of the Cancellation/Substitution Processes

Suppose a GDP is invoked with an acceptance rate of 12 arrivals per hour while preserving the original order of the arrivals (Grover Jack). Each flight in a GDP is assigned a controlled time of arrival (CTA) and a controlled time of departure (CTD). Once the CTA is fixed and since travel times can be predicted with great accuracy, the CTD and the amount of assigned delay are easily computed: the CTD is CTA minus the en route time and the ground delay is the CTD minus the scheduled arrival time. Thus, a feasible solution to the single-airport ground-holding problem can be derived once each flight has been assigned a CTA. We need only deal with arrival times when formulating our models. As a result, the airline has a flight list consisting of all flights scheduled to arrive at the airport during the GDP, an arrival slot or CTA for each flight, and a corresponding departure time known as CTD. Grover Jack takes the original flight arrival order and spaces these flights so that they exactly meet the degraded rate. In this example the rate revised AAR is 12 flights per hour, compared to 24 flights originally, so there should be 5 minutes between arrivals (See Table 1).

Table 1: GDP generated delays using Grover Jack

Airline	Flt No	ETA	CTA	Delay
A	1	0700	0700	0
A	2	0700	0705	5
B	3	0705	0710	5
B	4	0705	0715	10
B	5	0710	0720	10
B	6	0710	0725	15
A	7	0710	0730	20
C	8	0720	0735	15
B	9	0740	0740	0
C	10	0740	0745	5
A	11	0830	0830	0
Total				**85**

ETA - Estimated Time of Arrival: the original arrival time
CTA - Controlled Time of Arrival: arrival slots assigned after GDP by the existing Estimated Departure Clearance Time (EDCT) software

There is a total of 85 minutes of delay assigned. Now assume Flight 1 is cancelled and removed from the list (See Table 2). Then Flight 2 will take the 0700 CTA slot and we continue in the same manner as above. This gives us a total delay of 50 minutes. It appears that the entire system benefited from this delay. Looking deeper, we see that Airline B benefited the most from Airline A's cancellation. Airline B saved 20 minutes but Airline A only saved 10 minutes.

Table 2: Revised delay times after Flight 1 is cancelled

Airline	Flt No	ETA	CTA	Delay
A	1	0700	-	-
A	2	0700	0700	0
B	3	0705	0705	0
B	4	0705	0710	5
B	5	0710	0715	5
B	6	0710	0720	10
A	7	0710	0725	15
C	8	0720	0730	10
B	9	0740	0740	0
C	10	0740	0745	5
A	11	0830	0830	0
Total				**50**

Consider what happens if Airline B went through with its normal substitution process (See Table 3). Flight 4 is cancelled and Flight 5 is substituted into Flight 4's CTA slot. This substitution is allowed since Flight 5's ETA is earlier than the CTA of Flight 4. Flight 6 then uses Flight 5's CTA slot. The revised delay (Rdly) column now shows a total delay of 55 minutes. By removing the cancellation, total delay was 50 minutes and Airline B had 20 minutes of ground delay. Using Table 3, total delay was 55 minutes and Airline B had 20 minutes of delay again. It is clear from these examples how airline and traffic management objectives, maximizing efficiency and satisfying user preferences, could come in conflict.

Table 3: Substitution process for Airline B

Airline	Flt No	ETA	CTA	Delay	Can/Sub	RTA	Rdly
A	1	0700	0700	0		0700	0
A	2	0700	0705	5		0705	5
B	3	0705	0710	5		0710	5
B	4	0705	0715	10	C	-	-
B	5	0710	0720	10	S	0715	5
B	6	0710	0725	15		0720	10
A	7	0710	0730	20		0725	15
C	8	0720	0735	15		0730	10
B	9	0740	0740	0		0740	0
C	10	0740	0745	5		0745	5
A	11	0830	0830	0		0830	0
Total							**55**

RTA - Revised Time of Arrival

To illustrate how the double penalty works, consider what would happen if Airline A, Flight 1 has a mechanical delay of 30 minutes and the ATCSCC issues a GDP with an AAR of 6 flights per hour. The new schedule is shown in Table 4. The column labeled ETA_1 contains the new ETA for Airline A, Flight 1.

Table 4: Mechanical delay

Airline	Flt No	ETA	ETA_1
A	1	0700	0730
A	2	0700	0700
B	3	0705	0705
B	4	0705	0705
B	5	0710	0710
B	6	0710	0710
A	7	0710	0710
C	8	0720	0720
B	9	0740	0740
C	10	0740	0740
A	11	0830	0830

Table 5 reorders the flights according to the new ETAs. Notice that Airline A, Flight 1 has moved further down the list. If a GDP is invoked with the given AAR, we obtain a CTA for each flight, given in the next to last column. Airline A, Flight 1 receives an additional 40 minutes of delay after the 30 minute mechanical delay. This is what is known as a double penalty.

Table 5: Mechanical delay with double penalty

Airline	Flt No	ETA_1	CTA	Delay
A	2	0700	0700	0
B	3	0705	0710	5
B	4	0705	0720	15
B	5	0710	0730	20
B	6	0710	0740	30
A	7	0710	0750	40
C	8	0720	0800	40
A	1	0730	0810	40
B	9	0740	0820	40
C	10	0740	0830	50
A	11	0830	0840	10
Total				**290**

These examples illustrate that individual airline criteria can conflict with an objective of the traffic flow management provider: to maximize system efficiency. The following section discusses ways to reconcile these conflicting objectives.

2.3 Fairness

The concept of fairness is that in an "ideal" GDP a given airline over any time period should receive a percentage of available time slots equal to the percentage "owned" by that airline in the OAG schedule. The notion of “owning” a time slot was agreed upon by participants in CDM; it is an idea presently being tested for use. Ration by schedule (RBS), a CDM program element, allocates slots based on this concept.

Currently, if an airline reports a mechanical delay in advance of a GDP, those flights would simply be dropped from the database. The airline would not be able to use their assigned arrival slots for substitution. The system will re-project the delayed flights' arrival times. If a GDP were run at that time, that flight could receive an additional delay on top of its mechanical delay. This double penalty clearly produces adverse economic consequences resulting in the airlines holding back pertinent information.

2.3.1 Ration-by-Schedule (RBS) and Compression Algorithms

RBS and compression remove the disincentive to provide accurate information. When arrival capacity is reduced, the limited arrival resources must be rationed. RBS assigns new arrival times to a set of flights. For scheduled carriers, the rationing should be based upon the original schedule, and not the current projections of demand. Here the standard schedule is the Official Airline Guide (OAG) schedule. The preservation of fairness and providing airlines with an incentive to provide accurate schedule information is essential.

RBS Algorithm

The purpose of RBS is to ration arrival slots according to the original scheduled arrival times and to serve as an initial assignment of CTAs for subsequent rounds of collaboration between the airlines and the FAA. The key difference between RBS and Grover Jack is Grover Jack is based on the current adjusted schedule and RBS is based on the OAG; flights delayed for other reasons are handled in a fair way given this allocation approach. RBS fixes the number of slots owned by an airline in the following manner.

In Table 5, Airline A, Flight 1 is delayed but under RBS, Airline A owns that first slot in addition to the second time slot. Thus Flight 2, or some other flight of that airline, may move up to the first slot provided that it is a feasible time. If Flight 1 is not cancelled, it is free to move to a feasible slot vacated by another flight. Subsequently, each airline has the opportunity to minimize delays during a GDP. Therefore, even though a flight has a mechanical delay, the airline still owns the original allocated slots. Another part of the solution to the double penalty issue is the compression algorithm. The compression algorithm from [5] is outlined below.

Compression Algorithm

After cancellations and substitutions, quite often there are gaps of time in the schedule where no flights are scheduled to arrive (see Table 3). This is a result of the number of flights being reduced. Compression assigns flights to these empty time slots by moving them up in the schedule where feasible.

Ultimately, when a slot is left open, compression attempts to assign another flight of that airline to that slot. If there is no flight available then compression will search within another airline for a feasible solution or declare the slot unusable. The algorithm is described below.

Based on an estimate of reduced capacity as reflected in the AAR, resources are rationed according to the original schedule. The rationing procedure could be the current GDP or some other method that allocates airport arrival resources (arrival slots) to users in some fair fashion. This step is analogous to RBS.

Step 1: Intra-airline mapping - In this step, scheduled updates (cancellations and delays) are applied and the new schedule is mapped to the original set of arrival slots in a manner that minimizes total delay for each user. This can be accomplished centrally or individual users can accomplish this mapping through their own substitution process.

Step 2: Compression(inter-airline mapping) - Identify a vacant slot (resulting from a cancellation, the end of a cancellation/substitution string, a delay, or the result of an airline delay where the arrival slots cannot be fully utilized through the exchange process) and the owner of that slot. Identify the owner of the slot and label the slot time as T*.

Step 2a: Search for a flight belonging to that user (or an express carrier of that user) that can be moved into that slot. Eligible flights must meet the following criteria:

1. The original time estimates appearing in the ETA column cannot occur later than the CTA of the available slot.
2. Delay reduction of the eligible flight must be greater than or equal to D=1 minute (Airlines are suggesting changing to D=10 minutes.)
3. The new EDCT of the eligible flight must occur at least *x* minutes after the present time (*x*=30 minutes currently) to allow prior notice to airlines.

If an eligible flight is found, move it into that slot, and set T* equal to its previous CTA. Return to 2a to fill this vacancy. If no eligible flight is found, go to 2b.

Step 2b: Search for the first flight of another user that can be moved into the vacant slot. The eligibility criteria are the same as 2a except for 1. ETA.

If no flight is eligible, terminate, and return to step 2. Else, identify the moved flight's previous CTA as T* and go to 2a.

The results from applying this algorithm to Table 3 are shown in Table 6 below.

Table 6: Compression algorithm delays

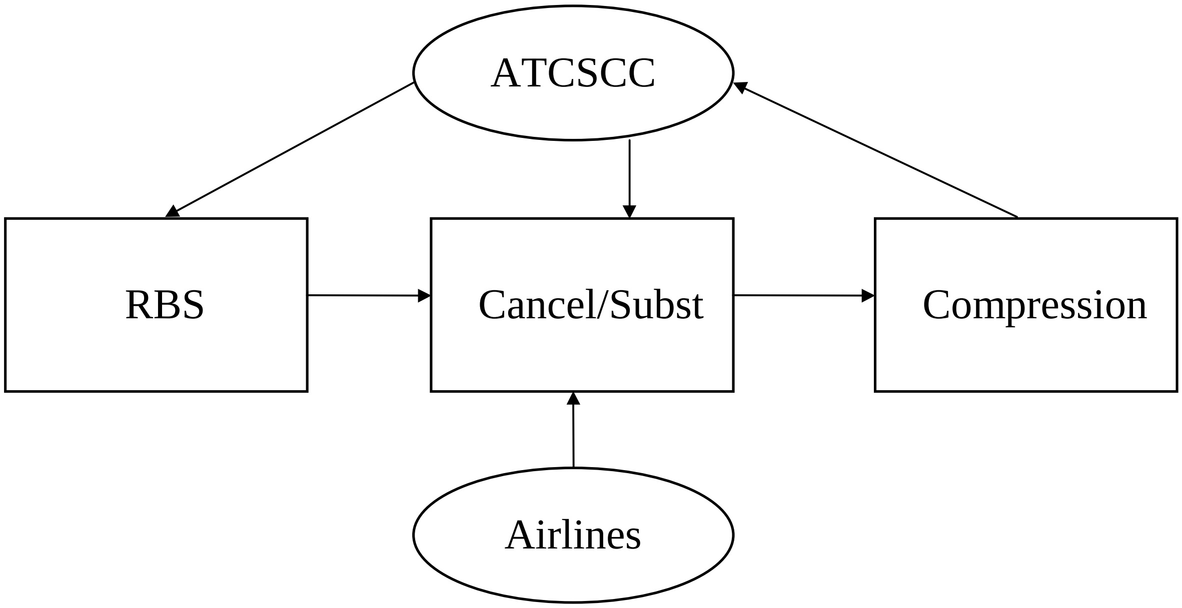
Airline	Flt No	ETA	CTA	Delay	Can/Sub	New CTA	Rdly
A	1	0700	0700	0	C	-	-
A	2	0700	0705	5	S	0700	0
B	3	0705	0710	5		0705	0
B	4	0705	0715	10		0715	10
B	5	0710	0720	10		0720	10
B	6	0710	0725	15		0725	15
A	7	0710	0730	20		0710	0
C	8	0720	0735	15		0730	10
B	9	0740	0740	0		0740	0
C	10	0740	0745	5		0745	5
A	11	0830	0830	0		0830	0
Total							**50**

The arrows show the substitutions made for each flight.

This illustrates the most efficient solution, which does not penalize the airline that substituted. This algorithm yields no delay for Airline A and a total delay of 50 minutes. Since Flight 7 could not make use of Flight 2's CTA, Flight 3 used Flight 2's vacated slot. Flight 7 could use the slot vacated by Flight 3 so it was given a new CTA of 0710. By moving Flight 7 of Airline A up, this provides Airline A with an incentive to provide accurate flight information. Also, Flight 8 used the slot vacated by Flight 7.

The following flow chart from Hoffman depicts the process of decision making by the ATCSCC and the industry described above:

Figure 2: Cycle of Decision Making



Chapter 3

Mathematical Models

All of the models presented here are for a single airport. The single airport problem can be a building block for the more complex multi-airport problem. Arrival operations at a determined destination airport are considered during a specific time interval. The time interval is then discretized into time periods with deterministic arrival capacities.

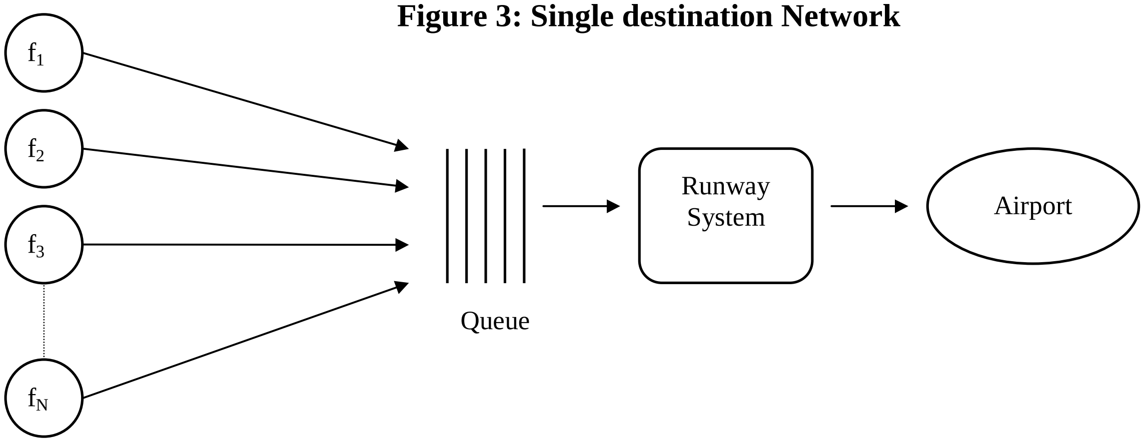
In [6], Andreatta, Odoni, and Richetta explain the different versions of the GDP. The models we present are deterministic and static. Deterministic models are currently the only problems being studied. Airport capacities are fixed values in this case. Static models may be used when there are significant lags in updating capacities or weather information, alternatively a ground delay is strategically planned at a single point in time (the beginning of the day) and revised marginally from that point on. As noted above, multiple time periods are used also.

A simplified model of the GDP is given in [6]. The macro model of the single-destination network (Figure 3) captures the essential elements needed to solve the GDP:

i. N flights ($f_1, \ldots, f_N$) are scheduled to arrive at the airport
ii. The airport is the only capacitated element of the network and thus the only source of delays.
iii. Departure and travel times are deterministic and known in advance.
iv. The time interval, [0, B], is discretized into I equal time periods numbered 1, 2,…, I, with the earliest arrival for the airport scheduled at 0 and the latest arrival scheduled at B.

v. Delay cost functions for each flight are known.

Figure 3: Single destination Network



3.1 OPTIFLOW Model

The OPTIFLOW model is the basis for all the models discussed in this thesis. OPTIFLOW is a formal model that minimizes delay costs while (1) satisfying the airport capacity, (2) ensuring all flights, not cancelled, arrive at the airport, and (3) satisfy any banking constraints. Banking constraints accommodate the hubbing operations of major airlines. Discussion of OPTIFLOW and some of its enhancements is given in [7] and [8]. See [12] or [15] for more discussion on banking constraints.

3.1.1 Formulation

Consider the set of airlines A={1,2,3,…,*a*}. For each airline there are corresponding flights F={1, 2, 3,…,N}. Let *O(a)* be the set of flights owned by airline *a*. There is also a set of time intervals, I={1, 2, 3,…,*i*}.

Data

v_{fi}^{a} = 1 if flight *f* of airline *a* arrives in interval *i*; 0 otherwise

c_{fi}^{a} = cost of flight *f* of airline *a* arriving in interval *i*

d_i = capacity for interval *i*

i_f = the time interval for flight *f* in the original schedule

Objective Function: Minimize delay costs.

Minimize $\sum_{a=1}^{A}\left(\sum_{f\in O(a)}\sum_{i=i_f}^{I} c_{fi}^{a} v_{fi}^{a}\right)$

Constraints:

(1) Each arrival time period is allowed (at most) a reduced number of flights.

(2) All flights, not cancelled, are assigned to some arrival time period.

Subject to:

(1) $\sum_{a}\sum_{f} v_{fi}^{a} \le d_i \qquad \forall i$

(2) $\sum_{i=i_f}^{I} v_{fi}^{a} = 1 \qquad \forall a, \forall f$

(3) $v_{fi}^{a} \in \{0,1\}$

where $c_{fi}^{a} = W_f\left(i - \hat{i}_f\right)^{1+\delta}$ with W_f a weight associated with flight *f* and $\delta < 1$ a positive number. The parameter $1+\delta$ is used for superlinear growth in the cost of tardiness of a flight so that the model tends to favor assigning a moderate amount of delay to two flights rather than the assigning of a large amount of delay to one and a small amount to another. Consider an airline with two flights, f_1 and f_2, that will be assigned delay. Suppose the choice is between f_1 and f_2 being assigned 30 and 120 minutes of delay, respectively, or being assigned 60 minutes of delay each. The model will choose the latter.

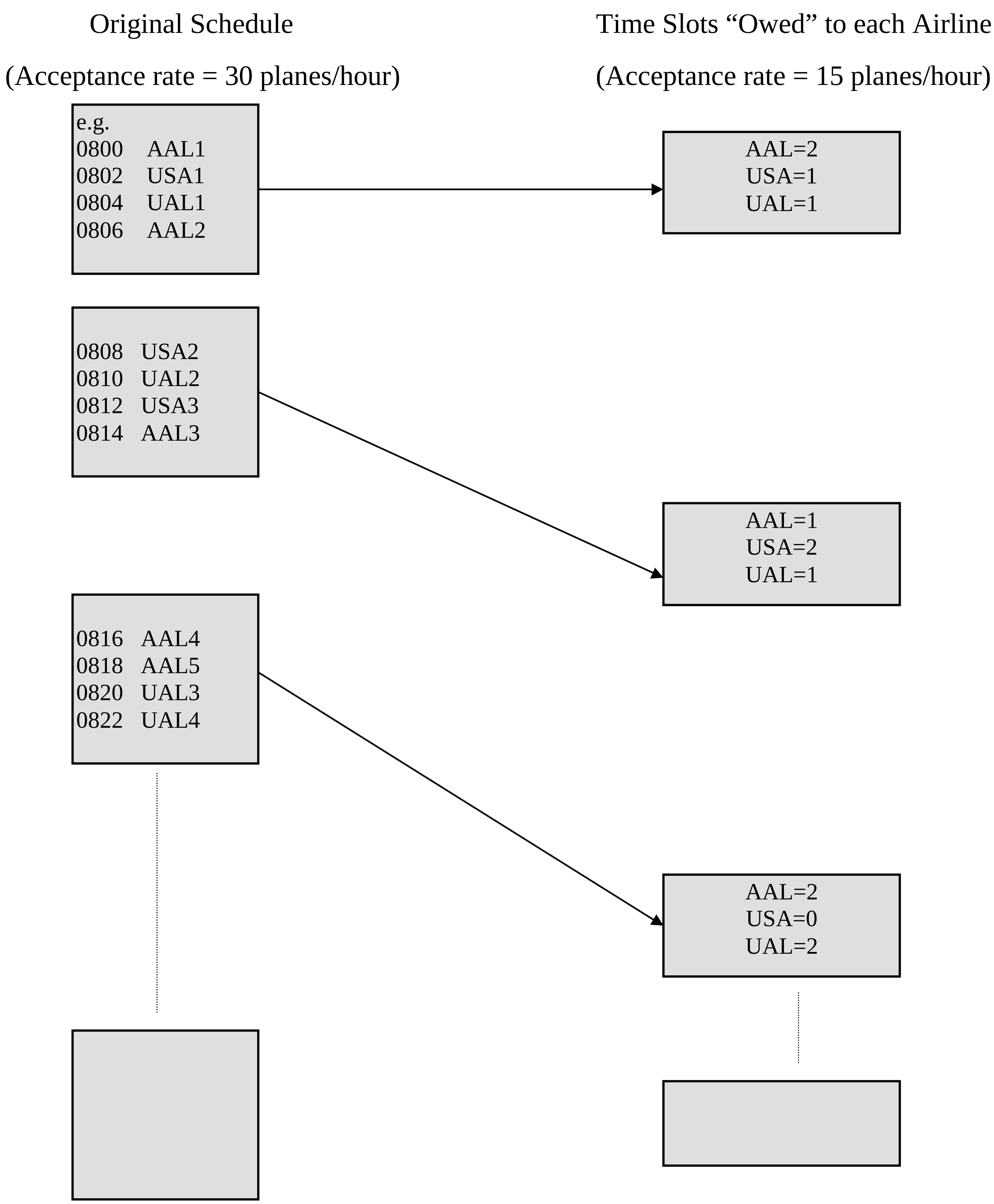
If $W_f = 1$, then the Grover Jack solution will be obtained. If more general weights are used, other solutions could be generated.

Observe that the OPTIFLOW model is a special case of the multi-airport ground-holding problem (MAGHP) given by Vranas, Bertisimas, and Odoni in [3] (See appendix).

3.2 Goal-Demand Model

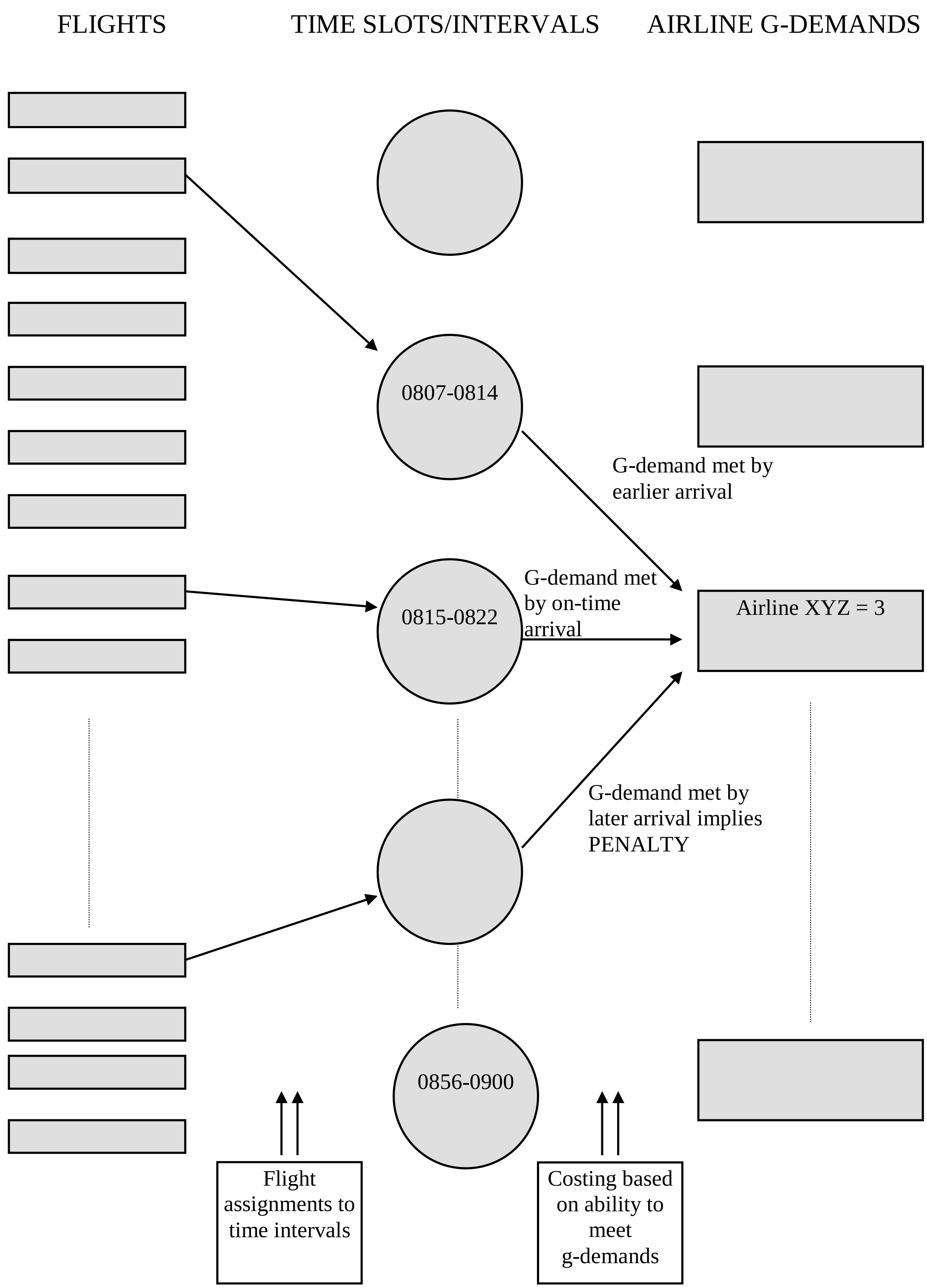
We now extend the OPTIFLOW model to include a fairness criterion and refer to the new model as the Goal-Demand, or G-Demand, model. When proposing the enhanced model, it should be clear what role OPTIFLOW plays in an environment where a highly dynamic substitution and cancellation process are being used and if fairness among airlines should be addressed. In this type of environment it may be impossible for the airline to make use of the time slots allocated by Grover Jack. Fairness considerations are necessary to provide an incentive for airlines to furnish accurate information. The G-Demand model seeks to minimize the deviation of each airline from certain fairness goals. For each airline *a* and time slot goal *t*, a time slot goal g_t^a is defined (Figure 4).

Figure 4: Concept of Fairness



Starting with the standard OAG schedule, we expand the schedule so that it is consistent with the projected reduced acceptance rates without sacrificing the original order. This is equivalent to the ration-by-schedule process explained earlier. The number of arrivals allocated to each airline in each time period represents a goal for that airline. This is represented by a "goal" demand (g-demand) denoted by g_t^a. These g-demands are used in RBS to allocate or ration the limited number of arrival resources among the airlines. The g-demand for period *t* is represented as a demand in period *t*. The g-demand for period *t* must be achieved by some incoming flight. No penalty occurs if the g-demand is met by an incoming flight that arrives in the same time period or by a flight that arrives in an earlier time period. If the g-demand is met by a later flight, then a penalty is incurred. There is a relationship between flight arrival times and these penalties: the later the flight, the larger the penalty. This is depicted in Figure 5. The model will always try first to satisfy the g-demand by a flight that arrives in the same period or in an earlier period. Second, it will attempt to satisfy the g-demand with the closest flight arrival time.

Figure 5: Fairness Costing



3.2.1 Integer Programming Formulation

New Data

$g_t^a =$ number of flights owed to airline *a* in interval *t*

$w_{it}^a =$ number of flights of airline *a* arriving in interval *i* to satisfy its g-demand in interval *t*

$k_{it}^a =$ cost of airline *a* arriving in interval *i* to satisfy its g-demand in interval *t*

Objective Function: Add new term to the OPTIFLOW objective function.

Minimize $\sum_{a=1}^{A} \left\{ \sum_{f \in O(a)} \sum_{i=i_f}^{I} c_{fi}^a v_{fi}^a + \sum_{i=1}^{T} \sum_{t=1}^{T} k_{it}^a w_{it}^a \right\}$

Constraints: Add two new constraints to the OPTIFLOW constraints.

(3) The number of flights of airline *a* arriving in time interval *t* equals the number assigned to a g-demand from time period *t*.

(4) The number of flights arriving in period *i* must equal the number of flights owed for period *t* for each airline.

(3) $$\sum_{f \in O(a)} v_{fi}^a = \sum_t w_{it}^a \qquad \forall a, \forall i$$

(4) $$\sum_i w_{it}^a = g_t^a \qquad \forall a, \forall t$$

The OPTIFLOW costs have changed from the original definition in the OPTIFLOW model. The new cost functions are defined by (1) and (2) below.

$$(1)\ c_{fi}^{a} = W_f \left(i - \hat{i}_f\right)^{1+\delta} \qquad if \quad \hat{i}_f \le i_f$$

$$c_{fi}^{a} = 0 \qquad otherwise$$

where the g-demand costs are defined as follows:

$$(2)\ k_{it}^{a} = P(i - t)^{1+\delta} \qquad if \ \ i > t$$

$$k_{it}^{a} = 0 \qquad otherwise$$

G-demand costs are incurred if a flight meets its demand late. If the demand is met by an earlier flight, as in (2), the cost is 0. P is a parameter used to trade off the overall penalty with other cost components and $\delta < 1$ is some positive parameter. Now the cost function k_{it}^{a} should dominate the function c_{fi}^{a}, for $i - i_f < 0$, which means $P >> W_f$. The cost function *k()* insures that flights meet their airlines' g-demands as early as possible and the function *c()* guarantees minimal delay costs. We seek to meet the airlines' g-demands first and then minimize delay costs which explains the need for *k()* to dominate *c()*. In our experiments, we set

$W_f = 1$ so that P can be as large as possible.

Thus, we can see the model in its entirety below:

$$\text{Minimize } \sum_{a=1}^{A} \left\{ \sum_{f \in O(a)} \sum_{i=i_f}^{I} c_{fi}^{a} v_{fi}^{a} + \sum_{i=1}^{T} \sum_{t=1}^{T} k_{it}^{a} w_{it}^{a} \right\}$$

Subject to:

$$(1)\ \sum_{a} \sum_{f} v_{fi}^{a} \le d_i \qquad \forall i$$

(2) $\sum_{i=i_F}^{I} v_{fi}^{a} = 1 \qquad \forall a, \forall f$

(3) $\sum_{f \in O(a)} v_{fi}^{a} = \sum_{t} w_{it}^{a} \qquad \forall a, \forall i$

(4) $\sum_{i} w_{it}^{a} = g_{t}^{a} \qquad \forall a, \forall t$

(5) $v_{fi}^{a} \in \{0,1\}$

3.2.2 Reformulation as a Multicommodity Flow Problem

In this section we formulate the model as a minimum-cost multicommodity network flow problem. We use the multicommodity problem structure to take advantage of special properties that make solving these problems easier. It is possible that this formulation could lead to a more efficient problem solution. Recall from network flow theory that unimodularity provides sufficient conditions for integer optimal solutions to the associated linear program. Since single-commodity flow problems have this property, highly efficient algorithms have been devised. Special algorithms for multicommodity flow problems exist. A multicommodity problem can be viewed as a single-commodity problem plus some side constraints. This structure provides an advantage in solving these types of problems.

It will be shown that the structure of this problem is similar to a transportation model. The major differences are that several commodities can share common arcs, and that flow of all commodities on an arc is constrained by the arc capacity.

For our formulation, each commodity is represented by a different airline. There are three sets of nodes. One set represents flights, the second represents arrival time periods, and the third represents the airlines' goals. Supply is located at the flight nodes and demand is located at the g-demand nodes. The time period nodes are similar to transshipment nodes or distribution centers in a network which models warehouse shipments. The general multicommodity flow problem from [9: p. 389] follows.

General Multicommodity Flow Problem

a_i^k = supply at node i

b_j^k = demand at node j

x_{ij}^k = flow of commodity k over arc (i, j)

c_{ij}^k = unit transportation cost of flow over arc (i, j) of commodity k

u_{ij} = capacity of arc (i, j)

E = set of arcs

Minimize $\displaystyle\sum_{k=1}^{r}\sum_{(i,j)\in A} c_{ij}^k x_{ij}^k$

subject to

$\sum_j x_{ij}^k - \sum_j x_{ji}^k = a_i^k$ if node i is a source for commodity k

$\sum_j x_{ij}^k - \sum_j x_{ji}^k = 0$ if node i is a transhipment node

$\sum_i x_{ij}^k - \sum_i x_{ji}^k = -b_j^k$ if node j is a sink for commodity k

$\sum_k x_{ij}^k \le u_{ij}$ for (i, j) $\in$ E

$x_{ij}^k \ge 0$ for all k and (i, j) $\in E$.

Let A, F, I, and T be defined as before, let E be the set of all feasible arcs and let $V=\{V_1, V_2, V_3\}$ be the set of all nodes with $\{V_2'', V_2'\} = V_2$. There is one node in V_1 for each flight and two nodes in V_2 representing each arrival time. The second set of nodes is broken into two parts to handle the capacity during the time intervals. There is also one node in V_3 representing each time slot that contains the g-demands for the airlines. The set of feasible arcs contains arcs from V_1 to V_2'', V_2'' to V_2', and V_2' to V_3.

Formulation

Data

Define $\hat{a}(f) =$ the airline that owns flight *f*. V_2' consists of a set of copies of each node in V_2''. For each $j \in V_2''$, denote by j' the copy of j in V_2'.

x_{ij}^{a} = flow of airline a across arc (i,j)

$D_{j'}$ = flow constraint

$c_{ij}^{\hat{a}(f)}$ = unit cost of flow on arc (i,j) for flight *f* of airline a

$c_{jj'}^{\hat{a}(f)}$ = unit cost of flow on arc (j, j′) for flight *f* of airline *a*

$k_{j'k}^{\hat{a}(f)}$ = unit cost of flow on arc (j′,k) for flight *f* of airline *a*

G_t^a = g-demand for airline a during interval *t*

Objective Function: Minimize costs.

Minimize $\sum_{\hat{a}(f)} \sum_{(i,j)\in E} c_{ij}^{\hat{a}(f)} x_{ij}^{\hat{a}(f)} + \sum_{\hat{a}(f)} \sum_{(j,j')\in E} c_{jj'}^{\hat{a}(f)} x_{jj'}^{\hat{a}(f)} + \sum_{\hat{a}(f)} \sum_{(j',k)\in E} k_{j'k}^{\hat{a}(f)} x_{j'k}^{\hat{a}(f)}$

Constraints:

(1) Supply at each flight node is 1.

(2) The demand for each airline must be met.

(3), (4) Capacity at nodes must not be violated.

(5) Flow across the arc cannot exceed the arc capacity.

(1) $\sum_{j=i_f}^{I} x_{fj}^{\hat{a}(f)} = 1 \qquad \forall \ f \in V_1$

(2) $\sum_{j'=1}^{I} x_{j't}^{a} = G_t^a \qquad \forall \ a \in A, \forall \ t \in V_3$

(3) $\sum_{f \in O(a): i_f \leq j} x_{fj}^{a} - x_{jj'}^{a} = 0 \qquad \forall \ a \in A, \ \forall j \in V_2''$

(4) $x_{jj'}^{a} - \sum_{t=1}^{I} x_{j't}^{a} = 0 \qquad \forall \ j' \in V_2' \text{ and } \forall \ a \in A$

(5) $\sum_{a=1}^{A} x_{jj'}^{a} \leq 1 \qquad \forall \ j \in V_2''$

(6) $x_{ij}^{a} \geq 0$

The cost function is the same as the cost function for the OPTIFLOW model for the first set of arcs (see Figure 7). The cost function for the second set is always 0 since these arcs simply carry the node capacities. The last set of arcs uses the fairness cost function from the G-demand model. The costs are defined as follows:

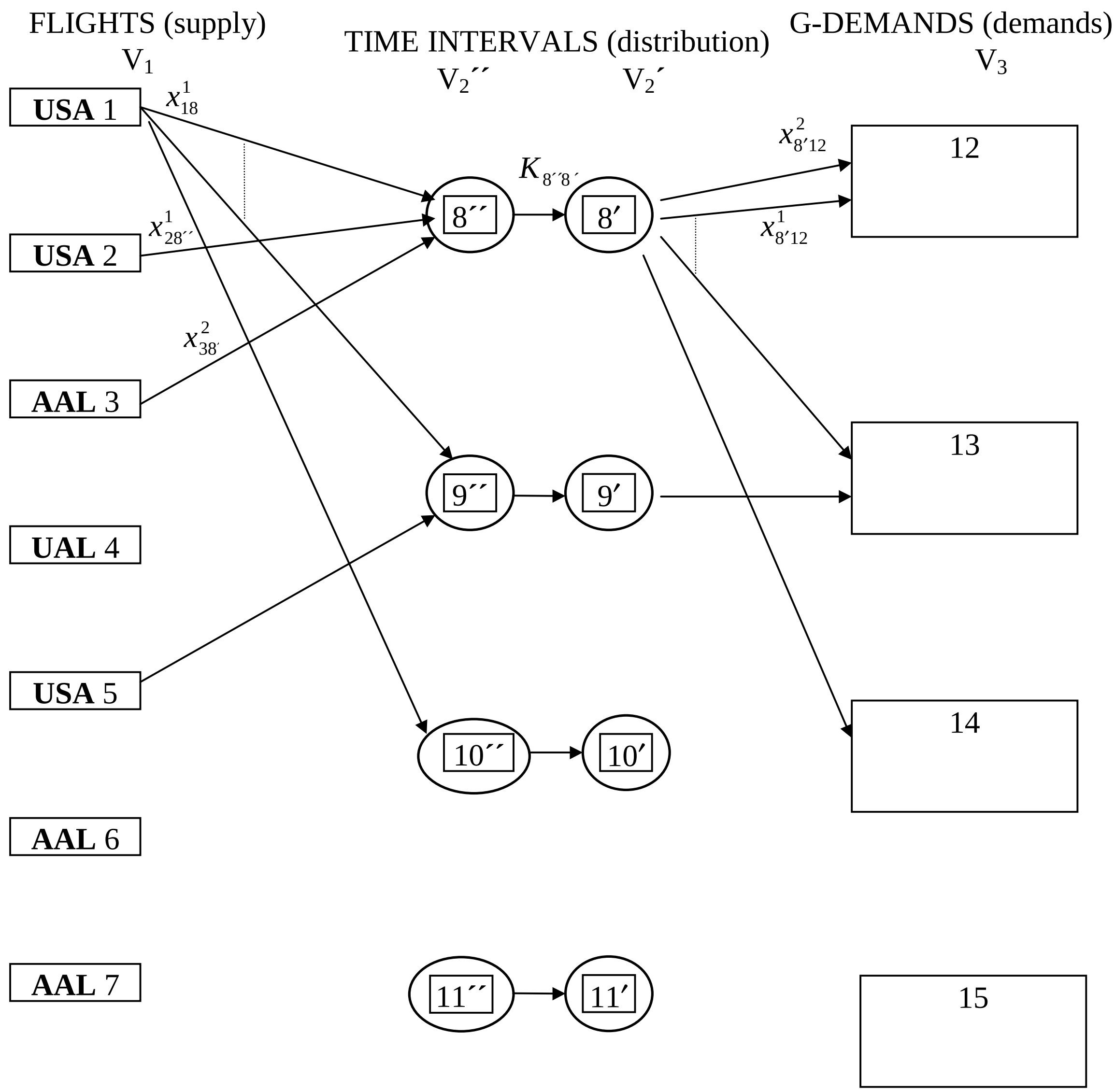
Case 1: $c_{ij}^{\hat{a}(f)} = W_f \left(i - \hat{i}_f\right)^{1+\delta}$ for all f and for all $i_f \leq i \leq I$.

Case 2: $c_{jj'}^{\hat{a}(f)} = 0$ for all f

Case 3: $k_{j'k}^{\hat{a}(f)} = P\left(j' - k\right)^{1+\delta}$

Again we have $W_f = 1$ and $\delta < 1$. The graphical representation of the directed multicommodity network can be seen in Figure 6.

Figure 6: Directed Network



Chapter 4

Experiments and Results

Experiments were conducted using the Goal-demand formulation given previously to determine if the new formulation, which takes fairness into consideration, is actually more equitable to the airlines than the current FSM. Data sets from actual GDPs were used here. The G-Demand model was translated into code, using various software, in order to test the model

4.1 Software Environment

Some background information about the software used for this experiment is necessary. AMPL is a relatively new entry into the field of algebraic modeling languages for mathematical programming. AMPL is notable for the similarity of its arithmetic expressions to customary algebraic notation, and for the generality of its set and subscripting expressions. AMPL also extends algebraic notation to express common mathematical programming structures such as network flow constraints. AMPL uses the solver CPLEX 4.0. The UNIX version was used for this experiment. Further explanations about AMPL can be found in [11].

CPLEX 4.0 is a math programming problem solver that solves problems quickly and accurately. CPLEX can handle large-scale, difficult problems in commercial settings where demand for performance and reliability are critical. CPLEX is available in a wide range of environments; the UNIX environment was used for the example problem that follows. More information is available at the CPLEX/ILOG web page [13].

The Flight Schedule Monitor (FSM) is the decision support tool developed for CDM. It contains three essential components: 1) graphical and timeline presentation of demand, 2) information extraction, and 3) ground delay utilities.

Through FSM, users will have the same picture of the problem that ATCSCC specialists see: the same information and the same capability to do the "what if" analysis and explore alternatives. NAS users can measure the expected effects of the program and begin developing their cancellation strategies or otherwise reschedule to mitigate the effects of irregular operations. FSM makes use several program elements. These include GDP Advisories, Ration by Schedule (RBS), Compression, and Simplified Substitutions. See [5] for more about FSM.

4.2 Example

First, the G-Demand model was translated into AMPL code (see code in appendix) and run with small data sets to test the validity of the model. A small data set was used consisting of two airlines, four flights per airline, and four time intervals. Each airline was given a g-demand for each interval and each interval had a specified arrival capacity. Following the definitions given previously in the g-demand formulation, each airline incurs a cost when it is assigned to a later time interval.

Because this problem is so small, a pictorial solution can be given easily. Using a specific example, we can observe how the model works. Using the data above, the two airlines are American Airlines (AAL) and United Airlines (UAL). The g-demands for AAL are 1, 2, 1, and 1 for intervals 1, 2, 3, and 4, respectively; UAL g-demands are 1, 1, 2, and 0 for intervals 1, 2, 3, and 4, respectively. Each time interval has a capacity of 2 arrivals. The costs are given in the tables below. The scheduled arrival times, which determine the costs, were arbitrarily chosen.

Table 7

c_{fi}^{a}	**AAL**			
	1	2	3	4
1	0	2	5	9
2	0	0	2	5
3	0	0	0	2
4	0	0	0	0

Table 8

c_{fi}^{a}	**UAL**			
	1	2	3	4
1	0	2	5	9
2	0	0	2	5
3	0	0	0	2
4	0	0	0	0

Table 9

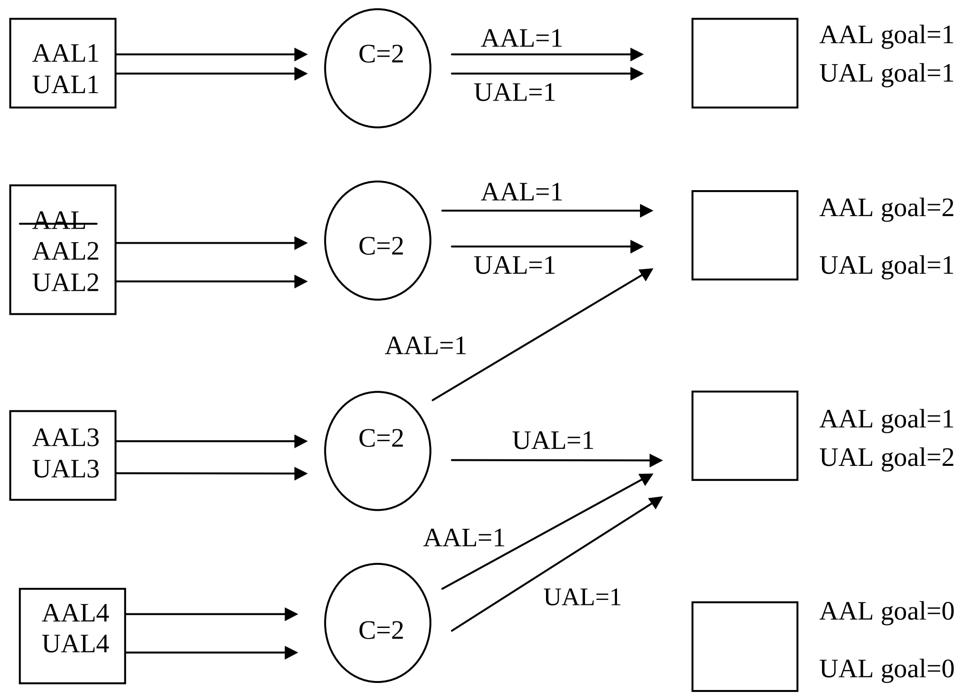
k_{it}^{a}	**AAL**			
	1	2	3	4
1	0	0	0	0
2	4	0	0	0
3	10	4	0	0
4	18	10	4	0

Table 10

k_{it}^{a}	**UAL**			
	1	2	3	4
1	0	0	0	0
2	4	0	0	0
3	10	4	0	0
4	18	10	4	0

Recall that c_{fi}^{a} = cost of flight *f* of airline *a* arriving during interval *i* and k_{it}^{a} = cost of airline *a* arriving during interval *t* to satisfy the g-demands for interval *i*. It was earlier stated that k_{it}^{a} should dominate c_{fi}^{a}. This is obvious from the tables above. Suppose American cancels a flight in interval 2. Below is the solution from AMPL.

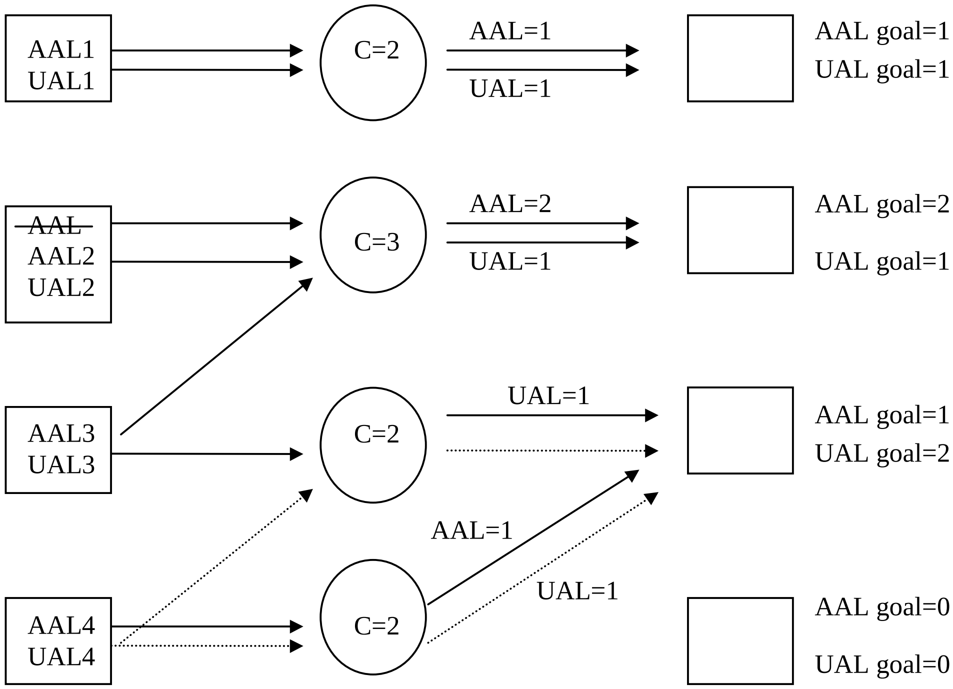
Figure 7: AMPL Solution 1



There are a few possible solutions to this example. A cost is incurred where flight AAL3 arrives in slot 3 to meet AAL's g-demand in interval 2. If we observe the cost tables, AAL incurs a cost of 4. Likewise, flight AAL4 incurs a cost of 4 and flight UAL4 incurs a cost of 4. The total cost is 12.

Suppose we change the capacity in interval 2 to 3 arrivals. The solution is as follows.

Figure 8, AMPL Solution 2



Since the capacity was increased in interval 2, flight AAL3 may arrive in interval 2, if feasible, yielding a cost of 0. This satisfies AAL's g-demand for interval 2 at no cost since it's an earlier flight. For an explanation of what AMPL sees, observe flight UAL4. Its choices were to arrive in an earlier time slot (cost =0), satisfying the g-demand in 3, at no cost, or land in its original time slot, satisfying the g-demand with a later arrival in 3, at a cost of 4. Subsequently, AMPL chose the former because it minimizes costs. The total cost for this problem is 0. Once the model was tested with small sets of data, larger data sets were used. The larger sets took into account the cost functions.

4.3 Tests on Real Data

For the larger experiments, instead of time intervals, time slots were used. A time slot is an actual clock time that denotes the arrival time of a flight. This was done to simplify the comparison process since FSM assigns flights this way. We set $P = 100$ and $\delta = .025$ in our experiments.

Four sets of data were used for the experiments, three from Newark International Airport (EWR) and one from Los Angeles International Airport (LAX). Two of the three Newark sets were for the same day, which indicates there were two GDPs run on that day. The GDPs were created by reducing the arrival capacities by half for each experiment. For each data set there was a flight list, RBS list, and compression list. These were merged using a C program that produced a data file containing all the variables and parameters necessary for input into the AMPL code.

The flight list contains all flight information, such as carrier, flight number, origin airport, destination airport, arrival and departure times, flight status, etc. The RBS lists and compression lists contain the assignments made by RBS and compression, respectively. Information such as EDCT, CTA, and slot are also included. Table 11 shows some of the statistics for each data set used.

Table 11: Airport Information

	EWR(1) 01/01/96	EWR(2) 01/01/96	EWR 01/02/96	LAX 01/01/97
Total number of flights	73	94	54	62
Total number of cancelled flights	12	21	6	10
Objective function value	15487.08	17735.70	12668.57	5899.39
Solution Time (in seconds)	1.85	4.05	0.67	0.95

The first large set of data used was historical data from Newark International Airport (EWR). The data was downloaded from FSM and included all cancelled flights. The Newark data was analyzed using the AMPL code for the G-Demand model (using time slots instead of intervals) with the results displayed in Table 12. The amount of delay reduction (in minutes) and the relative delay reduction were calculated for each airline for the compression algorithm, which FSM uses, and the G-Demand model.

The delay was calculated by subtracting the compression slot time from the initial slot time and the g-demand slot time from the initial slot time yielding the difference in minutes for each. There were a few instances where a flight remained in its original slot.

Baseline savings for each airline were also found. The baseline savings for a specific airline is the amount of delay reduction for that airline if each g-demand was filled by that airline's flights only. It provides a convenient basis for comparison of delay reduction on an airline-by-airline basis. In some sense it is the amount an airline should hope to achieve. In all cases each airline receives at least this amount. The fact that more total savings are available results from flight cancellations.

Table 12: Delay Reduction for Experiment 1 (EWR)

Airlines	Comp absolute	Comp relative	G-Demand absolute	G-Demand relative	Baseline Savings absolute	Baseline Savings relative
COA	402	46.53	391	42.25	281	57.00
UAL	200	23.15	199	23.03	142	28.80
TWA	17	1.97	17	1.97	0	0.0
AAL	123	14.24	126	14.58	70	14.20
ACA	2	0.23	0	0.0	0	0.0
USA	38	4.40	39	4.51	0	0.0
BSK	2	0.23	0	0.0	0	0.0
NWA	19	2.20	22	2.55	0	0.0
AWE	14	1.62	18	2.04	0	0.0
DAL	19	2.20	28	3.24	0	0.0
KMR	3	0.35	0	0.0	0	0.0
CAA	0	0.0	0	0.0	0	0.0
LOT	2	0.23	2	0.23	0	0.0
SJI	10	1.16	8	0.93	0	0.0
COM	13	1.50	14	1.62	0	0.0
TOTAL	**864**	**100.00**	**864**	**100.00**	**493**	**100.00**

The second experiment used data from a GDP run at EWR on the same day as Experiment 1. Two GDPs were run possibly due to inclement weather (snow, ice, etc.) since they occur in the winter. The results are shown in Table 13.

Table 13: Delay Reduction for Experiment 2 (EWR)

Airlines	Comp absolute	Comp relative	G-Demand absolute	G-Demand relative	Baseline Savings absolute	Baseline Savings relative
FDX	0	0.0	0	0.0	0	0.0
COA	521	50.63	469	45.58	420	75.0
NWA	79	7.68	80	7.77	0	0.0
ACA	4	0.39	0	0.0	0	0.0
UAL	171	16.62	169	16.42	68	12.14
AAL	81	7.87	104	10.11	72	12.86
USA	84	8.16	82	7.97	0	0.0
DAL	29	2.82	35	3.40	0	0.0
DLH	0	0.0	0	0.0	0	0.0
TWA	2	0.19	3	0.29	0	0.0
BSK	6	0.58	0	0.0	0	0.0
AWE	6	0.58	21	2.04	0	0.0
BAW	6	0.58	28	2.72	0	0.0
KMR	16	1.55	14	1.36	0	0.0
LOT	24	2.33	24	2.33	0	0.0
TOTAL	**1029**	**100.00**	**1029**	**100.00**	**560**	**100.00**

Results from Experiment 3 are given in Table 14.

Table 14: Delay Reduction for Experiment 3 (EWR)

Airlines	Comp Absolute	Comp relative	G-Demand absolute	G-Demand Relative	Baseline Savings absolute	Baseline Savings relative
COA	231	64.71	187	52.38	167	85.20
ACA	40	11.20	40	11.20	0	0.0
SJI	3	0.84	0	0.0	0	0.0
COM	2	0.56	5	1.40	0	0.0
N4I	2	0.56	2	0.56	0	0.0
UAL	60	16.81	60	16.81	29	14.80
MXA	2	0.56	0	0.0	0	0.0
NWA	5	1.40	0	0.0	0	0.0
VIR	3	0.84	22	6.16	0	0.0
TWA	3	0.84	7	1.96	0	0.0
PAL	2	0.56	5	1.40	0	0.0
AJM	1	0.28	0	0.0	0	0.0
USA	1	0.28	0	0.0	0	0.0
AAL	1	0.28	15	4.20	0	0.0
CAA	1	0.28	14	3.92	0	0.0
TOTAL	**357**	**100.00**	**357**	**100.00**	**196**	**100.00**

The large amounts of reduction for COA in the second and third experiments may be attributed to the fact that EWR is a hub for COA. This simply means COA uses EWR as a base of operation and a central point of transfer for passengers.

The last experiment used data from Los Angeles International Airport (LAX). The results are shown in Table 15.

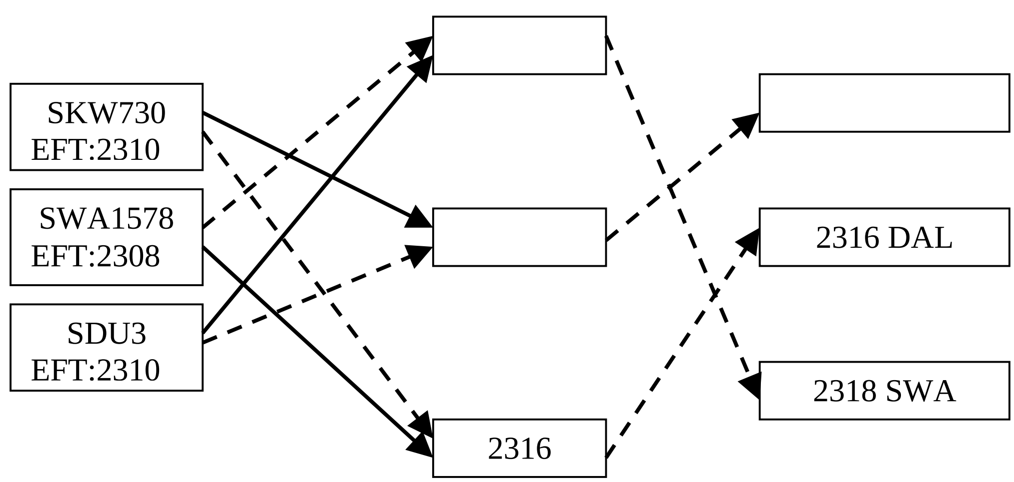
Table 15: Delay Reduction for Experiment 4 (LAX)

Airlines	Comp absolute	Comp relative	G-Demand absolute	G-Demand relative	Baseline Savings absolute	Baseline Savings relative
UAL	153	42.62	131	36.49	127	53.59
AAL	72	20.06	70	19.50	70	29.54
SWA	25	6.96	29	8.08	18	7.59
TWA	38	10.58	36	10.03	0	0.0
ASA	6	1.67	6	1.67	0	0.0
SER	0	0.0	0	0.0	0	0.0
DAL	8	2.23	6	1.67	0	0.0
FDX	4	1.11	6	1.67	0	0.0
RKT	2	0.56	0	0.0	0	0.0
ROA	9	2.51	11	3.06	0	0.0
AMX	2	0.56	16	4.46	0	0.0
ANZ	2	0.56	2	0.56	0	0.0
AWE	0	0.0	0	0.0	0	0.0
USA	24	6.69	22	6.13	22	9.28
COA	2	0.56	10	2.79	0	0.0
NWA	6	1.67	9	2.51	0	0.0
FFT	6	1.67	5	1.39	0	0.0
TOTAL	**359**	**100.00**	**359**	**100.00**	**237**	**100.00**

From the previous tables, it is clear that our objective, to mimic the compression model, was met. Generally, the solutions are similar, although, compression tends to allocate more to the dominant airline where as our model tends to spread savings among all airlines.

Each solution was validated to check that the model was performing properly. Then the solutions from the G-Demand model and compression were compared. As an example, a small sample was extracted from Experiment 4 for a closer look. Figure 9 shows the flights, the flight's earliest flight time (EFT), time slots, and g-demands for Experiment 4. The earliest flight time is the slot time assigned to that flight before the GDP was run. A flight cannot arrive earlier than this time. The table illustrates the flight assignments from the OPTIFLOW model (first set of dashed arcs) and the g-demands (second set of dashed arcs). The solid arcs represent the compression solution. In order to understand how the G-Demand model works, we can observe the differences in the figure below.

Figure 9: Experiment 4, Sample Solution



Compression reordered the flights based on internal allocation of each airline's flights. This is known as intra-airline mapping. The flights were then pushed up according to this new order. Looking at Figure 9 above, it seems rather arbitrary to base assignments on this. G-Demand pushes the flights up according to the EFT. Flight SWA1578 had the earliest EFT so it was assigned to the earliest slot.

The second set of arcs simply shows which flights met the airlines' goals. Delta Airlines had its g-demand at 2316 met by flight SKW730, which arrived in the same slot. This was the first available g-demand slot belonging to DAL that had not been met. In the G-Demand model, flight SWA1578 arrived in slot 2310 and satisfied the goal in slot 2318. This means the goal was met early. The United Airlines (UAL) g-demand at 2312 was met by flight SDU3, which arrived in this same slot.

For all of the experiments, the G-Demand model attempted to satisfy the g-demand by a flight that arrives in the same slot first. If there was no available flight, it then looked for an earlier flight. If this does not work, it simply looks for a flight with the closest arrival time. The flights were assigned such that they all are approximately the same distance from their original arrival times. This fits our objective of minimizing the deviation from the original flight time.

The differences explained above also explain the aggregate differences from the earlier tables showing delay reduction and baseline savings. Taking a closer look, we see the dominating airline at each airport gets special treatment in the compression model. Since UAL has more flights using LAX in Experiment 4, compression gave the UAL flight preference and allowed it to arrive first.

Chapter 5

Conclusions

The Goal-demand model presented in this thesis provides a solution to the air traffic management problem of minimizing the amount of ground-holding delay incurred by an airline during a Ground Delay Program. It uses criteria, previously defined by participants in CDM, to generate “fair” solutions. The benefits obtained in the model by each airline are not negatively impacted by the disclosure of up-to-date schedule changes. This will, hopefully, encourage improved data exchange among those involved.

The G-Demand model was formulated first to mimic the compression model. Our goal was to formulate a formal model that could replace compression. A formal model could then provide a basis for further research and be used in later models. Accordingly, we attempt to find ways to improve upon the compression model. The G-Demand model uses integer programming to solve the minimization problem. The practicality of the model is also important to note because it can be solved very quickly using commercial integer programming solvers. The fact that the G-Demand model is practical makes it a good candidate for future use in CDM decision support tools or other tools. An alternate multicommodity formulation was given that may be more efficient than the G-Demand model.

Experiments were conducted on the integer-programming model and output was compared from FSM and the IP model. Comparisons were then made between the G-Demand and Compression output and the baseline savings. The solutions are largely similar but there are some differences. We uncovered areas where the G-Demand solution had more desirable characteristics than the compression solution. Thus, we feel the G-Demand model provides a very promising approach for use in GDP planning.

Appendix A

The Multi-Airport Ground-Holding Problem in Air Traffic Control

In Vranas, Bertsimas, and Odoni's paper [3], a ground-holding problem is formulated for multiple airports. Their formulation takes into account deterministic airport capacities and that ground delays (or airborne delays) are decided once at the beginning of the day. Consider a set of airports, K ={1,…, K}, time intervals, I ={1,…, I}, and flights, F={1,…,F}(A single aircraft may perform several of these flights). Here F is a closed network of airports, where departures from and arrivals to the external world are not considered important. For each flight $f \in$ F, the following is assumed:

$k_f^d \in$ K, the airport from which *f* is scheduled to depart

$k_f^a \in$ K, the airport to which *f* is scheduled to arrive

$d_f \in$ I, the scheduled departure time of *f*

$r_f \in$ I, the scheduled arrival time of *f*

$c_f^g(\cdot)$ = the ground delay cost function of *f* (whose argument is the ground delay of *f* in time intervals)

$c_f^a(\cdot)$ = the airborne delay cost function of *f* (whose argument is the airborne delay of *f* in time intervals)

$D_k(i)$ = the departure capacity for each $(k,i) \in \mathrm{K} \times \mathrm{I}$

$R_k(i)$ =the arrival capacity for each $(k,i) \in \mathrm{K} \times \mathrm{I}$

G_f = maximum number of time periods that flight *f* may be held on the ground

A_f = maximum number of time periods that flight *f* may be held in the air

Now consider the set $F' \subset F$ of continued flights. A flight is continued if the aircraft which is scheduled to perform it is also scheduled to perform at least one more flight later in the day. For each flight f', assume that the next flight f scheduled to be performed by the same aircraft, and the "slack" or "absorption" time $s_{f'}$ such that, if f' arrives at its destination at most $s_{f'}$ time periods late, the departure of the next flight f will not be affected. Then $s_{f'}$ is equal to the difference between the time interval between the scheduled departure time of f and the scheduled arrival time of f'; and the minimum turnaround time of the aircraft performing both flights.

The decision variables are:

g_f = the number of time periods that flight f is held on the ground before being allowed to take-off, $f \in F$

a_f = the number of time periods that flight f is further held in the air before being allowed to land $f \in F$

(Recall that the above delays are determined once at the beginning of the day for all flights.)

$u_{fi} = 1$ if flight f is assigned to take-off at period i, and 0 otherwise

$v_{fi} = 1$ if flight f is assigned to land at period i, and 0 otherwise

Following is the integer programming (IP) formulation for the multi-airport ground delay problem.

Minimize $\sum_{f=1}^{F}\left(c_f^g g_f + c_f^a a_f\right)$

subject to

1) $\sum_{f:k_f^d=k} u_{fi} \leq D_k(i),(k,i) \in \mathrm{K} \times \mathrm{I}$

2) $\sum_{f:k_f^a=k} v_{fi} \leq R_k(i),(k,i) \in \mathrm{K} \times \mathrm{I}$

3) $\sum_{i \in I_f^d} u_{fi} = 1 \quad f \in \mathcal{F}$ where $I_f^d = \{i \in I : d_f \leq i \leq \min(d_f + G_f, I)\}$

4) $\sum_{i \in I_f^d} v_{fi} = 1 \quad f \in \mathcal{F}$ where $I_f^a = \{i \in I : r_f \leq i \leq \min(r_f + G_f + A_f, I)\}$

5) $g_{f'} + a_{f'} - s_{f'} \leq g_f \qquad f' \in \mathcal{F}'$

6) $a_f \geq 0 \qquad f \in \mathcal{F}$

$u_{fi}, v_{fi} \in \{0,1\}$

Constraints 5) are the coupling constraints: They transfer any excessive delay of flight *f′* to its next flight *f*. The delay variables may be expressed in terms of the assignment variables:

$$g_f = \sum_{i \in I_f^d} i u_{fi} - d_f, \quad f \in \mathcal{F}$$

$$a_f = \sum_{i \in I_f^a} i v_{fi} - r_f - g_f, \quad f \in \mathcal{F}.$$

Thus these variables may be eliminated totally from the formulation leaving u_{fi} and v_{fi} as the only decision variables.

To derive the OPTIFLOW model, mentioned in an earlier section of this thesis, let K=1 for one airport; $a_f = 0$ since we are only concerned with ground delays; $s_{f'} = 0$ because there are no continuing flights in our model; and $D_k(i) = 0$ and $d_f = 0$ since we only want to observe arrival capacities and arrival times. By making any necessary substitutions, we get the OPTIFLOW model.

Appendix B

AMPL Code

```
set AIRLINE;       # different airlines
set TIME1 ordered; # actual time intervals
set FLT{AIRLINE};  # different flights for each airline
set ALL_FLIGHTS := union {a in AIRLINE} FLT[a];
                    # set of all possible flights
set COMPRESSION_ASSIGNMENTS within (ALL_FLIGHTS cross TIME1);
                    # compression assignments


param P;            # constant value used in cost equation

param W;            # constant value used in cost equation

param goal {AIRLINE, TIME1} >= 0, default 0;
                                  # goal for airline a in interval t

param cap {TIME1} >= 0;           # capacity for interval t

param flttime{a in AIRLINE, f in FLT[a]};
                                  # scheduled arrival time interval for
                                  # each flight

param flttime_earliest{a in AIRLINE, FLT[a]};
                                  # the earliest interval each flight can
                                  # arrive in


param cost1 {a in AIRLINE, f in FLT[a], i in TIME1:
                         flttime_earliest[a,f] <= i <= flttime[a,f]}
        :=if (i - flttime_earliest[a,f])=0 then 0 else
 (W*((i-flttime_earliest[a,f])^1.025));
        # cost of airline a arriving during interval i

param cost2 {a in AIRLINE, i in TIME1, t in TIME1}
        :=if (i - t)<0 then 0 else
 (P*((i - t)^1.025));
        # cost of airline a arriving during interval i
        # to satisfy interval t

var air1 {a in AIRLINE, f in FLT[a], i in TIME1 :
ord(flttime_earliest[a,f],TIME1) <= ord(i,TIME1) <=
ord(flttime[a,f],TIME1)} binary;
```

```
                    # 1 if flight f of airline a is assigned to interval i

var air2 {a in AIRLINE, i in TIME1, t in TIME1} >= 0;

                    # The total number of flights of airline a arriving in
                    # i to satisfy the goal in t


minimize TOTAL_COST:
    sum {a in AIRLINE} ((sum {i in TIME1, f in FLT[a]:
                              flttime_earliest[a,f] <= i <= flttime[a,f]}
          cost1[a, f, i] * air1[a, f, i])+(sum {i in TIME1,t in TIME1}
          cost2[a, i, t] * air2[a, i, t]));

subject to FLT_ASSGN {a in AIRLINE, f in FLT[a]}:
    sum {i in TIME1: flttime_earliest[a,f] <= i <= flttime[a,f]}
          air1[a,f, i] = 1;

# A flight can only be assigned to one time interval

subject to INT_CAP {i in TIME1}:
     sum {a in AIRLINE,f in FLT[a] : flttime[a,f] >= i >=
          flttime_earliest[a,f]} air1[a, f, i]<=cap[i];

# The total number of flights cannot exceed the arrival capacity for
# interval i

subject to FLOW {a in AIRLINE, i in TIME1}:
   sum {f in FLT[a]: flttime_earliest[a,f] <= i <= flttime[a,f]}
          air1[a, f, i] - sum {t in TIME1} air2[a, i, t] = 0;

# The total number of flights arriving to satisfy i must equal the
# total arriving during t

subject to TOT_GOAL {a in AIRLINE, t in TIME1}:
   sum {i in TIME1} air2[a, i, t] = goal[a, t];

#The total number of flights arriving in t must equal the total owed for i
```

References

[1] Machol, Robert E. June 1995, Managing Airspace. *OR/MS Today*.

[2] Glockner, Gregory D. 1996, "Effects of Air Traffic Congestion Delays Under Several Flow Management Policies," Ph.D. dissertation, Georgia Institute of Technology.

[3] Vranas, Peter B., Bertsimas, Dimitris J., and Odoni, Amedeo R. 1994, "The Multi-Airport Ground-Holding Problem in Air Traffic Control," *Operations Research***,** **42**, 249-261.

[4] Terrab, Mostafa and Odoni, Amedeo R. 1993, "Strategic Flow Management for Air Traffic Control", *Operations Research*, **41**, 138-152.

[5] *Collaborative Decision Making in Aviation Transportation* Web Pages, October 1997, http://www.metsci.com/faa/faa.html.

[6] Andreatta, Giovanni, Odoni, Amedeo R., and Richetta, Octavio. 1993, "Models for the Ground Holding Problem," in *Large Scale Computation and Information Processing in Air Traffic Control*, L. Bianco and A.R. Odoni, eds., New York: Springer-Verlag.

[7] Distinct Memorandum to the OPTIFLOW Team. 1994, "Extension of OPTIFLOW Model to Include Fairness among Airlines."

[8] Ball, Michael O. *Extension of OPTIFLOW Model*. Fall 1996, Seminar, University of Maryland at College Park.

[9] Phillips, Don T. and Garcia-Diaz, Alberto. 1981, *Fundamentals of Network Analysis***.** Englewood Cliffs, NJ: Prentice-Hall, Inc.

[10] Golden, Bruce. 1995, Network Flows. *Lecture notes from Network Theory*.

[11] Fourer, Robert, Gay, David M., and Kernighan, Brian W. 1993, *AMPL A Modeling Language for Mathematical Programming.* Danvers, MA: Boyd & Fraser Publishing Company.

[12] Hoffman, Robert and Ball, Michael O. 1997, "A Comparison of Formulations for the Single-Airport Ground Holding Problem with Banking Constraints," Submitted for publication.

[13] *CPLEX/ILOG* Web Pages, October 1997, http://www.cplex.com.

[14] Hoffman, Robert, Rifkin R., Ball, Michael O., Odoni A. 1998, "The Stochastic Ground-Holding Problem," Manuscript in preparation.

[15] Hoffman, Robert L. 1997, "Integer Programming Models for Ground-Holding in Air Traffic Flow Management," PhD dissertation, University of Maryland at College Park.