USING TACTICAL FLIGHT LEVEL RESOURCE ALLOCATION TO ALLEVIATE CONGESTED EN-ROUTE AIRSPACE

by

Antonio Manuel Abad

B.S., Aerospace Engineering
Georgia Institute of Technology, 2004

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN AERONAUTICS AND ASTRONAUTICS
AT THE
MASSACHUSETTS INSTUTUTE OF TECHNOLOGY

JUNE 2004

Signature of Author: ______________________________
Department of Aeronautics and Astronautics
May 21, 2004

Certified by: ______________________________
John-Paul Barrington Clarke, Sc.D.
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by: ______________________________
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

USING TACTICAL FLIGHT LEVEL RESOURCE ALLOCATION TO ALLEVIATE CONGESTED EN-ROUTE AIRSPACE

by

Antonio Manuel Abad

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

ABSTRACT

A motivation exists to formulate and implement new tools and methodologies to address the problem of congestion in the National Airspace System (NAS). This thesis presents a novel methodology for allocating aircraft among En Route flight levels as a means to mitigate air traffic congestion and stakeholder operating costs. The core of the methodology is a decision-aiding tool comprised of a Mixed-Integer Linear Program (MILP) that is solved using a an A* Search-based Branch & Bound framework. Two metrics, measuring cumulative delay reduction and fuel burn savings, are used to benchmark the performance of the methodology. A combination of these two metrics is also explored as a means to minimize overall airline operating costs.

A subsection of the Northeast Corridor is modeled and forms part of the analytic structure used to quantify the potential benefits of the proposed methodology. Simulations are generated from these models in order to gain an understanding of the benefits as they relate to varying NAS conditions. The following scenarios were modeled: 1) A baseline single jetway corridor, 2) Reduced Vertical Separation Minimum (RVSM), 3) Miles in Trail (MIT) restrictions on corridor traffic, and 4) the merging of Terminal Area air traffic with En route air traffic. Thus, this research also provides a preliminary, quantitative measure of the delay reduction, fuel burn savings and operating cost savings possible under each scenario, within a NAS corridor setting.

Results indicate that 8.5 minutes of delay reduction per flight can be achieved when minimizing air traffic delay. Similarly, 16.47 kg/min of fuel burn savings per flight can be achieved when minimizing air traffic fuel burn. Instituting RVSM procedures result in an additional 45% of delay reduction. Imposing MIT restrictions result in a 41% loss of delay reduction savings. These results were obtained for corridor simulations of 30 minutes in duration. Finally, the methodology is shown to be effective for use as a decision-aiding tool to merge air traffic streams.

Thesis Supervisor: John-Paul Barrington Clarke
Title: Assistant Professor of Aeronautics and Astronautics

Acknowledgements

The person who deserves the greatest measure of my gratitude is my thesis advisor, John-Paul Barrington Clarke. It is with the utmost sincerity that I commend him for his generosity, support, patience, and ultimately, understanding. I am very fortunate to have benefited from being under his tutelage, and my completion of this thesis and my tenure as a Master's Degree student is due, in no small part, to him. He is the one person who demonstrated unwavering confidence in my abilities during the most tumultuous and difficult portions of the past two years.

I would also like to thank my primary benefactor, the Department of Defense. This work and my graduate school education would not have been possible without their generous support.

I am also indebted to my family and friends for all the love, support, and humor that I have received from them. I have always considered myself blessed for the friendships that I have established in my life thus far. My experience at MIT was no exception in this regard.

Finally, I can honestly write that I have never been in the midst of so many brilliant, accomplished, and motivated people. MIT is a unique and comically idiosyncratic culture, and I consider myself honored to have been a part of it.

Table of Contents

List of Figures

List of Tables

1. Introduction

Delays in the National Airspace System (NAS) are a source of unnecessary cost to three primary stakeholders: airlines, passengers, and air transportation-dependent businesses. Furthermore, a significant increase in the magnitude of delays within the next decade under the current air traffic control infrastructure is indicated in numerous industry forecasts. Thus, if the current air traffic control infrastructure is not modified in any significant way, there exists a need to develop and implement new tools to address the problem of congestion in the NAS.

A novel methodology for allocating aircraft among En Route flight levels is proposed in this thesis as a means to mitigate En Route air traffic congestion and thereby reduce stakeholder operating costs. The algorithmic basis of the resulting decision-aiding tool is a Mixed-Integer Linear Program (MILP) that utilizes an Artificial Intelligence-based search heuristic.

The potential benefits of such a decision-aiding tool was determined though the simulation of air traffic operations in a section of the NAS that is known for its significant levels of congestion and delay.

1.1 Problem Statement

Air transportation in the U.S. economy represents 3% of the Gross Domestic Product, or a total economic contribution of $273 billion in 1998 [Kostiuk, et al, 1998]. Furthermore, these economic figures increase in relevance when considering the significant growth forecasted for the air traffic industry. Estimates include:

- An annual growth of 2.5% in U.S. aircraft operations. [Boeing, 2001]
- An annual growth of 3.9% in U.S. domestic enplanements (543 million in 1997 to 821 million in 2009). [Boeing. 2001]
- An increase of approximately 2,500 jets to accommodate the 43% increase in the number of enplanements between 1999 (5,236 jets) and 2008 (7,737). [ATA, 1999]
- An increase of approximately 250% in daily delay hours, from 2,710 hours in 1998 to 9,605 hours in 2008. [ATA, 1999]

Air traffic congestion represents a problem that is estimated to cost the aviation industry, passengers, and shippers approximately $5 billion per year [Boeing, 2001]. This cost can be further segregated into a $3 billion impact upon direct airline operating costs and a $2 billion impact upon the value of collective passenger time. Given the enormous growth potential in air transportation, the cost of delays in the system is also expected to grow. Thus, the forecasted growth in air traffic demand, aircraft operations, and emplanements will further exacerbate NAS congestion and the resulting economic cost.

With this expected level of network congestion and the need for a safe, reliable, and robust NAS, the Federal Aviation Administration (FAA) has initiated a 10-year initiative referred to as the Operational Evolution Plan (OEP), with the primary goal of increasing NAS capacity. Four of the commitments outlined in the OEP include [CAASD, 2003]:

- Match airspace designs to demands.
- Collaborate to manage congestion.
- Reduced Vertical Separation Minimums (RVSM)
- Accommodate user preferred routing.

It is important to note that these commitments are specifically tailored to increasing En Route NAS capacity because airborne delays are the second largest contributor to congestion in the network, as illustrated in Figure 1-1 [ATA, 1999].

Previous solution methodologies have relied on framing the problem in a way that has ignored the short-term dynamics of the system. In many of the models and decision aiding tools that have been developed to address the problem of En Route congestion,

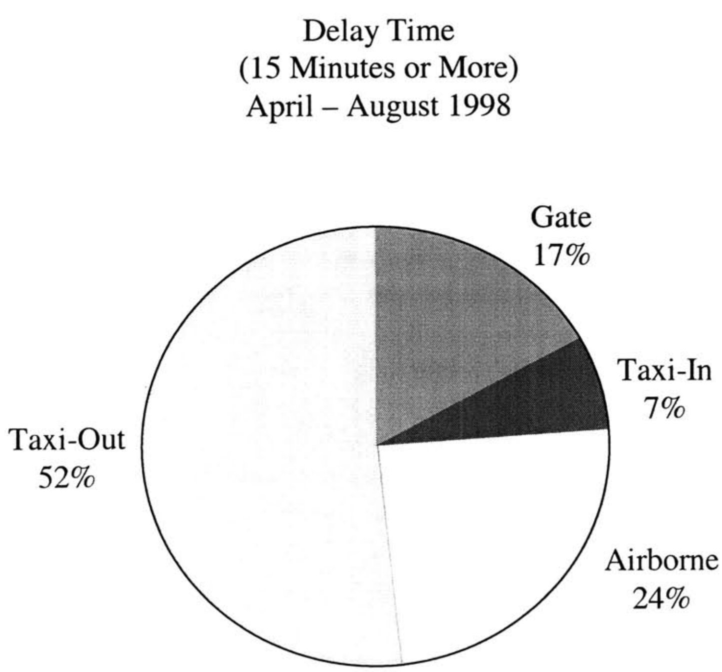


Figure 1-1: Distribution of Delay Time

traffic has been modeled at the strategic level. That is, the traffic in the network is abstracted and modeled as flows. For example, in 2000, the FAA introduced an initiative to reduce flight delays caused by convective weather through the use of two strategic elements: Traffic Flow Management (TFM) and Collaborative Routing (CR).

Evans best summarizes the properties and shortcomings of this approach as it relates to reducing delays caused by convective weather:

> "This strategic approach has been quite successful in improving operations in many cases. However, in congested airspace, the inability to forecast convective weather impacts requires a complimentary tactical decision support capability [Evans, 2001]"

Essentially, Evans argues that there is a need to address the issue of air traffic management from a tactical, real-time perspective. In other words, there is a need to command aircraft individually in order to address congestion in those areas requiring the immediate attention of controllers.

The intent of this thesis is to determine the extent to which the tactical rerouting of aircraft among the available En Route flight levels can maximize airspace efficiency, minimize system congestion, and minimize stakeholder operating costs. To that end, a model of a representative, congested subset of the NAS was developed and used to determine the cost savings afforded by a decision-aiding model which tactically allocated aircraft to flight levels. The analysis also provides a qualitative basis for determining whether decision-aiding tools can and should be developed.

1.2 A Representative Example: The Northeast Corridor

Several areas within the NAS are severely congested. In fact, the FAA has identified seven so-called "choke points" as illustrated in Figure 1-2. During instances of severe congestion, the effects of these choke points propagate throughout the NAS and ultimately stress the whole air traffic system because of its interconnected and dependent nature. The most notorious of these areas is the "Northeast Corridor," which encompasses the East Coast of the United States from the Mid Atlantic to Northeast regions. Due in no small part to its geography and urban density, the Northeast Corridor has often been cited as the busiest air traffic area in the world [ATC, 2000]. Thus, because this airspace has a natural proclivity to congestion, the Northeast Corridor is an ideal and relevant candidate to model in this research. Two expected benefits of modeling the Northeast Corridor include:

1. A generalized methodology capable of mitigating delays in the other parts of the NAS.
2. A system-wide benefit resulting from the reduced congestion in this particular parcel of airspace.

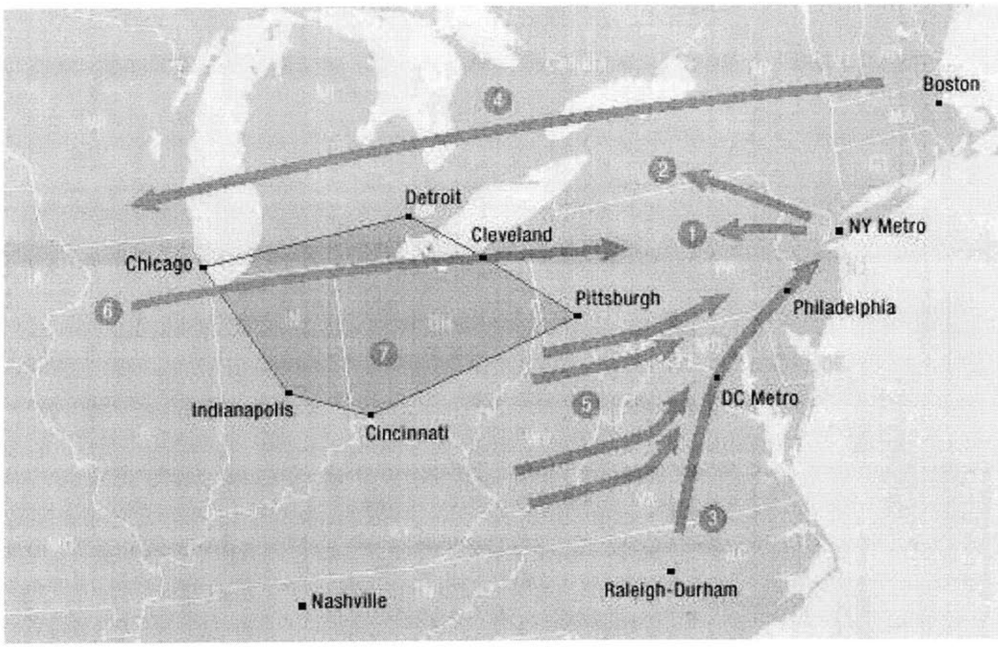


Figure 1-2: The Seven National Choke Points

1.2.1 Characteristics of the Northeast Corridor

As its name implies, the Northeast Corridor is a parcel of airspace where because of various constraints, traffic is restricted to a long but narrow corridor. There are 3 primary factors responsible for the corridor:

- The linear alignment of the major urban centers along the East Coast.
- The high population density within these urban centers.
- The presence of restricted military airspace over the Atlantic Ocean.

The geography of the largest urban centers along the East Coast is shown in Figure 1-3. As shown in the figure, the major urban centers lie along a line or corridor, stretching from Washington D.C. to Boston, MA. Another feature to note is the relative proximity

Figure 1-3: Major Population Centers from Mid-Atlantic to Northeast Regions

of the major urban centers to one another. The impact of this latter observation is that a significant amount of the air traffic in the given airspace will be in the so-called "transition phase" of flight, when the aircraft are ascending from and/or descending to the major airports along the East Coast.

The high population density of the major urban centers also drives the traffic density. There exists an intuitive and strong correlation between airport activity and population size in the area around an airport. Simply stated, a large population living in proximity to a major airport (or set of airports) typically begets a large magnitude of air traffic traveling to and from that airport (or set of airports). To illustrate this relationship, Table 1-1 lists the population sizes of the top 10 metropolitan areas from the year 2000 Census, ranked from most populous to least populous [USCB, 2003]. The most important point to note from these rankings is that 4 of the top 10 population centers reside within the geographical area of the corridor, suggesting a proportionally higher quantity of traffic within the corresponding area of the NAS.

Given the constraints imposed by the geography and population density, one might expect that a viable alternative would be to route air traffic over the Atlantic Ocean. However, as illustrated by Figure 1-4, that option is precluded in all three relevant Air Routing Traffic Control Centers (ARTCCs) because of restricted airspace situated off the East Coast. For example, a third of the airspace controlled by the Washington ARTCC

Table 1-1: 10 Highest Populated U.S. Metropolitan Areas (2000)

Metropolitan Area	Population
New York - Northern New Jersey – Long	**21,199,865**
Los Angeles – Riverside - Orange County	16,373,645
Chicago-Gary - Kenosha	9,157,540
Washington - Baltimore	**7,608,070**
San Francisco – Oakland - San Jose	7,039,362
Philadelphia - Wilmington - Atlantic City	**6,188,463**
Boston – Worcester - Lawrence	**5,819,100**
Detroit - Ann Arbor - Flint	5,456,428
Dallas - Fort Worth	5,221,801
Houston – Galveston - Brazoria	4,669,571

resides above the Atlantic Ocean. However, the Instrument Flight Rule (IFR) High Altitude Map presented in Figures 1-5 & 1-6 illustrates that the area immediately off the East Coast is restricted airspace that can only be used by military aircraft [FAA, 2002]. Thus, no commercial aircraft can be routed over the area under normal circumstances. The map illustrates the resulting feature that all of the commercial jet routes are confined to an area bounded by the East Coast. Further perusal of the map and the jet routes illustrates the existence of a corridor as many of the jet routes serving the different urban centers are clustered closely together in parallel lines.

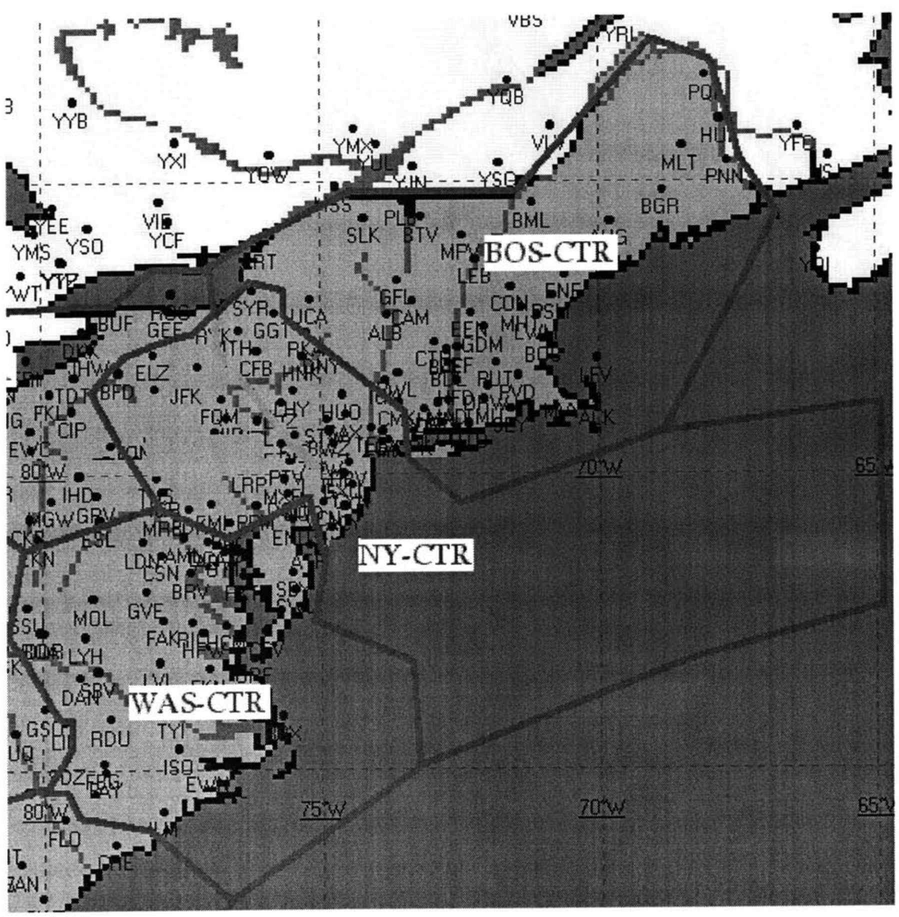


Figure 1-4: Northeast Corridor ARTCCs

The final piece of evidence supporting the existence of the congested corridor is illustrated in Figure 1-7, which is a depiction of the airports that exceeded the 20,000 hours threshold in annual delays in 2000 [FAA, 2001]. As illustrated in the figure, all of the major airports over the geographic region from Washington D.C. to Boston, MA exceeded the given threshold. No other region in the NAS of comparable geographic size demonstrates this magnitude of delay-plagued airports.

This concentration of delays illustrates the inter-dependency among all of the relevant major airports. Air traffic operating between any two cities within the corridor must compete for limited air space resources with the rest of the corridor traffic. Thus, delays that arise in the interior points of this corridor tend to readily propagate throughout the rest of the corridor [ATC, 2000]. For example, if there are delays due to heavy traffic heading into one of the New York airports, the trailing flights headed to other destinations, such as Providence or Boston, have no choice but to suffer these delays as well.

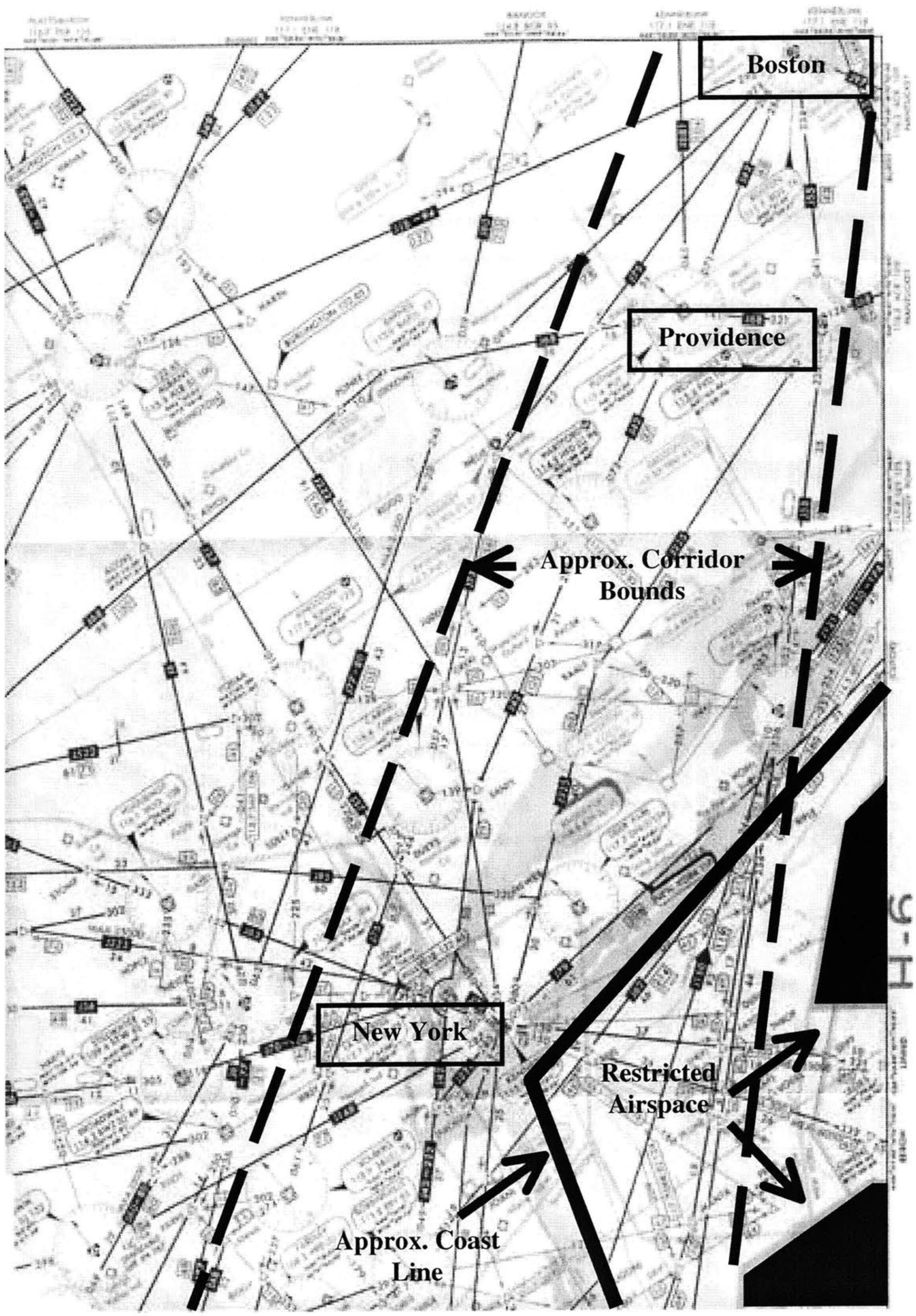


Figure 1-5: IFR High Altitude Map (New York – Boston)

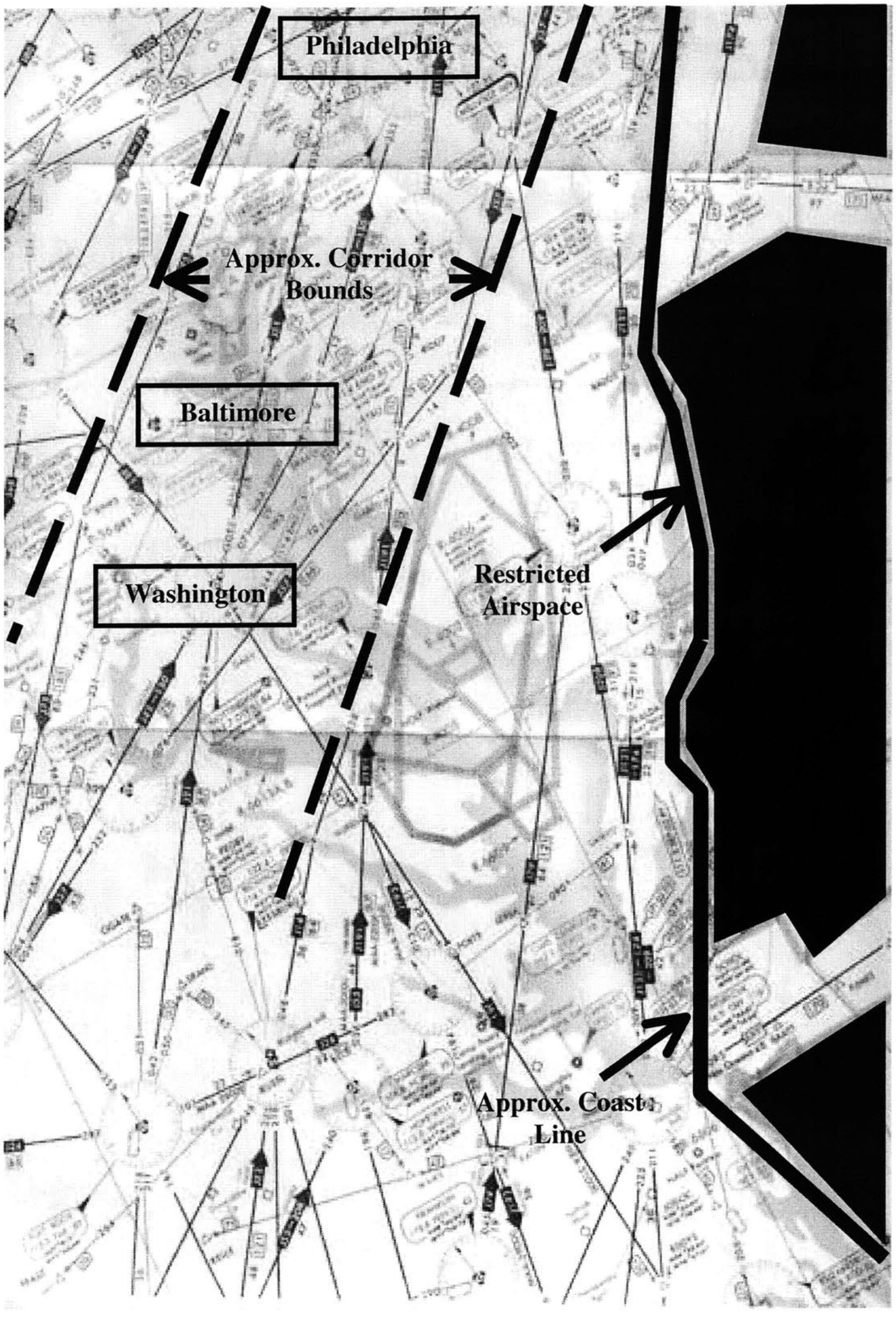


Figure 1-6: IFR High Altitude Map (Washington - Philadelphia)

Figure 1-7: Airports Exceeding 20,000 Hours of Annual Aircraft Delay in 2000

1.2.2 Representing Air Space Corridors as Two-Dimensional Models

The IFR High Altitude Map clearly shows the presence of several, parallel jet routes that can be used by the traffic in the Northeast Corridor. Given this corridor geometry, there is a natural motivation to reduce the dimension of the problem to mitigate system complexity. That is, rather than basing the model on latitude, longitude and altitude, the model may be based on range and altitude only. This reduction in the problem dimension significantly reduces the computational complexity of the mathematical models developed to solve the problem.

Despite the multiple, parallel jetways depicted in the IFR High Altitude Map, a 2-dimensional model is justified by the following two observations:

- Jet Routes are used to handle the inflow/outflow of traffic to/from specific locations.
- There may be times when weather conditions shut down portions of the airspace such that only one jet route is used to funnel air traffic through the corridor.

The first observation is a fundamental aspect of the inherent structure of the NAS. That is, because of the high volume of traffic, many jetways are defined as one-way routes or are used for traffic into or out of a specific terminal area. The second phenomenon occurs when inclement weather prohibits the use of one or more jetways, and even in instances of lower traffic loads, the remaining available jetways are congested.

1.3 Previous Work

The FAA has focused on solving the problem of NAS congestion via Collaborative Decision Making (CDM) [Thedford, et al., 1999]. This approach is based on the belief that a system-wide optimum will be achieved if users have common situational awareness via access to real-time data on current and predicted estimates of weather, traffic flow, and airport capacities. That is, in such an interconnected and open environment, users will be able to tell beforehand whether their selfish policies will result in a congested network. One such example is the scenario where all airlines prefer the same rerouting option around an area of inclement weather, thereby inadvertently congesting the associated routes.

1.3.1 Collaborative Decision Making

As the name suggests, CDM is a paradigm where NAS users, essentially the airlines and control centers, all have access to real-time NAS information and can collaborate to make mutually acceptable decisions of greater overall benefit. The CDM process is comprised of the following three components, all tailored to alleviating delays in the NAS:

- Ground Delay Program Enhancements (GDP-E)
- Initial Collaborative Routing (ICR)
- National Airspace System Status Information (NASSI)

The motivation behind GDP-E is simple and predicated upon the fact that holding an airplane on the ground, rather than in the air, is a more cost-effective option when certain destination airports are, or are predicted to be, operating at capacity. This tool became operational in September 1998 and relies on an accurate prediction of capacity at the major airports. While it is an extremely useful tool in curtailing future congestion in many instances, it does not alleviate an already-congested airspace.

In the case of ICR, traffic management specialists at various control centers share real-time traffic flow information with the Airline Operation Centers. This is the primary tool used in establishing rerouting decisions, especially in times of inclement weather and/or congestion. Of the three CDM tools, this is the most similar one in terms of the scope of this thesis in that the goal is to increase NAS efficiency via traffic rerouting. This capability is currently available in the following centers: Boston, New York, Washington, Indianapolis and Cleveland high altitude centers, as well as New York Terminal Radar Approach Control (TRACON), and the Air Traffic System Control Center (ATSCC) in Washington, D.C.

Finally, NASSI is a tool used to disseminate a wider range of information relating to the operational status of the NAS to all users. Example information may include the runway visual range at the major airports or the status of Special Use Airspace. While not a particularly relevant tool in actively alleviating a currently congested airspace, it functions as somewhat of a preventative measure.

1.3.2 Autonomy in Air Traffic Control

Although no research has been carried out on altitude assignment as a means to reduce congestion, the methodology does leverage on the research in the area of conflict-directed, sector-based En Route traffic rerouting [Pallottino, et al., 2002]. Furthermore, the decision model ultimately developed as part of the work in this thesis is inspired, in part, by the trajectory-based modeling and optimization of multi-vehicle systems [Vanderbie, 2000].

1.4 Thesis Scope and Goal

This document is divided into the following remaining chapters:

- The analytic structure is developed in Chapter 2.

- Background on MILP techniques is presented in Chapter 3 for readers who are not familiar with them.
- Similarly, background on search heuristics is presented in Chapter 4.
- The MILP decision model is presented in Chapter 5.
- The results of the analysis are presented and discussed in Chapter 6.
- Conclusions and suggestions for future work are offered in Chapter 7.

2 Research Approach

As mentioned in the previous section, one goal of this research is to produce a framework that can evaluate everything from the achievable problem size to how the output from the decision model changes when various input parameters change. Thus, a sound analysis structure is vital to this research. The development of this structure, which is depicted in Figure 2-1, is described in this section.

The development begins with the modeling of the air traffic on several chosen jetways in the Northeast Corridor using Enhanced Traffic Management System (ETMS) Data. The result of this first part of the analysis is a statistical summary of aircraft types and speeds

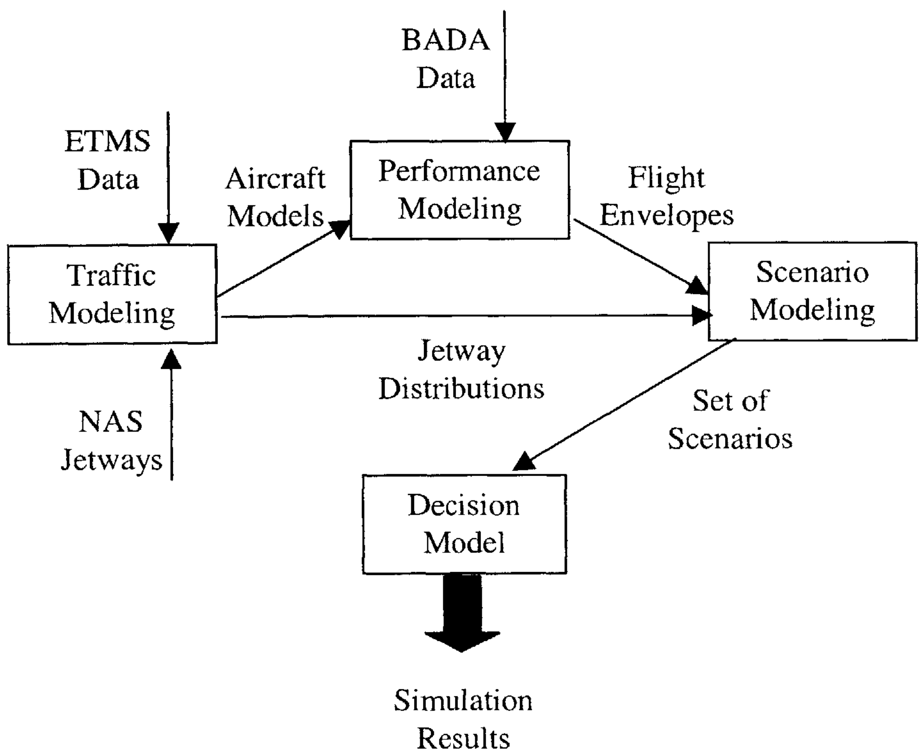


Figure 2-1: Analysis Architecture

as a function of flight levels. The resulting aircraft types, along with Base of Aircraft Data (BADA) are then used to model the corridor traffic. BADA is a performance and operating procedures database of 186 different types of aircraft. The result from this second part of the analysis is the definition of a flight envelope and a set of altitude-dependent minimum fuel burn speeds for each aircraft used in the simulation. This output is used in conjunction with the statistical traffic distributions to model a variety of scenarios. Each scenario describes a possible and interesting corridor traffic distribution. The result of this third part of the analysis is a set of scenarios that will serve as input for the final component of the analysis framework: the decision model. The decision model is a MILP, detailed in Chapter 5, which is used to determine the optimal redistribution of corridor traffic for each scenario. The output of the decision model provides a measure of the benefits of redistributing the corridor traffic using metrics that will be defined later in this chapter.

2.1 Traffic Modeling

In order to formulate representative simulations of corridor traffic, traffic level distributions must first be derived from a real source of air traffic data. The source for the distributions used in this research is ETMS data. ETMS is a system used by ARTCC Traffic Management Specialists to evaluate the current and projected state of the NAS. Specifically, the system provides real-time information including, but not limited to: flight position, flight speed, altitude, destination, and waypoint sequence. The trajectory information is updated every two minutes and is stored for both real-time decision-aiding tools and future analyses. One such tool is Monitor Alert, which provides controllers and strategic flow managers a prediction of the areas in the NAS where air traffic demand will exceed capacity.

The data that is described below was used to formulate the traffic models used in the ensuing simulations. Specifically, these models include Probability Density Functions of aircraft true air speed as a function of aircraft type for corridor traffic. The analysis used to derive these models is presented in detail in this section.

2.1.1 Data Source

The data, obtained from the Volpe National Transportation Systems Center, contains the flight listing and trajectory information for all commercial aircraft in the NAS. Note that General Aviation and Military air traffic were already filtered out from this data. The data includes traffic from the following dates: October 15, 2000; October 16, 2000; October 21, 2000; October 28, 2000; October 30, 2000.

2.1.2 Filtering ETMS Data

A filter was used to extract only those aircraft flying in the Northeast Corridor. This filter was designed using a combination of geographical, direction and temporal filters. A second filter was used to extract only those aircraft on a prescribed jetway. The output from these filters was used to generate the statistical distributions of corridor traffic on the specific jetways ultimately chosen as the basis of the simulations.

2.1.3 Exploiting Fundamental Air Space Structure

As shown in Table 2-1, northbound and southbound traffic are segregated into discrete altitudes [FAA, 2003]. Thus, rather than considering a continuous range of altitudes, established En Route flight levels were used as the basis to describe traffic in the ensuing modeling and simulations.

2.1.4 Choosing Representative Jetways

The two airways chosen for the analysis in this thesis are illustrated in Figure 2-2. The first jetway, J42, is used primarily as a northbound feeder into LaGuardia Airport. Northbound Terminal Area Traffic from the Washington D.C. area airports also transition into this jetway. The second jetway, J191, is used as a northbound/southbound feeder into/from Newark International Airport. Note that the two jetways are essentially parallel to each other and do not intersect.

2.1.5 Jetway Statistics

The percentage of each aircraft type using Jetway J42 and Jetway J191 are shown in Tables 2-2 and 2-3, respectively. Note that the aircraft type "other" consists of a variety of aircraft types, none of which individually account for more than 2% of the jetway

Table 2-1: Cruising Altitudes in Class A Airspace

Magnetic Ground	18,000' MSL to FL 290	FL 290 and Above
0 – 179	Odd 2000' Intervals	4000' Intervals Begin at FL 290
180 - 359	Even 2000' Intervals	4000' Intervals Begin at FL 310

traffic. Thus, only the following aircraft types were modeled in the simulation:

- Boeing 737 (All variants)
- Boeing 757 (All variants)
- McDonald Douglas MD80/MD88/MD90/Boeing 717
- Airbus A318/A319/A320
- Embraer EMB-145

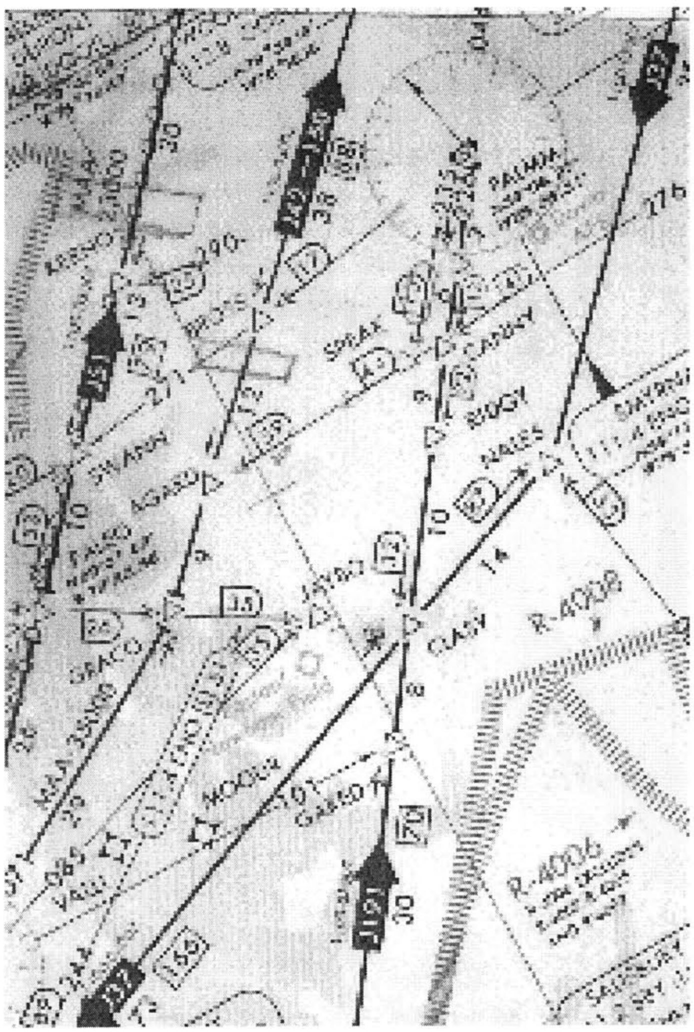


Figure 2-2: Jetway Samples

The set of percentages listed in Tables 2-2 and 2-3 were used to generate aircraft during subsequent simulations. For example, when generating an aircraft type for a particular simulation of J42, there was a 42.7 % chance that it will be a Boeing 737, a 10.5% chance that is was a Boeing 757, and so on.

Table 2-2: J42 Aircraft Model Distributions

Aircraft Type	10/15/00	10/15/00	10/15/00	10/15/00	10/15/00	Totals	%
Boeing 737	101	91	141	132	108	573	42.7
Boeing 757	21	24	33	33	30	141	10.5
Boeing MD80/88/90/717	48	33	76	79	43	279	20.8
Airbus A318/A319/A320	13	9	24	25	11	82	6.1
Embraer EMB-145	22	17	17	17	22	95	7.08
Other	25	27	44	47	28	171	12.8
Totals	230	201	335	333	242	1341	

Table 2-3: J191 Aircraft Model Distributions

Aircraft Type	10/15/00	10/15/00	10/15/00	10/15/00	10/15/00	Totals	%
Boeing 737	99	118	130	128	109	584	43.55
Boeing 757	26	32	34	30	31	153	11.41
Boeing MD80/88/90/717	67	51	76	76	60	330	24.61
Airbus A318/A319/A320	19	15	26	25	17	102	7.61
Embraer EMB-145	20	22	21	24	23	110	8.20
Other	33	35	38	41	39	186	13.87
Totals	264	273	325	324	279	1465	

The reader is also directed to Appendix I, which consists of all of the flight speed histograms for each aircraft type and flight level. Superimposed on each histogram is the normal distribution. The parameters of the normal distribution, namely the Mean and Standard Deviation of the True Air Speed (TAS), were used to generate an initial TAS for each aircraft. These parameters are listed in Tables 2-4 and 2-5.

No statistics were determined for Miles in Trail (MIT) spacing. This is due to the low traffic loading observed on the specified jetways. In fact, the maximum number of daily flights on all the jetways in the Northeast Corridor was 2201 flights, observed on October 28, 2000. This is a very low number of aircraft given the number of operational hours in

Table 2-4: J42 Flight Level-Specific TAS Distributions (kts)

	Boeing 737		Boeing 757		Airbus A320		Boeing MD80		Embraer EMB-145	
Flight Level	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
190	398	43	0	0	414	27	371	39	392	68
210	417	30	0	0	0	0	0	0	394	43
230	394	69	0	0	0	0	0	0	400	43
250	429	33	0	0	0	0	0	0	421	28
270	442	27	0	0	0	0	454	48	412	59
290	456	30	480	27	466	17	0	0	460	42
330	455	26	447	89	473	22	471	34	464	52
370	460	41	476	29	468	15	460	21	496	70
410	0	0	475	10	0	0	0	0	525	14

Table 2-5: J191 Flight-Specific TAS Distributions (kts)										
	Boeing 737		Boeing 757		Airbus A320		Boeing MD80		Embraer EMB-145	
Flight Level	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
190	396	47	0	0	400	27	360	43	387	67
210	421	46	0	0	0	0	0	0	389	51
230	404	63	0	0	0	0	0	0	397	38
250	438	31	0	0	463	22	0	0	414	29
270	457	36	0	0	0	0	0	0	410	64
290	456	33	482	23	464	20	0	0	463	45
330	454	27	436	90	470	21	472	34	469	49
370	464	40	477	27	468	14	461	21	508	77
410	0	0	474	24	0	0	0	0	490	25

a day, the number of total jetways in the corridor, the multitude of Eastbound/Westbound jetways, and the number of available flight levels.

2.2 Aircraft Modeling

In the second step of the modeling process, a detailed performance analysis was performed for the most common aircraft types that were identified. The result of this analysis was the definition of a flight envelope for each aircraft model, consisting of the maximum and minimum true air speeds specified for every flight level. In addition, the minimum fuel burn speed was identified for each aircraft on each flight level.

2.2.1 Performance Modeling

The Base of Aircraft Data (BADA), revision 3.3, was used to model aircraft performance. The European Organization for the Safety of Air Navigation has compiled this database, which incorporates the performance and operating procedures for 186 different aircraft types. Therefore, the most relevant data, the minimum and maximum altitude-dependent true airspeed was the first data referenced. These airspeeds reflect the typical operating limits and not the absolute bounds of the designed flight performance.

The minimum airspeed was specified as a Calibrated Air Speed (CAS), which is altitude-independent. This value yields the minimum TAS for any specific altitude via the following equation:

$$v_{min_{TAS}} = \left[\frac{2}{\mu}\frac{P}{\rho} \left\{ \left(1 + \frac{(P_0)_{ISA}}{P} \left[\left(1 + \frac{\mu}{2}\frac{(\rho_0)_{ISA}}{(P_0)_{ISA}} V_{min_{CAS}}^2 \right)^{1/\mu} - 1 \right] \right)^{\mu} - 1 \right\} \right]^{1/2}$$

$$\mu = \frac{(\gamma - 1)}{\gamma}$$

where:

$v_{min_{TAS}}$ - Minimum True Air Speed (Altitude-Dependent)

$v_{min_{CAS}}$ - Minimum Calibrated Air Speed (Altitude-Independent)

P - Air Pressure

$(P_0)_{ISA}$ - Standard Atmosphere Sea-Level Pressure (101,325 Pa)

P - Air Density

$(\rho_0)_{ISA}$ - Standard Atmosphere Sea-Level Density $\left(1.225 \ {}^{\text{kg}}\!/_{\text{m}^3}\right)$

γ - Isentropic Expansion Coefficient for Air (1.4)

The one notable assumption in the conversion from the minimum CAS value to the minimum TAS value is the specification of standard atmospheric conditions, which is standard practice in many performance analyses.

The maximum airspeed was specified as a Mach number, which is also altitude-independent. This value yields the maximum TAS for any specific altitude via the following equation:

$$v_{max_{TAS}} = M_{max} a = M \sqrt{\gamma RT}$$

where :

$v_{max_{TAS}}$ - Maximum True Air Speed (Altitude - Dependent)

M_{max} - Maximum Mach Number (Altitude - Independent)

a - Speed of sound

R - Real Gas Constant for Air $\left(287.04\ \mathrm{m}^2/\mathrm{Ks}^2\right)$

T - Temperature

Unlike the minimum and maximum TAS, the minimum and maximum longitudinal acceleration limits were specified uniformly for all aircraft and flight levels as:

$$a_{max} = 2.0\, fps$$
$$a_{min} = -2.0\, fps$$

These values were defined in a generic sense in order to reflect the limits necessary to ensure passenger safety and comfort [Nuic, 23].

With the altitude-dependent minimum and maximum speeds defined for each aircraft, the altitude-dependent minimum fuel burn speeds were then determined. The first step in this part of the analysis was to identify the cruise conditions, as illustrated in the Free Body Diagram in Figure 2-3. The Free Body Diagram establishes the following dynamics equations:

$$
\begin{aligned}
&L = W = mg \\
&T = D \\
&where: \\
&L \text{ - Lift} \\
&W \text{ - Weight} \\
&m \text{ - Mass} \\
&T \text{ - Thrust}
\end{aligned}
$$

One notable observation regarding these equations was the specification of static cruise conditions. In order for these relationships to relate to the present analysis, they were expressed as a function of the aircraft TAS. This was accomplished using the following fundamental aerodynamic relationships:

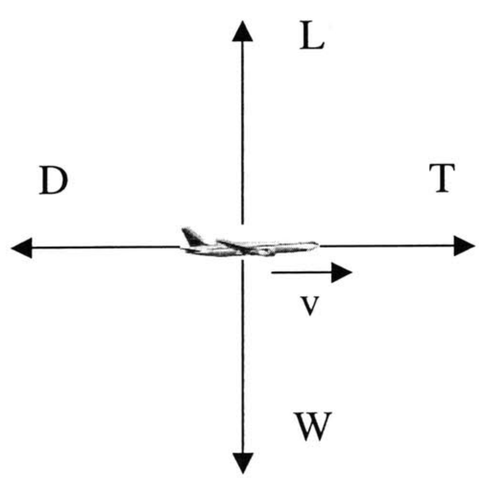


Figure 2-3: Free Body Diagram

$$
\begin{aligned}
&L = \frac{1}{2}\rho S C_L V_{TAS}^2 \\
&D = \frac{1}{2}\rho S C_D V_{TAS}^2 \\
&C_D = C_{D_0} + C_{D_2} C_L \\
&where: \\
&S \text{ - Wing Area} \\
&C_L \text{ - Coefficient of Lift} \\
&C_D \text{ - Coefficient of Drag} \\
&C_{D_0} \text{ - Zero - Lift Coefficent of Drag} \\
&C_{D_2} \text{ - } 2^{\text{nd}} \text{ Order Coefficient of Drag}
\end{aligned}
$$

Substituting these equations for Lift and Drag into the dynamics previously derived from the Free Body Diagram results in the following expression for Thrust as a function of TAS:

$$
T = \frac{1}{2}\rho S C_L V_{TAS}^2 + \frac{2m^2 g^2 C_{D_2}}{\rho S V_{TAS}^2}
$$

Finally, given this expression for Thrust, the final step in calculating the fuel burn as a function of TAS was to define the Thrust Specific Fuel Consumption (TSFC) equation, as follows:

$$
\begin{aligned}
&TSFC = \eta = C_{f_1}\left(1 + \frac{V_{TAS}}{C_{f_2}}\right) \\
&f = \eta T C_{f_{CR}} = C_{f_{CR}} C_{f_1}\left(1 + \frac{V_{TAS}}{C_{f_2}}\right) T \\
&where: \\
&TSFC\ (\eta) \text{ - Thrust Specific Fuel Consumption} \\
&f \text{ - Fuel Burn Rate} \\
&C_{f_1} \text{ - Fuel Correction Factor 1}
\end{aligned}
$$

C_{f_2} - Fuel Correction Factor 2

$C_{f_{CR}}$ - Cruise Fuel Correction Factor

Thus, when the TAS-specific expression is substituted for the Thrust T in this last expression, the result is an analytic equation for the fuel burn rate as a function of TAS. However, the derivative of this function does not yield an expression that can be solved analytically. Therefore, the TAS corresponding to the minimum fuel burn was calculated numerically. These results, as well as the results from the rest of the performance analysis, are summarized in the following section.

2.2.2 Performance Analysis Results

The minimum TAS, maximum TAS and minimum fuel burn TAS values are listed in Appendix II for each aircraft and flight level in the model. In addition, the maximum operating altitude h_{MO}, minimum CAS CAS_{min}, and maximum Mach Number M_{max} of each aircraft are also listed. The altitude-dependent fuel burn curves for each aircraft are illustrated in Appendix III.

2.3 Scenario Modeling

The final step in the modeling process was to define the scenarios that would be simulated. Three specific cases, along with a baseline case, were defined:

- The Single Jetway En Route Scenario (a.k.a. The Baseline Case)
- The Single Jetway Merging Traffic Scenario

- The Reduced Vertical Separation Minimum (RVSM) Scenario
- The Miles In Trail Restrictions (MIT) Scenario

2.3.1. The Single Jetway En Route Scenario (a.k.a. The Baseline Case)

In this scenario, all traffic was high level En Route traffic on a single jetway. Specifically, the traffic on Jetway J191, the main feeder into Newark Airport, was simulated. The main utility of this scenario was that it serves as the default analysis or baseline case. That is, this type of scenario was used to explore the solution properties of the decision model. Such properties include the maximum problem size and the solution quality. Both of these terms will be defined during the development of the decision model.

2.3.2 The Single Jetway Merging Traffic Scenario

While the network structure of the Northeast Corridor does not contain a significant amount of merging jetways, there are several Terminal Area boundaries where Terminal traffic from the lower altitudes are merged with En Route traffic from the higher altitudes. For this case, the traffic on Jetway J42, which serves as an outflow for the Washington D.C. Terminal Area traffic, was simulated. This scenario was developed to determine whether the routing logic could merge the intersecting traffic streams.

2.3.3 The Reduced Vertical Separation Minimum Scenario

This scenario was developed to determine the benefit of introducing a new flight level in a congested corridor. This idea results from the notion that current surveillance technology is sophisticated enough to allow controllers to maintain a comparable level of safety with a reduced vertical separation standard. Traffic on a congested Jetway J42 was simulated.

2.3.4 The Miles In Trail Restrictions Scenario

This final scenario was developed to determine whether the routing logic was capable of readjusting traffic that is suddenly subject to MIT restrictions that require a greater longitudinal separation among aircraft. For example, if corridor traffic density is on the order of the nominal 5 miles of separation, then it is important that the methodology determine a solution after controllers impose an MIT restriction of 10 miles. Such a situation would occur when inclement weather develops within the corridor, thus requiring a greater amount of separation between aircraft.

2.4 Analysis Objectives and Metric Definitions

The final component needed to completely define the research approach were the metrics used to assess the results from the ensuing simulations. This is equivalent to defining the objective used to allocate flight level resources to corridor traffic in the simulations. The first and perhaps most obvious objective candidate was to minimize traffic delay. However, there is a fundamental relationship between aircraft flight time and aircraft performance that argues in favor of minimizing aircraft fuel burn as an alternative objective. Thus, a second fuel burn objective was developed. A third, mixed objective, composed of a linear combination of aircraft delay and aircraft performance was also defined.

2.4.1 Cumulative Delay Reduction

Minimizing the cumulative corridor traffic delay, or maximizing the network throughput, requires the routing of aircraft through the network while ensuring that each flight remains within its flight envelope. Namely, this requires that its commanded speed lies within the minimum and maximum true air speed on the flight level for which it is ultimately rerouted to.

However, using flight delay as the optimization parameter potentially results in traffic redistributions that are prohibitively expensive for airlines. Using this objective to reroute the traffic results in configurations where individual aircraft are commanded to speeds as close as possible to their altitude-dependent maximum true air speed. In addition, because the maximum true airspeed does not correspond to the minimum fuel burn speed, more fuel is consumed.

The apparent stakeholder preference for optimizing with respect to fuel burn begs the following question: why consider the use of cumulative delay reduction as the objective? However, the use of cumulative delay reduction is still very useful in that it can serve as the basis of analyzing the properties of the decision model. That is, sensitivity analyses can be carried out to gauge what effects varying certain parameters have on the resulting flight level allocation. Perhaps the most applicable use to the stakeholder lies in the fact that it can serve as a means to measure any additional and available capacity. Finally, it can be used to quantitatively analyze the effect of instituting RVSM tunneling in the corridor.

Thus, the cumulative delay reduction objective was the metric used to answer the following fundamental research questions:

- How much available capacity exists in a given corridor scenario?
- What is the practical problem size limitation for the decision model?
- How does the redistribution change when other factors change?
- What are the quantitative benefits of tunneling in a given corridor scenario?

2.4.2 Cumulative Fuel Burn Reduction

Consider that fuel burn often represents the metric used to gauge aircraft performance because it has a direct impact upon airline operating costs. Further consider that the coupling between fuel burn and flight time is complex due to the occasional trade-off in optimizing for one at the expense of the other. This coupling is obvious when considering the causality of flight time upon fuel burn: a longer flight time necessitates a greater fuel burn. Indeed, delays incur both fuel burn and time penalties on an aircraft. However, solely considering this relationship may result in the incorrect presumption that minimizing flight time implicitly minimizes fuel burn as well. The assumption is invalid because minimizing flight time for a fixed distance requires the aircraft to maintain higher speeds in steady, level flight during the cruise segment. This reduction in flight time, along with the associated higher flight speed may require a larger fuel burn rate due to the fact that summing the fuel burn rate over the total flight time does not guarantee a net fuel burn saving for every expedited flight profile.

The assumption must be adjusted so as to consider the impact of aircraft performance. In particular, a given aircraft model operates optimally at a given altitude-speed combination as illustrated in the generalized thrust required versus flight speed profile in Figure 2-6. Notice the specification of two distinct optimization points on the curve. The minimum of the curve represents the optimal operating point for the purpose of minimizing fuel burn per unit time (i.e. endurance), while the other designated location represents the optimal operating point for the purpose of minimizing fuel burn per unit distance (i.e. range). In order to explain the relevance of this latter point on the curve, consider the definition of the Thrust Specific Fuel Consumption (TSFC) parameter, which is a measure of the fuel burn rate:

$$\mathrm{TSFC} = \frac{dm_{\mathrm{fuel}}/dt}{\mathrm{T}}, where:$$
$$dm_{\mathrm{fuel}}/dt - Fuel\ Flow\ Rate$$
$$T - Thrust\ \mathrm{Pr}oduced\ by\ Engines$$

giving the following for the fuel burn rate:

$$dm_{\mathrm{fuel}}/dt = TSFC \times T$$

As shown, the dependent variable is only the thrust, *T*. Thus, the minimization of the fuel burn per unit time is achieved simply via the minimization of the required thrust, as illustrated in Figure 2-4.

On the other hand, consider how this defining equation changes when considering the fuel burn over an incremental range *dR*:

$$\frac{TSFC}{dR} = \frac{dm_{\text{fuel}}/dt}{T \times dR}$$
$$\Rightarrow dm_{\text{fuel}}/dR = \frac{TSFC \times T}{V} = TSFC \times \left(\frac{T}{V}\right)$$

The dependent variable in this case is T_R/V_∞. The minimization of fuel burn per unit distance, illustrated in Figure 2-6, is achieved by the value that represents the one and only line tangent to the fuel burn curve.

Therefore, if optimizing over a fixed period of time, minimizing the fuel burn with

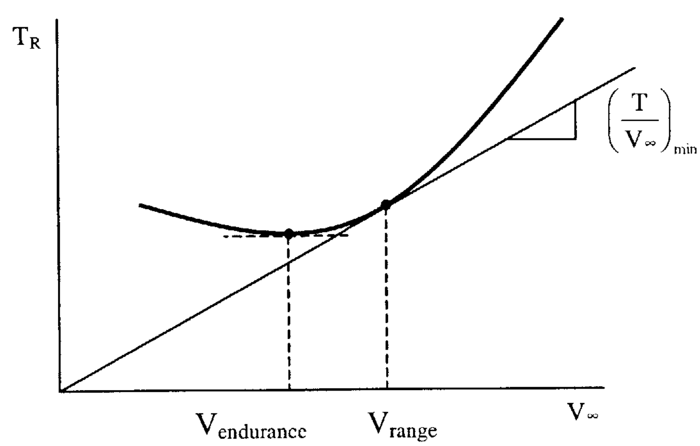


Figure 2-4: Typical Thrust vs. Flight Speed Curve

respect to time is the objective definition that should be considered when redistributing corridor traffic. In addition, considering the aforementioned fact that engine performance itself varies with altitude, there exists a preferred design altitude of maximum aircraft performance and thus, minimum aircraft operating cost from the perspective of the airlines. Indeed, this is the objective chosen when optimizing with respect to fuel burn.

Thus, the cumulative fuel burn objective was used to answer the following fundamental research questions:

- How does the redistribution change when other factors change?
- How severely is the rest of the corridor traffic penalized in terms of operating costs?

2.4.3 Metric Costs

The cost of delaying any given flight is listed in Table 2-11 for each aircraft model. These values were used to associate a monetary value for the delay reductions determined in the simulations [Malconian, 2001].

Similarly, the fuel cost of delaying any flight is established using the latest market price for Kerosene-Type Jet Fuel in the Northeast: $0.85/gallon [DOE, 2003]. With the density of Kerosene-Type Jet Fuel given as 3.1 kg/gallon, this cost can be alternatively expressed as $0.27/kg.

Table 2-6: Delay Costs per Block Hour	
Aircraft Model	($/minute)
Boeing 737	$1874
Boeing 757	$2316
Boeing MD80	$1835
Airbus A320	$1959
Embraer EMB145	$1498

2.4.4 Mixed Strategies

Thus far, two separate objectives have been mentioned:

- Minimization of cumulative aircraft delay
- Minimization of cumulative aircraft fuel burn

However, the optimal strategy may involve some combination of these two basic objectives because airlines incur costs both as a function of their operations as well as the degree of delays. That is, the optimum objective strategy from the perspective of the airlines may involve some combination of optimizing with respect to fuel burn and delay. Thus, mixed strategies that consider both objectives were also explored in order to ascertain how the distribution changes as a function of the relative weight given to each of the objectives.

The mixed strategy objective is expressed as follows:

$$\begin{aligned}&objective = C_{delay} \times D + C_{fuel\ burn} \times F, where:\\&C_{delay} - \text{Cost of Flight Delays}\\&C_{fuel\ burn} - \text{Cost of Aircraft Fuel Burn}\\&D - \text{Cumulative Flight Delay}\\&F - \text{Cumulative Aircraft Fuel Burn}\end{aligned}$$

$$objective = Cost_{delay} \times Cumulative\ Delay + Cost_{fuel\,burn} \times Cumulative\ Fuel\ Burn$$

The costs used in the objective function are those listed in the previous section, namely the Delay Costs per Block Hour and the Cost of Kerosene-Type Jet Fuel.

Thus, a mixed strategy was used to answer the following fundamental research question:

- What are the total cost savings that aircraft in the corridor can achieve if the mixed strategy objective is used?

Defining the metrics is the first step in developing the final stage in the Research Approach: the decision model. However, the development of the decision model requires a deep knowledge of Integer Programming Techniques. Thus, the next two chapters will provide the necessary background for developing the last piece of the Research Approach presented here.

3 Integer Programming

As the name suggests, Integer Programming problems optimize over feasible sets of integer variables. Unlike many problems in Linear Programming, where the feasible sets of variables are continuous, Integer Programs are typically very difficult to solve in practice. Many Integer Programs are exponentially complex in the number of variables and/or constraints present and often fall under a category of problems referred to as being NP-Hard. The absence of any efficient general algorithm for this category of problems is a primary topic of Applied Mathematics research and often motivates the development of analysis techniques for specific problem instances, as is the case with this research.

In the first section of this chapter, three common techniques used to solve Integer Programming problems are explored and the potential benefits and drawbacks of each one are discussed. The resulting complexity of the chosen technique is shown to fundamentally involve the efficient solution of an associated search problem. Therefore, in the second section of this chapter, several techniques used to solve search problems are discussed within the context of subsequently developing an efficient search algorithm within the core of the final analysis structure.

3.1 Integer Programming Techniques

The fundamental difficulty in solving Integer Programs arises from the disjoint nature of the feasible set of integers in the optimization model. A cursory comparison to the solution techniques commonly used to solve Linear Programs offers a way in which to ascertain this difficulty. Consider that many of the techniques used to solve Linear Programs take advantage of the guarantee that the optimal solution exists at a corner point. Thus, techniques like the Simplex Method work by systematically moving from extreme point to extreme point until encountering one point such that the cost of moving away from it in any feasible direction results in a higher objective function cost. Fundamentally, this type of approach depends on a feasible set representation consisting of a bounded polyhedron that is convex. Convexity of the feasible region is defined as:

Definition 3.1 A Set $S \subset \Re^n$ is convex if for any $\mathbf{x}, \mathbf{y} \in S$, and any $\lambda \in [0,1]$, we have $\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in S$.

The importance of this property can be explained with the aid of Figure 3-1. Set S is convex, while set T is not. The optimal solution for both sets for a given cost vector c exists at the point of tangency with each set. Notice that only one such point exists for the convex set S, while two such points exist for the non-convex set T. Thus, the impact of convexity on solution quality is derived from the fact that while convex sets contain only one global optimum for a given cost vector, there is no such guarantee for non-

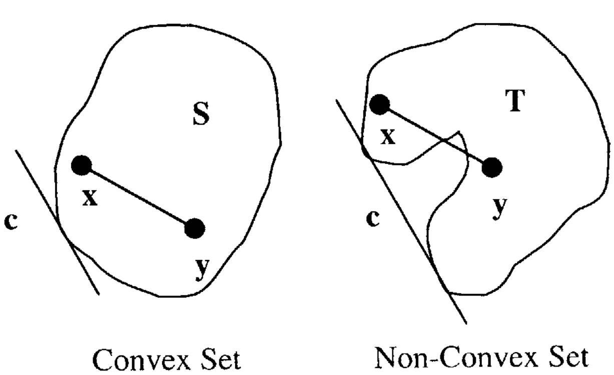


Figure 3-1: Set Convexity Example

convex sets. For problems characterized by a non-convex feasible region, each local extreme must be considered before the global optimum can be identified. This is the crux of the fundamental difficulty in solving Integer Programs: the feasible region is disjoint and thus, non-convex.

With this property in mind, three of the most common techniques used to solve Integer Programs involve:

- Cutting Plane Methods
- Integer Duality Theory
- Branch and Bound

3.1.1 Cutting Plane Methods

Cutting Plane Methods solve a sequence of linear programs until an integer solution is found. This integer solution is guaranteed to represent the optimal integer solution to the original integer program [Bertsimas and Tsitsiklis, 1997]. Let an integer program be denoted by I as follows:

$$\begin{aligned}
&I : minimize\, c' x \\
&subject\, to \\
&Ax = b \\
&x \in \mathbb{Z}
\end{aligned}$$

In addition, let the following linear program, denoted by L, represent the relaxation of I:

$$
\begin{aligned}
&L : minimize\, c' x \\
&subject\, to \\
&Ax = b \\
&x \geq 0
\end{aligned}
$$

Using these definitions, a basic cutting plane algorithm proceeds as follows:

1. Solve the linear programming relaxation, *L*, of the integer program, *I*. The solution is denoted as *x*.
2. If *x* is an integer solution, then stop. This value corresponds to the optimal solution to *I*.
3. If *x* is not an integer solution, then add a linear inequality constraint to *L* that all integer solutions to *I* satisfy, but *x* does not.
4. Go to step 1.

At each iteration, the algorithm generates a constraint that the non-integer solution x^i violates. Thus, in subsequent iterations, the algorithm does not generate x^i ever again. The manner in which the cutting plane algorithm works is illustrated in Figure 3-2. The original feasible region is depicted as the shaded polygon, with the set of feasible integer solutions represented by all the pairs of integer points residing within this region. The first three steps of a generic cutting plane method are illustrated. The initial iteration results in the non-integer solution x^1. Thus, a linear inequality is generated and added to *L*, effectively pruning out x^1 from the feasible space of subsequent iterations. Note that in making the cut, none of the integer solutions in the feasible integer space were pruned. The non-integer solution x^2 is generated in the second iteratation. Again, a linear

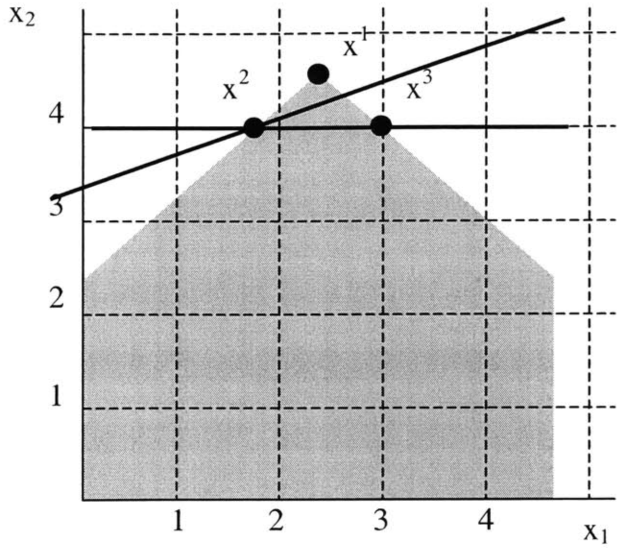


Figure 3-2: Cutting Plane Example

inequality is generated and added to *L*, pruning x^2 from the feasible space of subsequent iterations.

Note that the algorithm continues generating cuts until one is generated in which the resulting feasible region has an integer extreme point. Thus, the performance of the algorithm is seen to depend on the nature of the cutting heuristic used to generate the cuts. One such method is the Gomory Cutting Plane Algorithm, which was the first finitely terminating algorithm developed for integer programs [Bertsimas and Tsitsiklis, 1997].

3.1.2 Integer Duality Theory

One of the fundamental foundations for linear programming theory involves the concept of the Dual Problem. The Dual Problem is an alternative representation to a linear program that can be used to find the optimal solution when the original representation is characterized by a notoriously difficult, complex and highly coupled constraint set. The foundation of Duality Theory is predicated on a property known as Strong Duality, which can be stated as follows:

> **Definition 3.2** If a linear program has an optimal solution, so does its dual, and the respective optimal costs are equal [Bertsimas and Tsitsiklis, 1997].

In essence, the dual problem is what results from "relaxing" some or all of the constraints that characterize the feasible region of the original linear program. To relax a constraint means to remove the constraint from the constraint set, thus effectively increasing the feasible region to include previously infeasible solutions. Thus, a penalty is now associated with these solutions rather than dismissing them from consideration, as before.

In order to better explain the nature of the dual problem, consider the following Linear Program L:

$$\begin{aligned} &L: \textit{minimize}\; c'x \\ &\textit{subject to} \\ &Ax = b \\ &x \geq 0 \end{aligned}$$

Finding a feasible solution to L may entail significant computational effort, as the feasible region defined by the constraint set may be very complex in nature. Often, a high degree of coupling between the variables is the main culprit. As such, consider the ramifications of relaxing the constraints in L:

$$\begin{aligned}&minimize\, c'x + p'(b - Ax)\\&subject\ to\\&x \geq 0\end{aligned}$$

The first observation of importance is that the feasible region of this relaxed problem is larger than that of the original linear program. As such, the optimal objective function value of this problem could be lower than the original linear program, despite the penalty p associated with choosing previously infeasible solutions. The importance of this last statement is that the solution to this relaxed problem is guaranteed to be a lower bound on the optimal objective function value, denoted as $c'x^*$, of the original linear program:

$$\begin{aligned}&g(p) = \min_{x\geq 0}\left[c'x + p'(b - Ax)\right] \leq c'x^* + p'(b - Ax) = c'x^*\\&\therefore\ g(p) \leq c'x^*\end{aligned}$$

This property is known as Weak Duality. Thus, intuition suggests that the objective of the relaxed problem should be maximized so that the gap between its optimal solution and the optimal solution to the original linear program is decreased. This results in the definition of the dual problem D:

$$\begin{aligned}&D: maximize\left\{\min_{x\geq 0}\left[c'x + p'(b - Ax)\right]\right\}\\&\Rightarrow maximize\left\{p'b + \min_{x\geq 0}\left(c' - p'A\right)x\right\}\\&\min_{x\geq 0}\left(c' - p'A\right)x = \left\{\begin{array}{ll}0, & if\ c' - p'A > 0\\ \hline -\infty, & otherwise\end{array}\right\}\\&\therefore\ D: maximize\ p'b\\&\qquad subject\ to\ p'A \leq c\end{aligned}$$

Finally, the aforementioned property of strong duality guarantees that this gap vanishes, resulting in the equality of the objective function values for L and D.

In the case of an integer program, there is no property tantamount to the Strong Duality property for linear programs. That is, the optimal solution to the dual of an integer program is not guaranteed to return the optimal solution of the original integer program. However, the dual of an integer program does exhibit the property of weak duality. Namely, the dual provides a lower bound on the optimal solution of the integer program. This property is useful because the solution to the dual can be used to calculate the bounds used in other integer programming methods, such as Branch & Bound. In fact it has been shown that the following ordering holds among the integer program z_{IP}, its linear relaxation z_{LP}, and its dual z_D:

$$z_{LP} \leq z_D \leq z_{IP}$$

Thus, the dual provides a tighter lower bound on the optimal value to the integer program than that obtained by simply solving the linear relaxation.

3.1.3 Branch and Bound

In absence of any formal knowledge of the more sophisticated Integer Programming techniques that have been developed, the most intuitive approach involves the enumeration and solution of all feasible integer combinations of the set of variables. This is especially obvious in the case where the integer variables take on the form of binary decision variables. As illustrated in Figure 3-3, for the example of $x_1, x_2 \in \{0,1\}$, this is equivalent to searching the whole tree of solutions that characterizes the entire feasible integer space.

However, even in the case of binary decision variables, the set of all possible permutations grows explosively at the rate of 2^n, where n is the number of variables. Thus, simply exploring the entire feasible integer solution space is deemed impractical for all but the smallest problem instances. Such a method could be rendered tractable if it could avoid solving a large number of these permutations, especially when considering the fact that each solution instance has the potential to represent a bound on the optimal solution. This is the crux of the Branch and Bound method used to solve integer programs.

Central to the Branch and Bound method is the decomposition of the problem into a

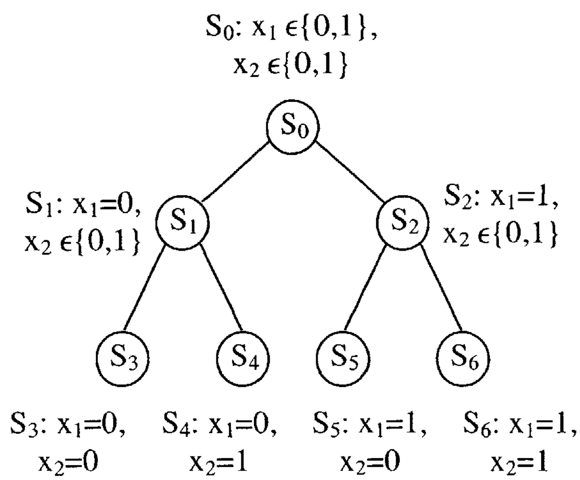


Figure 3-3: Branch & Bound Example

sequence of subproblem instances that are used to efficiently update the bound on the optimal solution. As illustrated in Figure 3-3, each subproblem represents a portion of the feasible integer space. For example, the subproblem S_1 is defined as the solution space where $x_1 = 0$ and x_2 is a free variable. Note that there exists a certain measure of freedom in how exactly a subproblem is defined.

One crucial aspect that makes the Branch and Bound method attractive is the notion that each subproblem need not be solved to optimality. Rather, only a lower bound on the subproblem solution is needed. Therefore, one popular approach used to obtain a lower bound involves solving the linear relaxation of the subproblem instance. Referring again to the example of Figure 3-3, assume that the linear relaxation of the subproblem instance S_1 returns an optimal solution such that $x_2 = 0.5$. One subsequent suproblem instance S_3 can be defined by imposing the additional constraint $x_2 \leq 0$ to the original constraint set. Similarly, an alternate subproblem instance S_4 may be defined by imposing the constraint $x_2 \geq 1$. These two alternatives represent the branching that takes place when exploring the space of feasible integer solutions.

Because this simple example uses binary decision variables, both of the subsequent subproblem instances are guaranteed to return a solution such that x_2 is integer. This is not the case for integer variables in general. However, the key notion is that if both subproblem instances are assumed feasible, at least one of the branches is guaranteed to define a subproblem instance for which the linear relaxation will return an integer solution for x_2. That is, one of these constraints will force subsequent linear relaxation solutions to an integer bound, which represents the new, lowest cost solution point for x_2. The other will not, but its subproblem instance may still represent a node that will be further branched in the exploration of the solution space.

Whether either of these subproblem instances serves as a root node for further branching depends on the objective value of their linear relaxation, which was previously shown to be a lower bound on its integer objective function value. The sufficiency of calculating the subproblem lower bound, b_i, is based on the fact that it can be used to prune certain portions of the feasible solution space when used in concert with an updated value for the upper bound, or best estimate, of the solution to the integer program. That is, at any point during the search, there is a best estimate value, U, of the optimal objective value of the integer program. The value is an upper bound on the optimal value, since further searching may produce a better value. Thus, if $b_i < U$ and b_i is integer, then the upper bound is updated so that $U = b_i$ and the space branching from subproblem i remains in the search space left for further exploration. Conversely, if $b_i \geq U$ then no further exploration of the space branching from subproblem i need be explored. This statement is based on the observation that the algorithm has already identified an alternate portion of the solution space that is guaranteed to provide a better bound on the objective function value of the original integer program.

Using these observations, a basic Branch and Bound algorithm can be constructed as follows:

1. Set U to be an arbitrarily large number

2. Repeat steps 3-9 until the active subproblem list S is empty
3. Select an active subproblem S_i
4. If S_i is infeasible, then omit it from further consideration
5. If S_i is feasible, then compute b_i
6. If $b_i \geq U$, then omit it from further consideration
7. If $b_i < U$ and b_i is integer, then set $U = b_i$
8. If $b_i < U$ and b_i is non-integer, then branch S_i into two further subproblem instances which are added to the list of active subproblem instances
9. Return to step 2
10. Return U

Note that during the course of the Branch and Bound procedure, there exists an active list of subproblem instances to be explored. Thus, another measure of freedom in the implementation of the procedure is how the search space is explored. This issue boils down to the matter of a search and the performance of the algorithm. This topic is explored in subsequent sections.

4 Search Methods

The relevance of search heuristics when used within the context of solving non-convex optimization problems was introduced in the previous section. An in-depth discussion of the most commonly used search heuristics and how they can be used to solve integer programs using a Branch & Bound implementation is presented in this chapter. In general, all search heuristics can be classified either as a uniformed or informed search method. Examples of both categories of searches are presented and compared.

4.1 Uninformed Search Methods

Knowledge can be acquired in one of two ways: by experience or by estimation. More strictly, the former process results from the transitions that a goal-based agent has already undertaken. The agent is privy to its current location in the solution space relative to where the process began as well as to the sequence of transitions that it undertook in arriving there. This knowledge, coupled with the ability to identify the goal state, allows the agent to test whether the current state is the goal state and bases its next transition on this collection of information. Such a search is called uninformed because there is no knowledge as to the "value" of the current state relative to the goal state, and thus, no manner in which to base the impending transition decision on what the expected "value" of the next state will be. However, it is often the case that a problem may harbor some discernable structure, which may then be used in deciding which uninformed search heuristic to implement. Example heuristics presented here include:

- Depth First Search

- Breadth First Search
- Uniform Cost Search

4.1.1 Depth First Search

Several stages of a search tree during the course of a Depth First Search are illustrated in Figure 4-1. This search heuristic explores the node furthest down along all of the available branches. Therefore, when the problem at hand involves a goal state that can be described by a large number of possible configurations, then the Depth Fist Search should be applied. The reason for this is that a focused search along one region of the solution space would most likely encounter a goal state using minimal computational effort.

4.1.2 Breadth First Search

On the other hand, when the nodes in a search space are not expected to spawn a great number of nodes, then the argument can be made to explore the solution space in a more

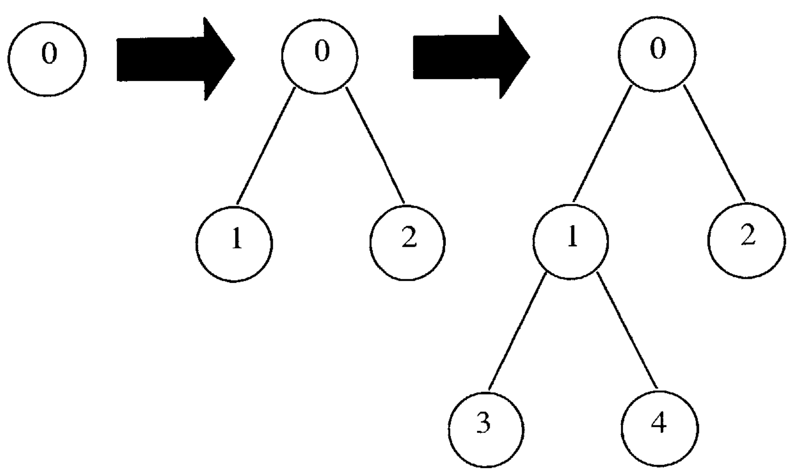


Figure 4-1: Depth-First Search Illustration

uniform manner. This is the basis of the Breadth First Search, the first several stages of which are illustrated in Figure 4-2. Note that the Breadth First Search operates by exploring and expanding the last node on each branch in the search tree. Thus, the search expands the branches of the search tree uniformly, resulting in the exploration of an increasing fraction of the solution space.

4.1.3 Uniform-cost Search

The Uniform-cost Search falls under the Uninformed Search category as its search strategy is unilaterally dictated by the "cost" of the search path undertaken thus far. That is, it always formulates a search path of minimal cost until it reaches a goal state. The goal state reached is guaranteed to be the least-cost solution by virtue of the minimal cost nature of the strategy and can be proved in a very informal manner. Simply put, if there exists a less costly goal, the path would have been constructed to reach it because of the mantra of the strategy: always expand the minimal cost path. This proof enforces the rule that a path must monotonically increase. Equivalently, this means that the cost of transitions must always assume a non-negative value. Without such a requirement the

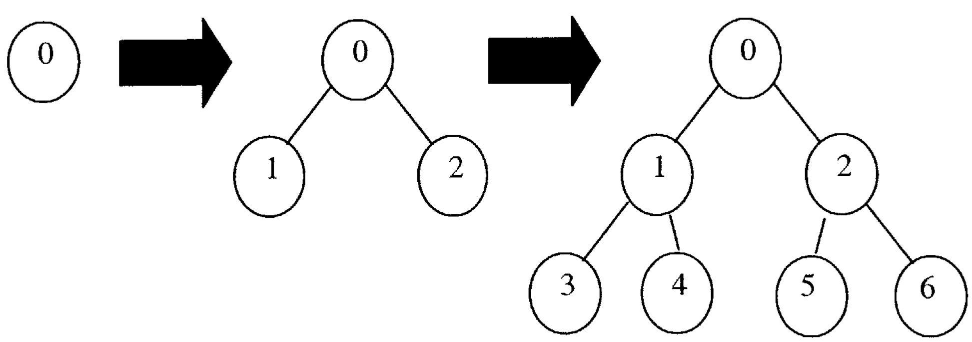


Figure 4-2: Breadth-First Search Illustration

search could never produce the least-cost goal state without exploring the whole search tree.

An example of how the Uniform-cost Search operates is illustrated in Figure 4-3. In the first step of the strategy the expansion of node 2 is chosen over nodes 1 and 3 because of the lower path cost of going from node a to node c versus the other two available paths. Of particular importance is the classification of nodes 1, 2, and 3 as "fringe nodes" because of the fact that they represent the set of nodes used in determining the transition strategy. While this point seems trivial during the first step, it is a subtle yet powerful expression of the parent-child node mapping inherent in the search tree and is best rationalized in the description of the subsequent transition provided in the example. After expanding node 2, the fringe node set is comprised of nodes 1, 3, 4, 5, and 6. Considering the cost of the paths so far, and assuming that none of the states reached is a goal state, the next transition involves an expansion of node 1 since the path to node 1 is the now the minimum cost path. After the second transition, the fringe node set is comprised of nodes 3, 4, 5, 6, 7, 8, and 9.

Based on this simple example involving only two transitions, the explosive growth in the computational complexity of the Uniform-cost search is readily observed. That is, the ignorance of any estimate of the value of the fringe nodes relative to the goal state, the Uniform-cost search is deemed NP Hard due to the non-polynomial computational complexity of obtaining the solution.

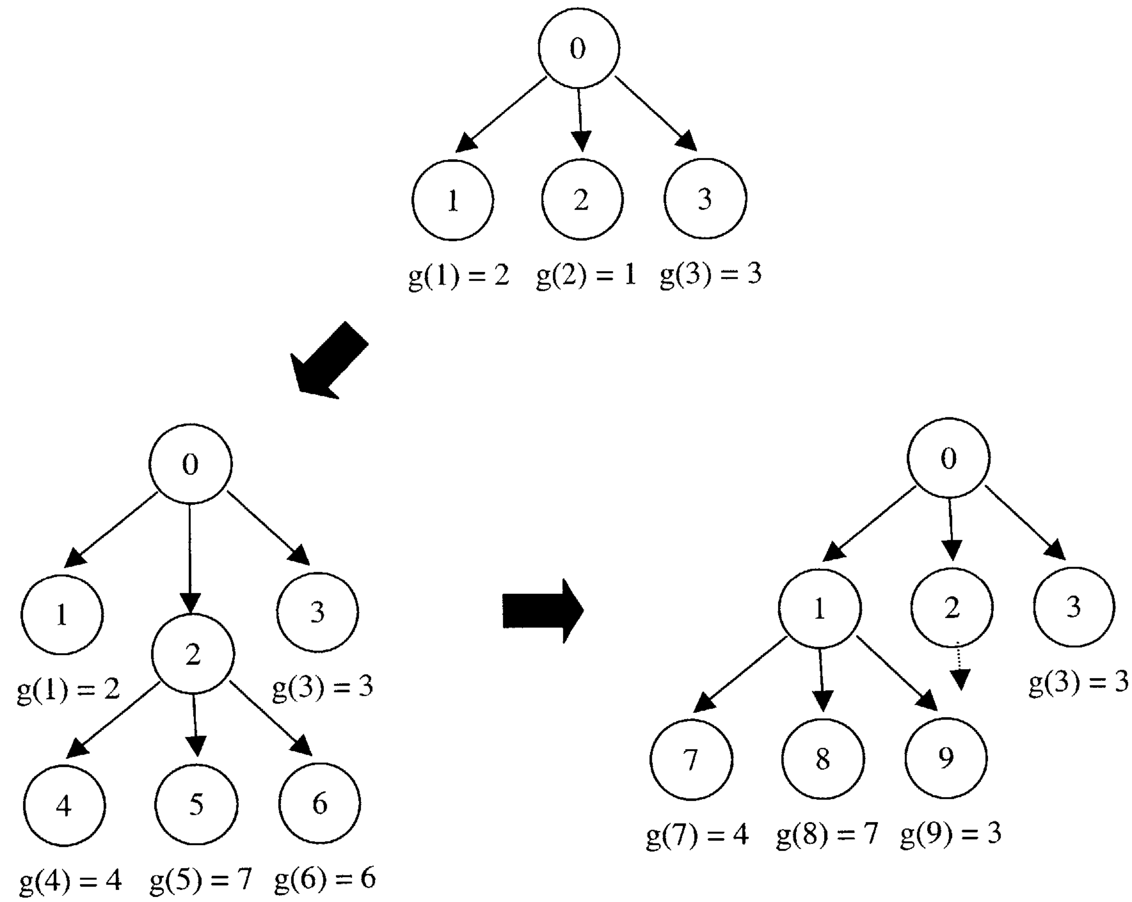


Figure 4-3: Uniform-cost Search Illustration

4.2 Informed Search Methods

Any search instance could benefit immensely from information about the set of possible future states. The information need not definitively lead to the goal state; rather it should be some useful estimate of the necessary sequence of transitions. This is the foundation of the class of Informed Search Methods. The benefit of implementing an informed search is the marked decrease in computational complexity that it affords. One such method is detailed here: the A* Search.

4.2.1 A* Search

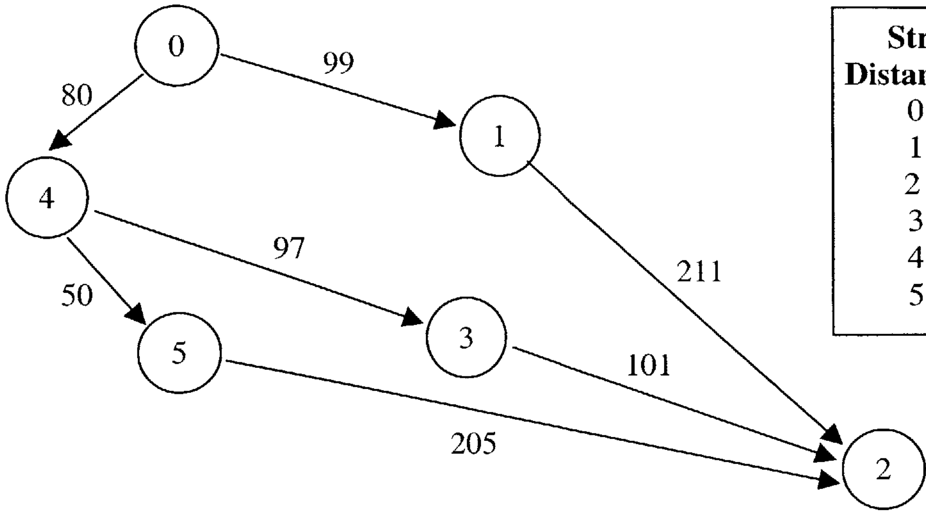
Basing the transition policy solely upon the strategy to minimize the cost to the goal, without regard to the cost of the search thus far is known as a greedy search. This kind of search is neither complete nor optimal. That is, there is no guarantee that the search will terminate, or that it will converge upon the lowest cost goal state because of the unilateral dependence on the goal-seeking heuristic.

However, a search methodology that combines the benefits of a greedy search heuristic with the completeness and optimality guarantees of a uniform-cost search would represent an ideal search strategy. One such example is the A* Search, whose behavior will be illustrated via example. The A* Search heuristic is composed by summing the path cost of the transition policy thus far with a greedy, forward-looking heuristic:

$$
\begin{aligned}
&f(n) = g(n) + h(n), where: \\
&g(n) = Path\ cost\ thus\ far \\
&h(n) = Path\ cost\ estimate\ to\ goal\ via\ n
\end{aligned}
$$

The former component represents the Uniform-cost Search portion of the heuristic while the latter component represents the greedy search portion of the heuristic. Such a composition takes advantage of the natural segregation of knowledge into experience-based and estimation-based.

An example problem that demonstrates the benefits and completeness of the A* Search is illustrated in Figure 4-4. In the example, the goal consists of finding the optimal route to node 2 from node 0. The mapping between nodes represents the distance between two



Straight Line Distances to Node 2	
0	253
1	178
2	0
3	98
4	193
5	192

Figure 4-4: A* Search Example Mapping

adjacent nodes, while the table lists the straight line "distance" from each node to node 2. The example can be rationalized as a simple shortest path problem where the nodes represent intermediate towns between the departure town and the destination town. The mapping is representative of the mileage incurred on the roadways between two towns, while the tabulated data is representative of the straight-line geographical distances. The former can be used in composing the Uniform-cost Search portion of the heuristic while the latter can be used in composing the Greedy Search portion of the heuristic. The search strategy resulting from the A* Search is illustrated in Figure 4-5.

Proving the completeness of the A* Search is performed in a similar fashion to the previous proof for the Uniform-cost Search. The intuition is very similar: a monotonically increasing path will eventually increase until it has converged upon a goal state. However, proving the optimality of the solution requires a more formal proof. Assume there exists an optimal goal state G and a sub-optimal goal state G_2.

Furthermore, assume that G_2 was returned by the A* search. The development that follows will prove that the A* search can never return G_2, thereby guaranteeing optimality. Consider that at an arbitrary time during the search there exists a node n that is on the optimal path to G. The existence of such a node is guaranteed unless the path has been completely explored, in which case the A* Search would have already encountered goal state G. At this juncture, it should be noted that monotonic heuristics fall under the more general category of admissible heuristics, which simply require that the heuristics never overestimate the cost of reaching a goal state:

$$\begin{aligned} f^* &\geq f(n), \; where: \\ f^* &= actual\ path\cos t\ to\ goal\ state\ G \\ f(n) &= estimated\ path\cos t\ to\ goal\ state\ through\ n \end{aligned}$$

Next, assume that a path to G_2 is chosen over the further exploration beyond node n, which implies:

$$f(n) \geq f(G_2)$$

Combing these two inequalities yields:

$$f^* \geq f(G_2)$$

Because G_2 is a goal state, the estimate part of the A* heuristic is equal to zero, resulting in:

$$\begin{aligned} h(G_2) &= 0 \\ \Rightarrow f(G_2) &= g(G_2) \\ \therefore f^* &\geq g(G_2) \end{aligned}$$

This last constraint asserts that the path cost to goal G_2 is cheaper than the path cost to goal G, contradicting the original assumption that goal G is the optimal state. Thus, the A* Search is always guaranteed to return the optimal goal state.

The resulting analysis framework in this research uses the A* Search because of its potential to markedly reduce the computational complexity in terms of time and memory of the logic developed for the simulations. The critical element in successfully implementing the A* Search in this application is to define an appropriate admissible heuristic for the Branch & Bound procedure used in the process of solving the MILP. The resulting heuristic is described in the next section since it requires the intuition gained from defining the MILP.

5 MILP Decision Model

Having provided a background for Integer Programming methods in previous sections, the intention in this section is to develop the model used for the subsequent analyses. The model is a MILP with binary decision variables used to establish both traffic sequencing and flight level allocation. A decomposition of the binary decision variables into two distinct sets is presented and shown to substantially reduce problem complexity. The complete model is then summarized and an analytical discussion is provided in order to demonstrate that the model is computationally tractable.

5.1 Set Definitions

The following two sets are used to develop the model:

N – Set of aircraft in the network
M – Set of available flight levels

The set N is obvious in its definition and requires no further explanation. However, the set M may change depending on the situation being analyzed. While the set of feasible flight levels was previously defined when discussing the characteristics of the En Route corridor, defining M as such may not be necessary or even sufficient in some cases. For example, if the analysis at hand involves the characterization of the tunneling phenomenon previously discussed, then an extra flight level needs to be added to the set of available flight levels. In addition, a fair assumption may be to assume that if each flight level is congested to the same degree then only a subset of the available flight levels need be studied in order to derive any possible improvement on a per-flight basis. This assumption constitutes one of the questions explored within the body of this research.

5.2 Input Parameters

The following parameters define a particular problem instance and are input into the model:

T - Time horizon over which the optimization is performed
s - Minimum separation standard between aircraft (5 nm = 30380.5 ft)
x_{0_i} - Initial range position of each aircraft $i \in N$
v_{0_i} - Initial longitudinal speed of each aircraft $i \in N$
$v^k_{\max i}$ - Maximum speed on each flight level $k \in N$ for each aircraft $i \in N$
$v^k_{\min i}$ - Minimum speed on each flight level $k \in N$ for each aircraft $i \in N$
$a_{\max}$ - Maximum longitudal acceleration for all aircraft (2 fps)
$a_{\min}$ - Minimum longitudal acceleration for all aircraft (-2 fps)

5.2.1 Time Horizon Input Parameter

As previously discussed, optimization problems involving vehicle dynamics often use a discretized time interval set, resulting in the modeling of each dynamic variable at each interval. The resulting complexity of these representations is a function of the discretization of the given time set T. While this may not be a problem in a single vehicle problem because of the continuous nature of all of the variables involved, it becomes prohibitively complex in any problem with a significant degree of integer variables.

Fortunately, the great deal of structure fundamentally inherent in the problem at hand enables an alternate, computationally tractable formulation. Consider that the mechanism that air traffic controllers use when monitoring and controlling network traffic consists of issuing a command flight level and speed to each aircraft. That is, a flight is instructed to maintain a certain speed and to transition to a specific flight level. Thus, the controller is interested in the final aircraft state, rather than the particular dynamics and kinematics states that the aircraft will go through in the given time. This argument is valid with the implicit understanding that controllers are mindful of basic aircraft performance limits and thus do not issue commands that can not be satisfied by the aircraft.

Similarly, the modeling of the problem at hand can be designed such that the optimization is performed by manipulating the final, "command" states for each aircraft. Specifically, these include the final aircraft flight level and speed distributions. As with the real world case, all commands issued to the aircraft in the network conform to stipulated performance capabilities and safety requirements.

5.2.2 Initial Position and Speed Parameters

As previously discussed, the corridor structure considered here is 2-dimensional in nature, and requires the definition of only two position coordinates: range and flight level. The initial range, x_0, can be specified from any arbitrary origin. It is assumed that all aircraft are in static, level flight; thus only the initial longitudinal speed, v_0, must be defined.

5.2.3 Performance Input Parameters

The maximum and minimum aircraft speeds, $v^k_{\max i}$ and $v^k_{\min i}$ respectively, are both flight and flight-level specific. Thus, this representation accounts for the significant variations in performance that each aircraft experiences at different flight levels. However, note that the longitudinal accelerations are independent of aircraft type and flight level. These limits are a realistic representations of limits used to ensure passenger safety and comfort in all commercial flights.

5.3 Variables

The following variables are used to optimize the problem:

$x_i, i \in N$ - Range at time horizon T
$v_i, i \in N$ - Longitudinal speed at time horizon T
$a_i, i \in N$ - Command accelertion used to speed-up/slow-down flight
$y_{ij}, i \in N, j \in N$ - Sequencing binary decision variable
$z_i^k, i \in N, k \in M$ - Flight Level binary decision variable
$f_i, i \in N$ - Aircraft Fuel Burn (used only when optimizing w.r.t. fuel burn)

5.3.1 Kinematics Variables

The range and velocity, x_i & v_i respectively, are the values that characterize the position of each flight in the network at the time horizon T. The acceleration, a_i, is used to achieve these values. Thus, the main benefit of using the time horizon representation is made clear at this juncture; rather than having to consider the range and speed for each flight at each time t in a time-discretized representation, only the final range and speed are considered. This reduces the problem complexity by the number of time intervals used to define the discretized interval T: x_i & v_i versus x_{il} & v_{il}, where l is the number of time intervals in T. This is a valid formulation so long as attention is paid to maintaining the appropriate separation standards among all aircraft in the airspace. This stipulation will be satisfied when formulating the separation standards as part of the constraint set.

5.3.2 Flight Level and Flight Sequencing Binary Decision Variables

Two sets of binary decision variables are defined, with z_i^k used to assign aircraft i to flight level k and y_{ij} used to sequence aircraft j ahead of aircraft i ($x_j > x_i$). For example, if Flight 1 is assigned to Flight Level 3, then $z_1^3 = 1$ and $z_1^k = 0 \forall k \in M \mid k \neq 3$. In addition, if Flight 1 is instructed to go ahead of Flight 2 (i.e. $x_1 > x_2$) then $y_{21} = 1$ and $y_{12} = 0$. Note that these variables establish a sequencing order only when Flight 1 and Flight 2 are both instructed to go to the same Flight Level. If Flight 1 and Flight 2 are instructed to go to different Flight Levels, then these variables do not provide any sense of sequencing.

Decomposing the mathematical decision structure in this way decreases the complexity of the resulting formulation from $O(n^2m)$ to $O(n^2)$ in the number of variables. Consider that an alternate formulation may instead group these two decisions into one binary decision variable, y_{ij}^k. Referencing the example given above: If Flight Level 1 and Flight Level 2 are both commanded onto Flight Level 3, with $x_1 > x_2$ then $y_{21}^3 = 1$ and $y_{12}^3 = 0$.

Also note that $y_{1j}^{k} = 0\,\forall j \in N, \forall k \in M$ and $y_{j1}^{k} = 0\,\forall j \in N, \forall k \in M$. However, the nature of this binary decision variable is such that it must be used to impose a strict sequencing order. That is, whereas y_{ij} signifies that Flight *j* is further ahead in range than Flight *i*, y_{ij}^{k} must be used to signify that Flight *j* is directly ahead of Flight *i* on Flight Level *k*. The significance of this distinction can be appreciated by comparing the complexity of both representations: the order of this latter representation is $O(n^2m)$ in the number of variables, while the order of the formulation chosen is $O(n^2)$ in the number of variables.

The fundamental reason why a strict sequencing decision variable such as y_{ij}^{k} need not be used is based on the definition of the variables x_i and the subsequent enforcement of separation among flights, which together infer a degree of sequencing. If $x_i > x_j$, then it is enough to enforce $|x_i - x_j| \geq \delta$, where δ is the longitudinal separation standard, among only those flights that are commanded onto the same flight level rather than among all of the flights in the network. In addition, the use of binary decision variables of the form y_{ij}^{k} lead to the definition of a far greater complex constraint set. A proper defense of this statement is deferred until the definition of the constraint set.

5.4 Objective Function Definitions

The fact that the analysis is based on the definition of two distinct metrics begs the definition of two distinct objective functions. Specifically, one objective will be used to minimize cumulative network delay, while the other will be used to minimize the cumulative fuel burn in the network. As previously argued, the utility of the former metric is that it can be used both as a means to gauge how much additional capacity can be "recovered" from the current corridor conditions and as a means to establish a foundation to analyze the properties of the solution and its sensitivity to changes in various parameters. The latter metric is more representative of what the real-world objective of an aircraft in the network might be: to minimize the operating costs of the aircraft by minimizing the fuel expended.

5.4.1 Cumulative Delay Minimization Objective Function

Having specified a time horizon T over which to optimize, the minimization of delay for a specific flight i involves the maximization of its range x_i at T. That is, the minimization of delay for any given flight over a fixed distance is equivalent to maximizing the distance traveled in a fixed interval of time. These two representations are equivalent because they express the same fundamental objective in the network: to increase throughput. In addition, to increase throughput equitably in this formulation equates to maximizing the total distance traveled within T for each aircraft in the network with equal emphasis. In other words, no preference is overtly given to one aircraft over another. Therefore, the mathematical representation of this objective can be expressed succinctly as follows:

$$\max \sum_{i \in N} x_i$$

5.4.2 Cumulative Fuel Burn Minimization Objective Function

The difficulty in minimizing with respect to fuel burn resides in the quadratic nature of the fuel burn as a function of flight speed that was previously discussed. However, this function is convex in nature and can be approximated using a convex set of piecewise linear functions. The property of convexity is illustrated in Figure 5.1 and is defined as follows:

Definition 5.1: A function $f : \Re^n \mapsto \Re$ is convex if $\forall x, y \in \Re^n$, and $\forall \lambda \in [0,1]$:

$$f(\lambda x + (1-\lambda) y) \leq \lambda x + (1-\lambda) y$$

As Figure 5.1 illustrates, the property of convexity is fulfilled if the chord between any two points on the given function always resides tangent to or above the function.

Thus, the quadratic fuel burn function for each flight level can be approximated via the use of a piecewise linear function of the form illustrated in Figure 5.2. The number of segments that comprise the linear approximation is a design choice with a greater number of segments resulting in a higher-fidelity model of the function. For the flight level-dependent fuel burn functions considered in this research, linear approximations

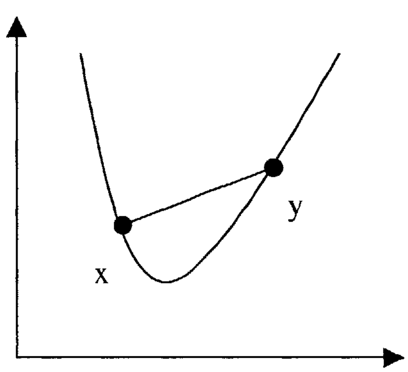


Figure 5-1: Example of Function Convexity

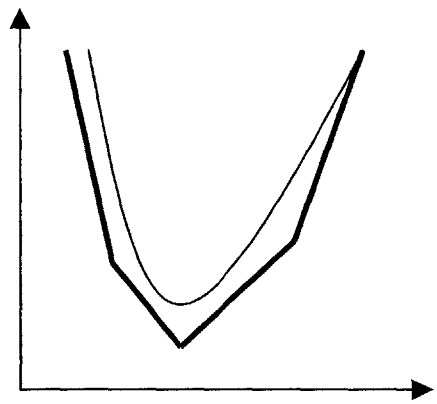


Figure 5-2: Piecewise Convex Linear Approximation

consisting of four segments were chosen and found to be approximations of sufficient fidelity.

The linear approximations were found as solutions to the following LP formulation:

$$\min \sum_{i \in N} \left| y_i - y_i^* \right| = \min \sum_{i \in N} \gamma_i$$

$$\begin{aligned}
&\textit{subject to:} \\
&y_i - y_i^* \leq \gamma_i, \forall i \in N \\
&y_i^* - y_i \leq \gamma_i, \forall i \in N \\
&a_1 x_i + b_1 = y_i, \forall i \in N_1 \\
&a_2 x_i + b_2 = y_i, \forall i \in N_2 \\
&a_3 x_i + b_3 = y_i, \forall i \in N_3 \\
&a_4 x_i + b_4 = y_i, \forall i \in N_4
\end{aligned}$$

Where the following definitions apply:

Sets:

N - Set of all air speed values i

N_1 - Set of air speed values for segment 1

N_2 - Set of air speed values for segment 2

N_3 - Set of air speed values for segment 3

N_4 - Set of air speed values for segment 4

Parameters:

y_i^* - Quadratic fuel burn value at air speed i

x_i - True air speed i

Variables:

y_i - Linear approximation fuel burn value at air speed i

a_i - Slope of segment i

b_i - Y-intercept of segment i

γ_i - Dummy variable for $\left|y_i - y_i^*\right|$

The resulting maximum error value for two and four segment approximations between points on every fuel burn profile model is listed in Table 5-1. Note the dramatic reduction in error when using a four-segment approximation rather than a two-segment

Table 5-1: Maximum Linear Approximation Error (%)

		FL190	FL230	FL270	FL310	FL350	FL390	FL430
Four Segments	B737	2.42	2.24	2.29	2.71	2.43	2.38	2.78
	B757	1.94	2.31	1.92	2.14	2.07	2.08	2.50
	A320	3.26	2.97	3.01	2.56	2.39	2.21	2.18
Two Segments	B737	9.89	9.02	9.53	11.29	9.67	10.66	10.83
	B757	8.21	7.56	8.10	7.08	6.86	6.51	4.94
	A320	10.96	11.78	10.52	9.31	9.63	8.58	6.51

approximation. The coefficient values for the four segments of each fuel burn curve are provided in Appendix IV.

With the piecewise linear convex function defined and f_i defined as the fuel burn rate of aircraft i, then the objective function that minimizes the cumulative fuel burn is defined as follows:

$$\min \sum_{i \in N} f_i$$

However, this objective function alone does not define the quadratic fuel burn since this variable is optimized over a set of four piecewise linear convex relationships on each flight level. In other words, the right function $f_i = a_i v_i + b_i$ must be used in the optimization procedure. This is accomplished by introducing additional constraints and is discussed in the definition of the constraint set in the subsequent section.

5.5 Constraint Set Definition

Many considerations are used to define the set of feasible rerouting solutions. Specifically, the dynamics and performance limits of each flight are considered when defining the set of feasible kinematics values. Separation constraints are used to ensure that there is sufficient spacing among the aircraft routed onto the same flight level. Combinatorial constraints are used to define the feasible set of flight sequencing on each flight level. Finally, when the objective involves the minimization of the cumulative fuel burn, the additional constraints formulated in order to approximate the fuel burn function as a piecewise linear convex function are imposed.

5.5.1 Flight Dynamics Constraints

These constraints enforce the kinematics relationship between range, flight speed and acceleration. They are stipulated for each aircraft at the time horizon T as follows:

$$Tv_i = x_i - x_{0_i}, \forall i \in N$$
$$Ta_i = v_i - v_{0_i}, \forall i \in N$$

The range and flight speed at T are directly related by the first constraint while the flight speed at T and the command acceleration are directly related by the second constraint. Therefore, taken together, these two constraint equations relate the range at T and the command acceleration and relegate any direct definition between these two variables unnecessary and superfluous.

5.5.2 Performance Constraints

These constraints establish the performance envelope for each aircraft. They are stipulated for each aircraft as follows:

$$v_i \le C(1 - z_i^k) + v_{max_i}^k, \forall i \in N, k \in M$$
$$v_i \ge -C(1 - z_i^k) + v_{min_i}^k, \forall i \in N, k \in M$$
$$a_i \ge a_{min} = -2\,fps, \forall i \in N$$
$$a_i \le a_{max} = 2\,fps, \forall i \in N$$

The first two constraints are defined for each aircraft and flight level permutation and are used to establish the flight level-dependent minimum and maximum flight speeds for each aircraft. C is an arbitrarily large constant used to program the selective logic necessary to stipulate the correct flight speed bounds. The manner in which the logic operates can be illustrated via a simple example. If aircraft 1 is ultimately routed onto flight level 3, then $z_1^3 = 1$ and $z_1^k = 0, \forall k \in M \mid k \neq 3$. Therefore:

$$
\begin{aligned}
v_1 &\leq C(1 - z_1^3) + v_{max_1}^3 = C(1-1) + v_{max_1}^3 = v_{max_1}^3 \\
v_i &\geq -C(1 - z_1^3) + v_{min_1}^3 = -C(1-1) + v_{min_1}^3 = v_{min_1}^3 \\
v_i &\leq C(1 - z_1^k) + v_{min_1}^k = C(1-0) + v_{min_1}^k = C + v_{min_1}^k, \forall k \in M \mid k \neq 3 \\
v_i &\geq C(1 - z_1^k) + v_{min_1}^k = -C(1-0) + v_{min_1}^k = -C + v_{min_1}^k, \forall k \in M \mid k \neq 3
\end{aligned}
$$

In essence, all permutations of the flight speed constraints for all flight levels other than flight level 3 are rendered vacuous for aircraft 1. The only flight speed constraints that are ultimately enforced for aircraft 1 are the two minimum and maximum flight speed constraints corresponding to flight level 3.

The last two performance constraints are aircraft and flight level independent and reflect the limits imposed by satisfying passenger safety and comfort for commercial flights. Essentially, they reflect the notion that lateral accelerations in commercial flight should not exceed 2 feet per second [Nuic, 2000]. These values are derived from the BADA User Manual previously referenced in the discussion of the fundamental performance modeling.

5.5.3 Flight Sequencing Constraints

These constraints establish the set of feasible aircraft sequencing for each flight level. They are stipulated as follows:

$$\sum_{k\in M} z_i^k = 1, \forall i \in N$$
$$y_{ij} + y_{ji} = 1, \forall i \in N,\ \forall j \in N \mid i \neq j$$
$$y_{ii} = 0, \forall i \in N$$

The first constraint stipulates that each aircraft should be routed onto exactly one flight level. The second constraint stipulates that between each pair of aircraft *i* and *j*, aircraft *i* is in front of aircraft *j* or aircraft *j* is in front of aircraft *i*. In essence, this constraint is used to guard against the infeasible situation characterized by the routing of one aircraft both in front and behind another aircraft. The third constraint stipulates that an aircraft cannot be routed in front or behind itself.

5.5.4 Longitudinal Separation Constraints

These constraints establish the necessary longitudinal spacing among aircraft in the corridor. They are stipulated as follows:

$$x_j - x_i \geq -C(3 - z_i^k - z_j^k - y_{ij}) + s, \forall i \in N, \forall j \in N, \forall k \in M$$

As with the case of the performance constraints, the manner in which the logic operates can be illustrated via a simple example. If aircraft 1 and aircraft 2 are both ultimately routed onto flight level 3, with aircraft 2 routed ahead of aircraft 1, then $z_1^3 = 1$, $z_2^3 = 1$, $y_{12} = 1$ and $z_i^k = 0, i = 1\,\&\,2, \forall k \in M \mid k \neq 3$. Therefore:

$$x_2 - x_1 \geq -C(3 - z_1^3 - z_2^3 - y_{12}) + s = -C(3-1-1-1) + s = s$$
$$x_1 - x_2 \geq -C(3 - z_1^k - z_2^k - y_{12}) + s = -C(1-0-0-0) + s = -C + s, \forall k \in M \mid k \neq 3$$

In essence, all permutations of the separation constraints for all flight levels other than flight level 3 are rendered vacuous for the aircraft 1-aircraft 2 flight pair. The only sequencing constraint that is ultimately enforced for the aircraft 1-aircraft 2 flight pair is the one corresponding to the sequencing of flight 2 ahead of flight 1, when both flight 1 and flight 2 are ultimately routed onto flight level 3.

5.5.5 Fuel Burn Constraints

These constraints define the flight level dependent piecewise convex linear fuel burn functions that are used with the fuel burn minimization objective. They are stipulated as follows:

$$f_i \geq -C(1 - z_i^k) + a_{ij}^k v_i + b_{ij}^k, \forall i \in N, j \in \{1,2,3,4\}, \forall k \in M$$

As with the case of the longitudinal separation constraints, the manner in which the logic operates can be illustrated via a simple example. If aircraft 1 is ultimately routed onto flight level 3, then $z_1^3 = 1$ and $z_1^k = 0, \forall k \in M \mid k \neq 3$. Therefore:

$$f_1 \geq -C(1 - z_1^3) + a_{11}^3 v_1 + b_{11}^3 = -C(1-1) + a_{11}^3 v_1 + b_{11}^3 = a_{11}^3 v_1 + b_{11}^3$$
$$f_1 \geq -C(1 - z_2^3) + a_{12}^3 v_2 + b_{12}^3 = -C(1-1) + a_{12}^3 v_2 + b_{12}^3 = a_{12}^3 v_2 + b_{12}^3$$
$$f_1 \geq -C(1 - z_3^3) + a_{13}^3 v_3 + b_{13}^3 = -C(1-1) + a_{13}^3 v_3 + b_{13}^3 = a_{13}^3 v_3 + b_{13}^3$$
$$f_1 \geq -C(1 - z_4^3) + a_{14}^3 v_4 + b_{14}^3 = -C(1-1) + a_{14}^3 v_4 + b_{14}^3 = a_{14}^3 v_4 + b_{14}^3$$

Note that by forcing f_i to be the maximum value of the four piecewise linear convex functions, the correct linear fuel burn approximation is enforced. In addition, note that no constraints on any other flight levels are enforced:

$$f_i \geq -C(1-z_i^k) + a_{ij}^k v_i + b_{ij}^k = -C(1-0) + a_{ij}^k v_i + b_{ij}^k$$
$$= a_{ij}^k v_i + b_{ij}^k - C+, \forall i \in N, j \in \{1,2,3,4\}, \forall k \in M \mid k \neq 3$$

5.6 Summary Model

The resulting model is summarized here for the benefit of the reader:

$$\max \sum_{i \in N} x_i \text{ or } \min \sum_{i \in N} f_i$$
$$\textit{subject to:}$$
$$Tv_i = x_i - x_{0_i}, \forall i \in N$$
$$Ta_i = v_i - v_{0_i}, \forall i \in N$$
$$v_i \leq C(1-z_i^k) + v_{max_i}^k, \forall i \in N, k \in M$$
$$v_i \geq -C(1-z_i^k) + v_{min_i}^k, \forall i \in N, k \in M$$
$$a_i \geq a_{min} = 2\,fps, \forall i \in N$$
$$a_i \leq a_{max} = -2\,fps, \forall i \in N$$
$$\sum_{k \in M} z_i^k = 1, \forall i \in N$$
$$y_{ij} + y_{ji} = 1, \forall i \in N,\ \forall j \in N \mid i \neq j$$
$$y_{ii} = 0, \forall i \in N$$
$$x_j - x_i \geq -C(3 - z_i^k - z_j^k - y_{ij}) + s, \forall i \in N, \forall j \in N, \forall k \in M$$
$$f_i \geq -C(1-z_i^k) + a_{i1}^k v_i + b_{i1}^k, \forall i \in N \text{ (for fuel burn objective only)}$$
$$f_i \geq -C(1-z_i^k) + a_{i2}^k v_i + b_{12}^k, \forall i \in N \text{ (for fuel burn objective only)}$$
$$f_i \geq -C(1-z_i^k) + a_{i3}^k v_i + b_{i3}^k, \forall i \in N \text{ (for fuel burn objective only)}$$
$$f_i \geq -C(1-z_i^k) + a_{i4}^k v_i + b_{i4}^k, \forall i \in N \text{ (for fuel burn objective only)}$$

5.7 Analysis of Model Complexity

As previously asserted, the optimization model is $O(n^2)$ complex in the number of variables due to the number of binary decision variables y_{ij}. In addition, the optimization model is $O(n^2m)$ complex in the number of constraints due to the number of separation constraints. However, a number of the flight sequencing constraints serve to mitigate the complexity of the ensuing branch and bound search by fixing the values of certain variables. Consider that when assigning a value to the binary decision variable y_{ij} in the midst of the branch and bound search, the particularly strong constraint $y_{ij} + y_{ji} = 1, \forall i \in N, \forall j \in N \mid i \neq j$ acts to fix the associated variable y_{ji}. Thus, the result is that the search is mitigated on the order of $O(\frac{n^2 - n}{2})$. Finally, the very strong constraint $y_{ii} = 0, \forall i \in N$ acts to mitigate the search on the order of $O(n)$.

The variation in complexity as a function of a growing number of variables and constraints is listed in Table 5-2. The numbers in the table reflect the exact number of variables and constraints for a particular case and were calculate using the following equations:

$$variables = n^2 + nm + 4n$$
$$constraints = n^2m + n^2 + 2nm + 6n$$

Based on the representative sample problem sizes in Table 5.2, the problem complexity stays well within the limits of the problem sizes presently considered computationally tractable [Magnanti, 2003]. However, more germane is the question of whether these

Table 5-2: MILP Formulation Complexity Analysis						
	Number of Variables			Number of Constraints		
	Flight Levels (m)			Flight Levels (m)		
Aircraft (n)	3	5	7	3	5	7
10	170	190	210	520	760	1000
20	540	580	620	1840	2720	3600
50	2850	6300	6450	10600	15800	21000
100	10700	10900	11100	41200	61600	82000

problem samples can be solved using a reasonable amount of memory with commonly available computational hardware? This question is answered as part of the subsequent analysis.

5.8 Branch & Bound Development

With the MILP defined, the Branch & Bound procedure may now be detailed. As previously mentioned, the Branch & Bound algorithm uses an A* Search in finding the solution to the MILP. Thus, the description of an admissible heuristic is necessary and now possible given the formal problem definition.

5.8.1 The Admissible Heuristic: Using Greedy Flight Level Allocations

The condition that defines an admissible heuristic is that it never overestimates the cost to the goal state from any other state. Consider what a state represents in this problem formulation: a partial solution. Remember that during the Branch & Bound procedure, the integrality of only a subset of the integer variables is maintained in the solution of the relaxed LP representation. Therefore, at any point of the Branch & Bound search, a subset of the simulation traffic has been assigned to a discrete flight level because the integrality of its corresponding binary decision variables, y_{ij} and z_i^k, have been enforced, while the rest of the traffic has not. The following example, Figure 5-3, serves to illustrate this point. In the example, the state s represents the case where aircraft i has yet to be sequenced onto any flight level. In other words, no decision has been made on where to place aircraft i in the airspace, and thus, its corresponding flight level binary decision variables z_i^k have not been integer thus far. Suppose now that it is time to enforce the integrality of the flight level binary decision variable z_i^k, corresponding to aircraft i being placed on flight level k. Therefore, two nodes branch out from this state: One corresponding to placing aircraft i on flight level k $z_i^k = 1$ and another corresponding to aircraft i not be placed on flight level k $z_i^k = 0$.

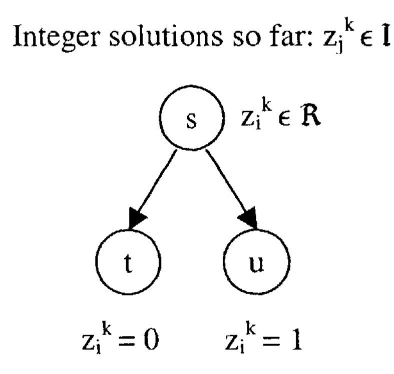


Figure 5-3: Solution State Space Illustration

In this regard, the definition of the flight level binary decision variables z_i^k is fortuitous. Consider that an admissible heuristic can be formulated by solving a partially relaxed MILP where the sequencing binary variables y_{ij} are relaxed while the integrality of the flight level binary decision variables z_i^k is maintained. This problem represents the case where the separation standards have been relaxed and do not factor in how the traffic is allocated among the flight levels. Thus, this problem is greedy in nature: which flight level does aircraft i prefer strictly from a performance standpoint? Consider that the admissibility of the heuristic is based on the Weak Duality Theorem for integer programs: a relaxed problem never overestimates its integer counterpart. Furthermore, consider that the complexity in solving this sub problem is only $O(nm)$, driven by the order of the flight level binary decision variables z_i^k. Solving this MILP sub problem at each node provides an efficient means to compute the estimated cost to the goal state and is admissible in nature. Thus, this is the admissible heuristic used in the A* Search of the solution space that is performed during the Branch & Bound procedure.

5.8.2 Sub-optimal Solutions

An additional method used to mitigate the complexity of the MILP involves truncating the search performed in the Branch & Bound procedure. This method is based on the collective experience of the optimization community with regard to many other examples of large scale, complex MILPs [Magnanti, 2003]. That is, obtaining the optimal IP solution may be an intractable problem, but a feasible IP solution might be obtained efficiently and within an acceptable degree of error. However, how can the feasible IP solution be judged to be within an acceptable degree of error if it can never be compared to the optimal IP solution? Again, the Weak Duality Theorem provides an answer.

Consider that the optimal LP value is guaranteed to never overestimate the optimal IP value: $z_{LP} \leq z_{IP}$. Furthermore, consider that any feasible IP value is guaranteed to never underestimate the optimal IP value: $z \geq z_{IP}$. Thus, the following relationships among these three values hold: $z_{LP} \leq z_{IP} \leq z$. Therefore, if a feasible IP value lies within a reasonable range of the relaxed LP value, then the feasible IP value as it relates to the optimal IP values is guaranteed to reside within this bound as well. Solving the LP provides an upper bound estimate on the degree of error between the feasible IP solution and the optimal IP solution.

Therefore, in implementing this mitigation procedure, the following questions should be addressed in the final analysis:

- How does the upper bound estimate of the error change when other parameters are changed?
- What is the minimum amount of computations (i.e. earliest truncation point) that result in an accepted degree of error?

The first question is relevant in establishing the properties of the simulation solutions. The second question is relevant in determining whether a real-time decision-aiding tool can ever be implemented that is based on the methods developed in this chapter.

-

6 Results

The results of the simulations are presented in this chapter. The results that establish the properties of the formulation are presented in Section 6.1. The utility of the A* Search in the Branch & Bound implementation is quantitatively defined and is used to establish the truncation point in the search. The variation in solution time versus problem size is presented in order to establish the formulation as tractable for practical problem sizes. Finally, the variation in the solution quality versus the optimization horizon is presented in order to demonstrate its impact on the solution.

The results for the scenarios defined in Chapter 2 are presented in Sections 6.2 – 6.5. A baseline improvement in delay and fuel burn savings is established for the single jetway scenario. The quantitative impact of instituting RVSM is presented. Finally, the effects of instituting MIT restrictions and merging traffic streams are presented.

The results of using a mixed strategy to reallocate traffic are presented in Section 6.6. This metric represents optimization with respect to minimizing overall operating costs and is compared with the baseline delay and fuel burn savings.

6.1 Establishing Model Properties

The various properties of the framework previously discussed are established in this section. These properties are explored using the delay minimization metric because the hardest set of constraints, namely the separation standard constraints, is common to each of the formulations.

6.1.1 Benchmarking A* Search Utility

The Weak Duality Theorem was used to demonstrate the fact that the LP solution could be used to establish a maximum error bound on a feasible IP solution as it relates to the optimal IP solution. This bound is referred to as the "solution gap." The solution gap versus the number of nodes visited in the problem search space for a representative problem size of 5 Flight Levels and 30 Aircraft are illustrated in Figure 6-1. Results are presented for both a Depth-First Search and A* Search in the Branch & Bound implementation. For up to 10,000 nodes in the search space, the A* Search provides a bound half as large as the Depth First Search. From the figure the fundamental drawback

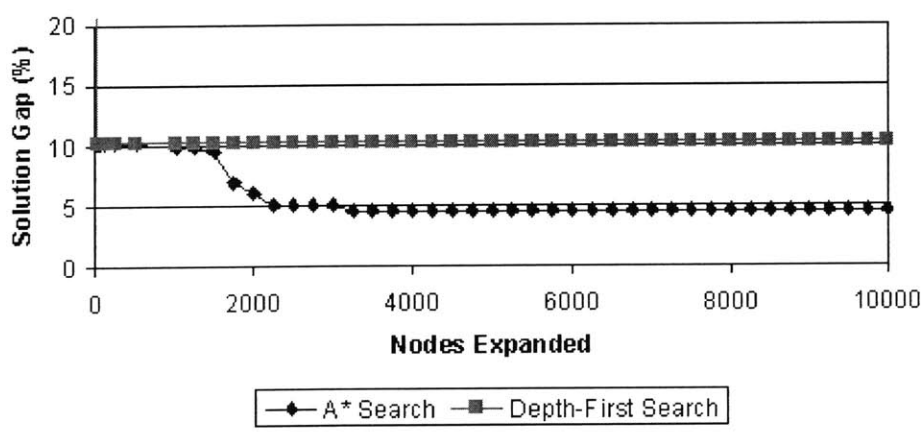


Figure 6-1: Solution Gap vs. Search Space Size
(5 Flight Levels, 30 Aircraft)

in using an Uninformed Search Method within the context of a Branch & Bound implementation is seen: the search can get "stuck" on one solution for especially large problems, as is the case here.

Based on this result, a bound on the solution search space was established. Five hundred simulations were randomly generated, each with a varying number of flight populations (25-75) and flight levels (3-5) in order to establish the following two results:

- The Branch & Bound implementation can be truncated by establishing a maximum node limit of 5,000 nodes.
- Using this node limit, the Branch & Bound implementation will return a feasible IP solution that is within 4.5% of the optimal IP solution.

6.1.2 Establishing Problem Size Limit

As discussed in Chapter 5, the problem size is driven by the order of the sequencing binary decision variables and grows with order $O(n^2)$. The variation in solution time as a function of the problem size is shown in Figure 6-2. These results were generated from 500 simulations for each data point, with the search truncated at 5,000 nodes. The linear scaling proves that solving the admissible heuristic as presented in Chapter 5 does not bog down the computation. In other words, the admissible heuristic does not grow exponentially as the problem size increases. These results were generated using a Sun Blade 100 workstation with a 500 MHz UltraSPARC-Iie CPU. The result of the problem size analysis in terms of the number of simplex iterations is illustrated in Figure 6-3.

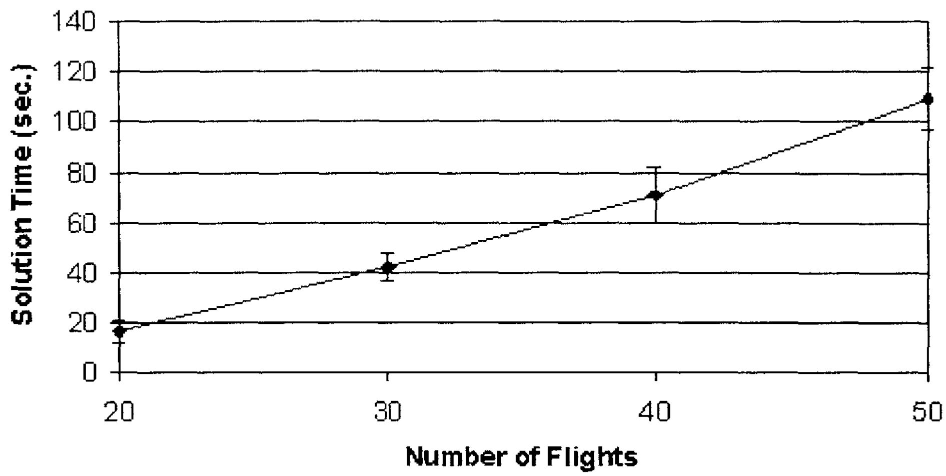


Figure 6-2: Solution Time vs. Problem Size

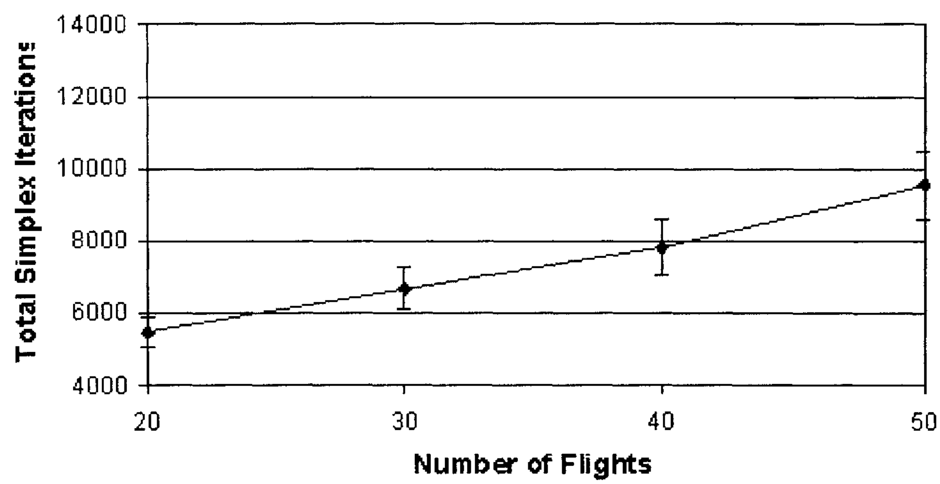


Figure 6-3: Simplex Iterations vs. Problem Size

This representation is platform-independent and can be used to estimate the computational time when using a different processor.

However, in order to establish that these computation times provide adequate results, the variation in the solution gap as a function of the problem size must be explored. The variation in the solution gap as a function of increasing problem size is illustrated in Figure 6-4. Note that the solution gap grows linearly to approximately 10% when solving problem instances of 50 aircraft. This trend suggests that sub-optimal feasible solutions of worse value are obtained as the problem size grows. However, simulations show that increasing the search space size does not result in any appreciable improvement. Thus, rather than suggesting solutions of poorer value, the trend can be explained by considering the definition of the solution gap, which is the difference between the LP value and the feasible IP value. Consider again that the LP value is the solution to a problem where the separation standards have not been enforced. In other words, the LP value represents the greedy solution. Also consider that as the problem size increases, there are a greater number of constraints imposed on the traffic, due to the

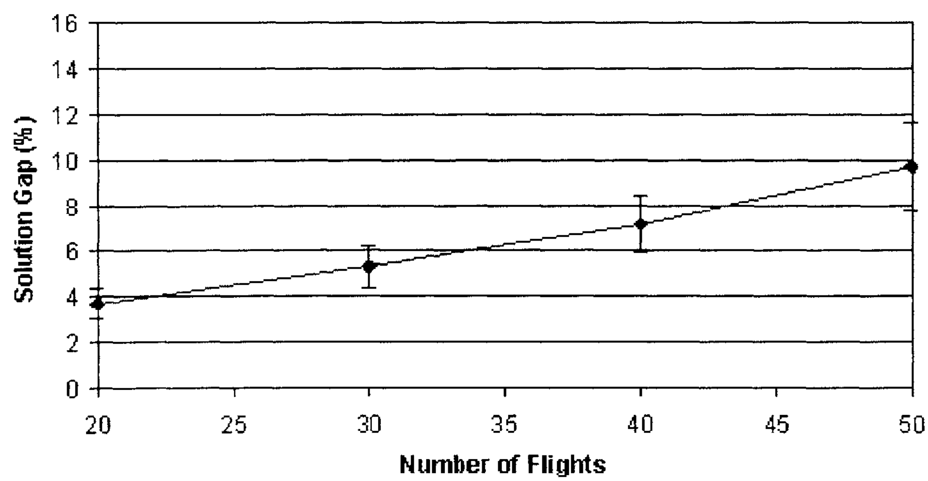


Figure 6-4: Solution Gap vs. Problem Size

separation constraints, and that it grows on the order of $O(n^2m)$. Therefore, it can be reasoned that the LP solution constitutes a bound of greater optimism as the problem size grows and that the optimal IP solution is closer to the feasible IP solution than the solution gap would indicate. Indeed, if the optimal IP solution were assumed to reside at the center of the bound, then there would only be a maximum error of 5% for problem sizes of 50 aircraft.

Given this discussion, it may be concluded that:

- The solution time increases linearly with time.
- For a maximum error of 10%, the maximum problem size is 100 aircraft.

6.1.3 Horizon Sensitivity Analysis

The arbitrary nature of the horizon parameter suggests that its impact upon the solution should be explored. The variation of the solution gap as a function of the time horizon is illustrated in Figure 6-5 for 500 simulations, using a mean separation of 5 miles. The trend suggests that feasible IP solutions of poorer value are obtained for shorter time horizons. Similar to the discussion in the previous section, the role that the LP solution plays in establishing the bound must be considered.

Tighter separation constraints result from smaller optimization horizons. That is, for the cases here, where aircraft are already spaced so closely together, time is crucial in allowing the network to develop slack. As the horizon is increased, the optimal speed of

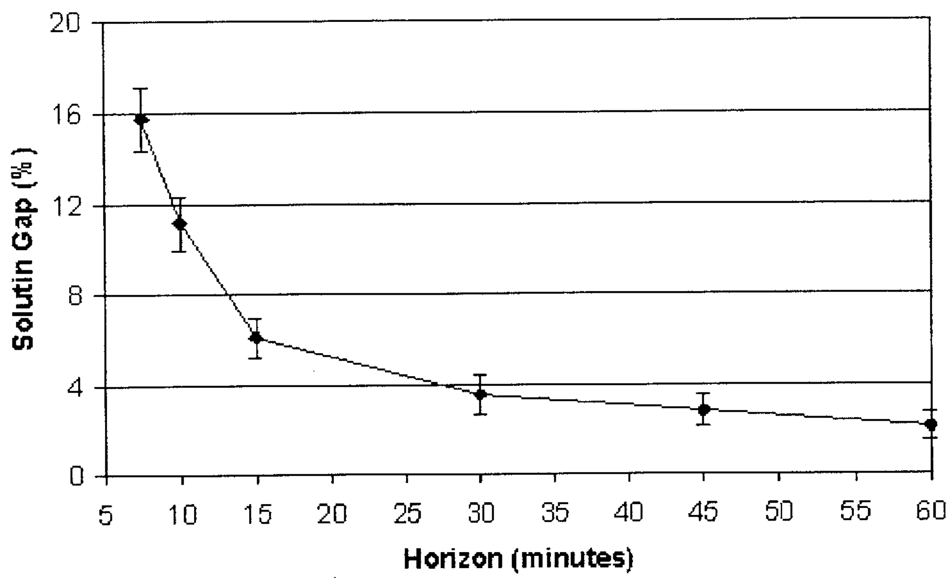


Figure 6-5: Solution Gap vs. Horizon

the aircraft distribution approaches that obtained by the LP solution. That is, after enough time, enough slack will be created in the network, so that the separation constraints become less binding. In fact, if the horizon is increased to a large enough value, then the ensuing solution will simply represent the LP solution. Thus, for the sake of the simulations, horizons of 30 minutes will be used, unless specified otherwise. The justification for this value is based upon the fact that the aircraft in the network would travel far beyond the boundaries of the corridor if a larger horizon is chosen.

6.2 Single Jetway Scenario Results

The results presented in this section are for the nominal scenarios involving a single jetway, modeled after Jetway J51, which was detailed in Chapter 2. The delay reductions and fuel burn savings resulting from the simulations are presented.

6.2.1 Delay Reduction

The distribution of delay reduction is illustrated in Figure 6-6 for 500 simulations. Note that the delay reduction is normalized with respect to a horizon of 30 minutes. The simulations were randomly generated, using 3-5 Flight Levels and 25-50 aircraft in each instance. The following results were derived from the analysis:

- An average of 8.5 minutes of delay reduction per flight is achieved in a congested network.
- For each aircraft model, an average cost savings per flight of:
    1. B737: $15938.95
    2. B757: $19,698.29
    3. MD80: $15,607.24
    4. A320: $16,661.90
    5. EMB-145: $11,907.43

It is important to note that the savings presented here and in the rest of the section reflect what can be theoretically achieved only over the horizon and should not be extrapolated over the remainder of the time each flight spends in the cruise phase. For example, it is

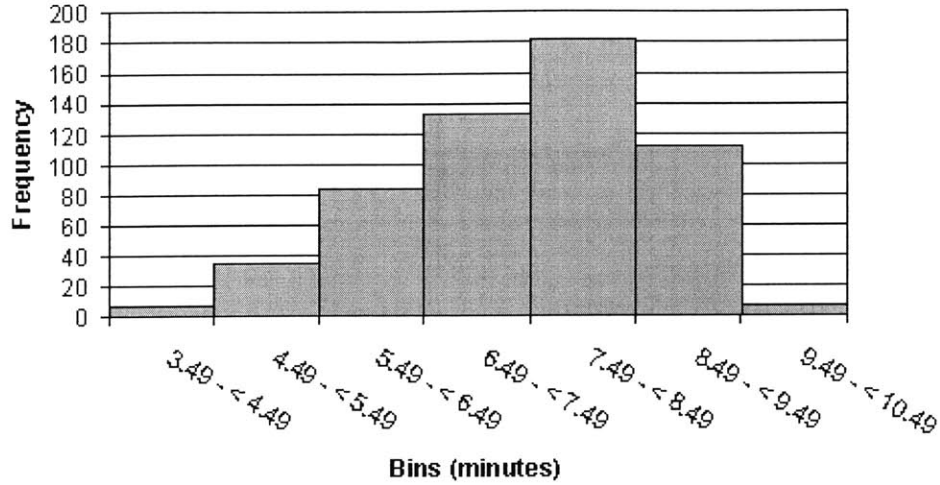


Figure 6-6: Single Jetway Delay Reduction

incorrect to assume that a flight with a cruise phase of 60 minutes will, on average, save 17 minutes of delay savings (i.e. 2 x 8.5 minutes). The reason for this is that the optimization is carried out over a finite horizon, in this case 30 minutes, over which a particular traffic configuration and set of operational constraints are defined. In a sense, the optimization tries to "push" the flow as much as possible over the given time frame. Thus, the horizon is seen as just a convenient benchmark for which the value of the optimization can be assessed. This argument holds for all of the results presented in this section.

In the future, a more rigorous approach may entail applying this optimization methodology over a simulation window so that the total potential savings can be gauged for all flights within the window.

6.2.2 Fuel Burn Savings

The distribution of fuel burn rate savings is illustrated in Figure 6-7 using the same 500 simulations. As with the delay reduction, the fuel burn rates can be normalized over the optimization horizon to yield the amount of net fuel savings. The following results were derived from the analysis:

- An average fuel burn rate reduction of 16.47 kg/min per flight is achieved in a congested network.
- An average savings of 159.4 gallons of fuel, resulting in $135.50 of cost savings, is achieved per flight.

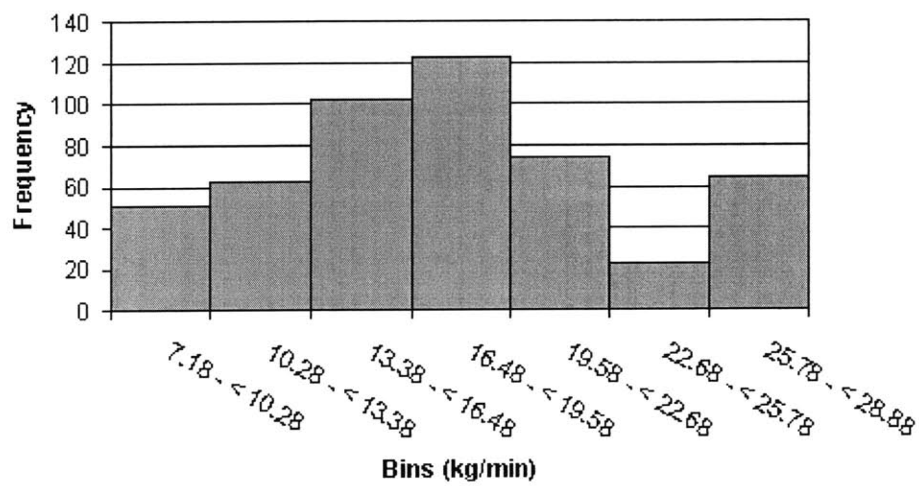


Figure 6-7: Single Jetway Fuel Burn Savings

6.3 RVSM Scenario Results

The results of introducing a new flight level in a congested network are presented below. The same 500 randomly generated simulations from the previous section were used as the basis for this analysis. The delay reductions and fuel burn savings resulting from the simulations are presented.

6.3.1 Delay Reduction

The distribution of improvement in delay savings is illustrated in Figure 6-8 for the case when RVSM is instituted. A substantial increase in the average delay savings is illustrated in the figure. This result agrees with the intuition that adding an additional

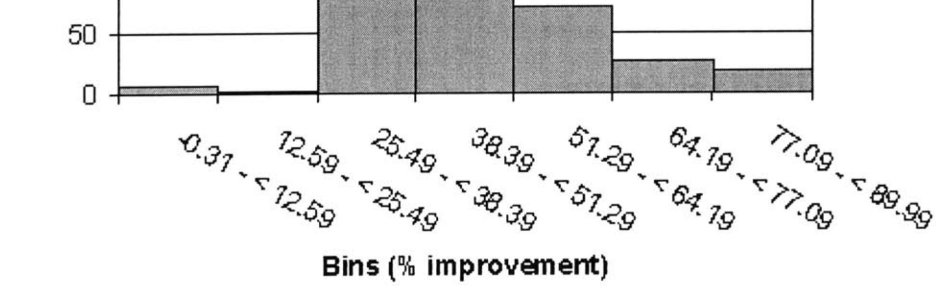


Figure 6-8: RVSM Delay Reduction

flight level can be an effective method of reducing corridor congestion. The following results were derived from the analysis:

- An average improvement of 45% over the default delay optimization case.
- An average of 12.3 minutes of delay reduction per flight is achieved in a congested network.

6.3.2 Fuel Burn Savings

The distribution of improvement in fuel burn rate savings when instituting RVSM over the default fuel burn optimization case is illustrated in Figure 6-9. From the figure, it is seen that no net fuel burn savings are realized by instituting RVSM. This result agrees with the expectation based on the performance analysis in Chapter 2. Namely, the fuel

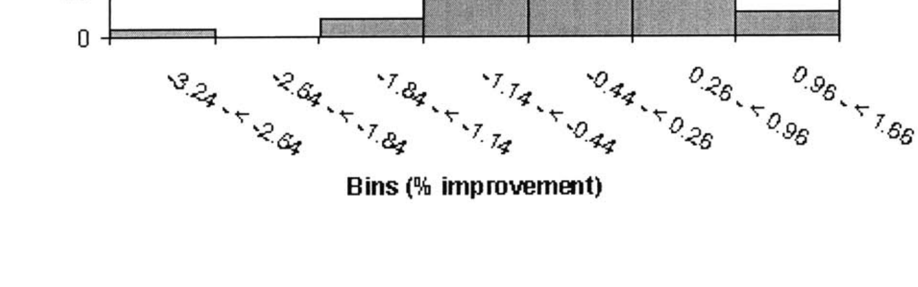


Figure 6-9: RVSM Fuel Burn Rate Reduction

burn optimization speed is well below the maximum aircraft speed, and thus, the aircraft do not need to be accelerated a great deal, as is the case in minimizing the delay. Therefore, the slack in the network is not a driving factor in reconfiguring the jetway traffic. This intuition is supported by the results in the figure.

6.4 MIT Restriction Scenario

The effect of imposing an MIT restriction upon the traffic in a congested network was also investigating. The same 500 randomly generated simulations from the previous sections were used as the basis for this analysis. For these solutions, only MIT Restrictions of 10 miles resulted in a feasible solution. Thus, for higher MIT restrictions, the network could not accommodate the traffic while ensuring enough separation among all aircraft. The delay reductions and fuel burn savings resulting from the simulations are presented below.

6.4.1 Delay Reduction

The distribution of delay reduction is presented in Figure 6-10 for 500 simulations. Intuitively, imposing an MIT restriction is expected to reduce the delay savings when compared to the default delay optimization case. This is because a greater separation must be maintained among the corridor aircraft. This intuition is supported by the results in the figure, as the amount of delay reduction is significantly lower than that achieved in the default delay reduction optimization case previously presented. The following results were derived from the analysis:

- An average of 5 minutes of delay reduction per flight is achieved in a congested network when imposing an MIT restriction of 10 miles.
- An average loss of -41% minutes of delay savings over the default delay optimization case.

6.4.1 Fuel Burn Savings

The distribution of fuel burn rate savings is illustrated in Figure 6-11 for the same 500 simulations that were generated in previous sections. Unlike the case for the delay reduction, there is no intuitive expectation for how the fuel burn rate should vary under

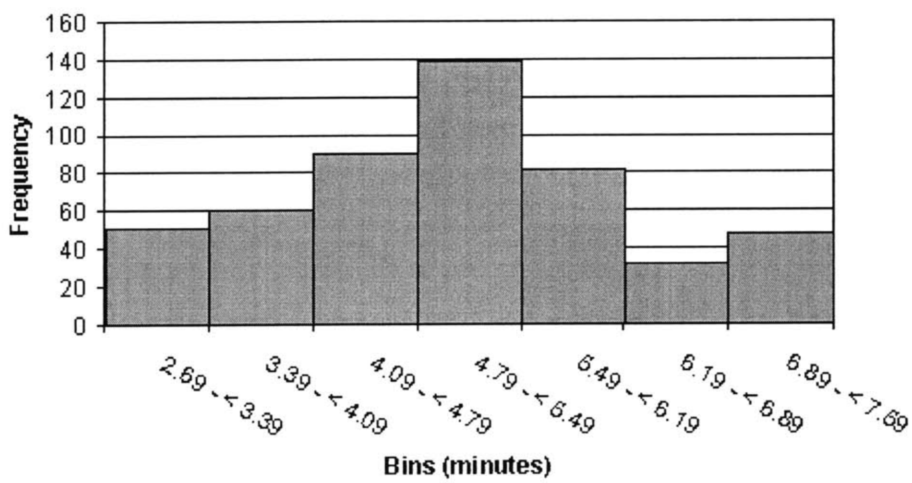


Figure 6-10: Delay Reduction - 10 MIT Restriction

such conditions. The figure indicates that a severe fuel burn penalty is imposed upon the aircraft. The following results were derived from the analysis:

- An average fuel burn rate reduction of 8.63 kg/min per flight is achieved in a congested network when imposing an MIT restriction of 10 miles.
- An average loss of -48% kg/min of fuel burn rate savings over the default fuel burn optimization case.

6.5 Merging Traffic Scenario

The results of merging climbing traffic into the overhead En Route traffic are presented in this section. In contrast with the previous scenarios, the main interest in this scenario is qualitative: can the decision model successfully merge traffic? For this analysis, 500

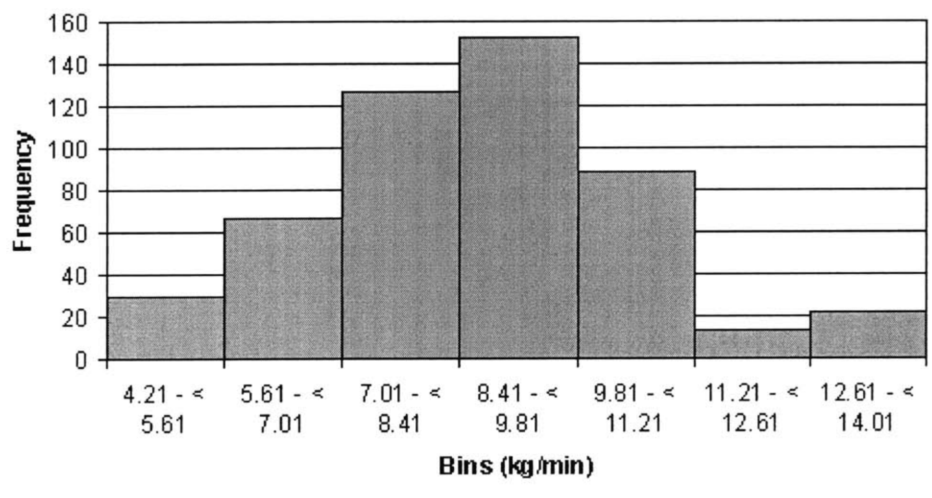


Figure 6-11: Fuel Burn Rate Reduction – 10 MIT Restriction

simulations were randomly generated, using 3-5 Flight Levels and 25-50 aircraft in each instance.

The model was able to merge the traffic successfully in every simulation. The distribution of delay reduction is illustrated in Figure 6-12. Note the slight shift in the histogram when compared to the baseline case in Figure 6-6. This shift occurs because without use of the decision logic, the climbing traffic would have to wait for "gaps" to occur in the overhead traffic before it can be merged onto the higher flight levels. The following results were derived from the analysis:

- The decision logic can be used to merge traffic streams safely.

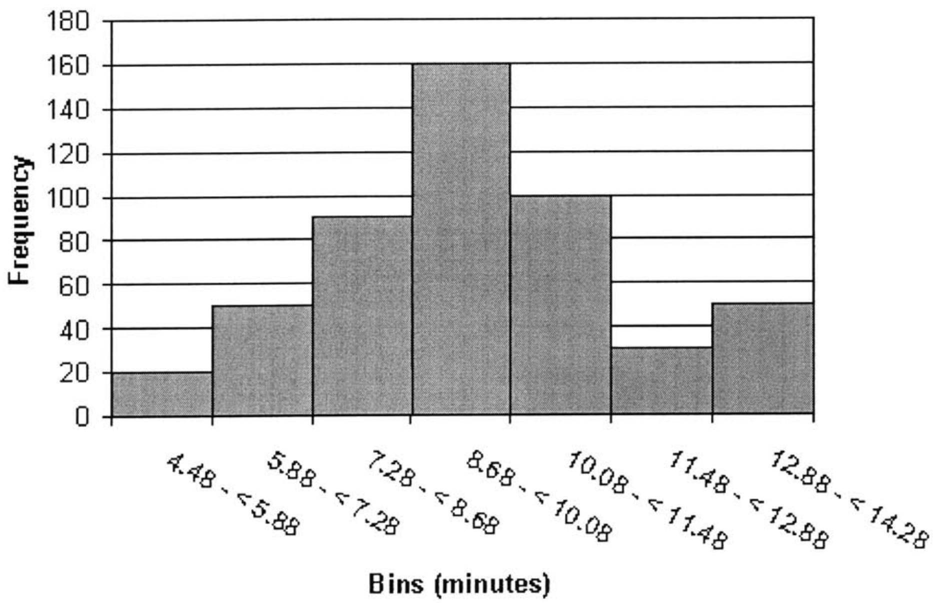


Figure 6-12: Merging Traffic Delay Reduction

- An average of 9.5 minutes of delay reduction per flight is achieved in a congested network when merging traffic streams.

6.6 Mixed Strategy Results

The results of using a mixed objective function metric to redistribute the corridor traffic are presented in this section. This corresponds to an objective that minimizes overall operating costs. The same 500 randomly generated simulations from the previous sections were used as the basis for this analysis. The distribution of cost savings is illustrated in Figure 6-13. Note that the histogram illustrates that these costs are about a third of the values obtained from simply using delay reduction as the objective function metric. This observation is expected when considering the quadratic increase in fuel burn as a function of flight speed. Thus, a trade-off point between delay reduction and fuel burn savings can be identified. The following results were derived from the analysis:

- An average cost savings of $5,552.98 per flight.
- A trade-off point that corresponds to a value that is 65% of the baseline delay reduction and 57% of the baseline fuel burn savings.
- An average of 5.5 minutes of delay reduction per flight.
- An average of 9.46 kg/min of fuel burn rate savings per flight.

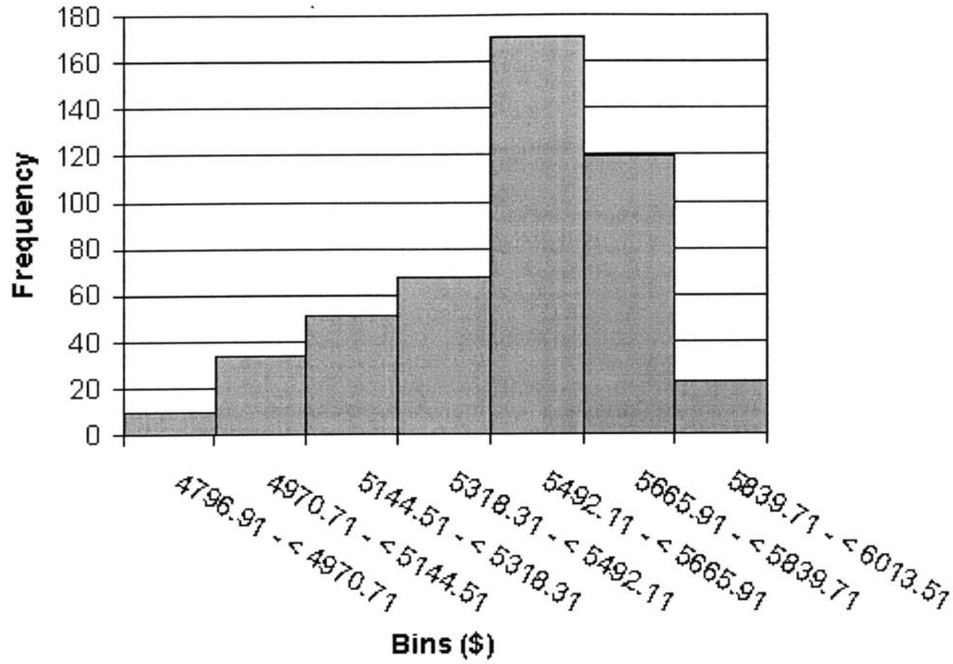


Figure 6-13: Cost Savings for Mixed Metric

7 Conclusions and Recommendations

A methodology for alleviating congested En Route airspace was developed in this thesis. The proposed research approach is novel because the problem is developed within a tactical context, thereby capturing the short-term dynamics that cause congestion in a corridor setting. The methodology resulted in the development of a decision-aiding tool which used an A* Search-Based Branch & Bound procedure to solve a MILP formulation of the problem. Decomposing the integer variables in the MILP formulation mitigated the potential barrier posed by the computational complexity of the methodology. Running multiple simulations of a portion of the Northeast Corridor over a variety of different scenarios validated the approach. A summary of the key conclusions is presented here.

7.1 Conclusions

The following results were achieved for an optimization horizon of 30 minutes:

- The Branch & Bound implementation has been shown to provide a feasible IP solution that is within 4.5% of the optimal IP solution. Furthermore, the A* Search heuristic is an efficient method that allows large problem instances consisting of approximately 100 aircraft to be solved within a reasonable time. Therefore, the MILP presented in this thesis can be further developed as a basis for a real-time decision-aiding tool for reallocating traffic in NAS corridors.

- When delay is used as the objective function metric, an average of 8.5 minutes of delay reduction per flight is achieved in a congested network. For each aircraft modeled, this value results in the following cost savings per flight:
    - B737: $15,938.95
    - B757: $19,698.29
    - MD80: $15,607.24
    - A320: $16,661.90
    - EMB-145: $11,907.43

- When fuel burn is used as the objective function metric, an average fuel burn rate reduction of 16.47 kg/min per flight is achieved in a congested network. This value results in an average fuel savings of 159.4 gallons of fuel per flight, resulting in $135.50 of cost savings per flight.

- When an RVSM program is instituted and delay is used as the objective function metric, an average of 12.3 minutes of delay reduction per flight is achieved in a congested corridor. This value represents an average improvement of 45% over the default delay optimization case.

- When an MIT restriction of 10 miles is imposed and delay is used as the objective function metric, an average of 5 minutes of delay reduction per flight is achieved in a congested corridor. This value represents an average loss of -41% minutes of delay savings over the default case when no MIT restrictions have been imposed.

- When fuel burn is used as the objective function metric, an average fuel burn rate reduction of 8.63 kg/min per flight is achieved in a congested corridor. This value represents an average loss of -48% kg/min of fuel burn rate savings over the default fuel burn optimization case.

- When a mixed objective function metric is used, an average cost savings per flight of $5,552.98 is achieved. This value represents the trade-off point between delay reduction and fuel burn savings. This value corresponds to 65% of the baseline delay reduction and 57% of the baseline fuel burn savings. An average of 5.5 minutes of delay reduction per flight is achieved in a congested corridor. In addition, an average of 9.46 kg/min of fuel burn savings per flight is achieved in a congested corridor.

- The decision logic can be used to merge traffic streams safely.

7.2 Recommendations

The work presented here represents only the first step in researching the issue of air traffic congestion within a tactical context. As mentioned within this thesis, most, if not all of the previous approaches neglected the short-term dynamics of the system either because the computational complexity was believed intractable or because previous researchers believed the problem to be strategic in nature. Given the advances in the field of Applied Mathematics as well as the computational power of computer processors today, the implication that can be taken away from this thesis is that future work tailored to mitigating En Route traffic congestion should seek to model the tactical aspect of the problem as well as the strategic aspect. A summary of the recommendations leveraged from the experience gained during the course of this research is presented here.

- As the flight level histograms in Appendix I illustrate, more data samples of the corridor traffic are needed. Thus, more ETMS data must be acquired, filtered and sampled in order to guarantee that the models used to create the simulations are faithful representations of the traffic distributions in the NAS.

- A dialogue with En Route controllers should be established in order to ascertain how the NAS becomes congested. The results from the ETMS data filtering demonstrate that a minimum of 60 miles of separation exists between aircraft for each of the five days sampled. Either the data is incomplete or a better understanding of the NAS constraints is in order.

- A stronger constraint set formulation would allow the implementation to be used for far larger problem instances. Attention should be specifically paid to reformulating the separation constraints, which act to reduce the efficacy of the LP that is used as the criterion to bound the Branch & Bound search. Further research into the area of scheduling problem formulations could potentially yield a stronger formulation.

- The MILP should be adapted to handle a three-dimensional airspace configuration. This would increase the potential efficacy of the methodology that has been developed.

Bibliography

[ATA, 1999] Anon., *Approaching Gridlock: Air Traffic Control Delays*, Air Transportation Association, October 14, 1999.

[ATC, 2000] Anon., *Washington Center (ZDC)*, Air Traffic Cafe, http://airtrafficcafe.com/facility/ZDC.shtml, 2000.

[Bertsimas and Tsitsiklis, 1997] Betrsimas, D., and Tsitsiklis, J.N., *Introduction to Linear Optimization*, Athena Scientific, 1997.

[Boeing, 2001] Anon. *Air Traffic Fact Sheet*, The Boeing Company, Air Traffic Management Group, March, 2001.

[DOE, 2003] Anon., *Petroleum Status Report*, Department of Energy, September, 2003.

[Evans, 2001] Evans, J.E., *Tactical Weather Decision Support to Complement "Strategic" Traffic Flow Management for Convective Weather*, Presented at the 4th USA/Europe Air Traffic Management R&D Seminar, Santa Fe, New Mexico, December 3-7, 2001.

[FAA, 2001] Anon., *Aviation System Capacity Plans*, Federal Aviation Administration, 2001.

[FAA, 2002] Anon., *IFR Enroute High Altitude Chart – U.S., H-5 & H6*, Federal Aviation Administration, February 21, 2002.

[FAA, 2003] Anon., *Terminal and En Route Airspace Procedures for Handling Airspace Matters*, Federal Aviation Administration, FAA Order No. 7400.2E, December 7, 2000. Amended May 15, 2003.

[Harrison and Rehmann, 2000] Harrison, M., and Rehmann, J., *Evaluation and Demonstration of Operational Evolution Plan Improvements*, Federal Aviation Administration, 2000.

[Kostiuk, et al., 1998] Kostiuk, P.F., Gaier, E., Long, D., *The Economic Impact of Air Traffic Congestion*, Logistics Management Institute, Report No. IR811S1, April, 1998.

[Magnanti, 2003] Magnanti, T.L., *15.094 Class Notes*, MIT, 2003.

[Melconian, 2001] Melconian, T., *Effects of Increased Nonstop Routing on Airline Cost and Profit*, Master's Thesis, September, 2001.

[Nuic, 2000] Nuic, A., *User Manual for the Base of Aircraft Data (BADA), Revision 3.3*, Eurocontrol Experimental Center Report No. 20/00, December, 2000.

[Pallottino, et al., 2002] Pallottino, L., Feron, E., Bicchi, E., *Conflict Resolution Problems for Air Traffic Management Systems Solved with Mixed Integer Programming*, IEEE Intelligent Transportation Systems, Vol. 3, No. 1, March 2002, pp. 3-11.

[Thedford, et al., 1999] Thedford, W., Vivona, R.A., Hodgon, C., *En-Route Constrained Airspace Concept Definition – Final Report*, System Resources Corporation, NASA Contract No. RTO-07: NAS2-98005, September, 1999.

[Vanderbei, 2000] Vanderbei, R. J., *Case Studies in Trajectory Optimization: Trains, Planes, and Other Pastimes*, ONR Report No. ORFE-00-3, 2000.

[USCB, 2003] Anon., *Table SUB-EST2002-01 - City and Town Population Estimates: April 1, 2000 to July 1, 2002*, United States Census Bureau, Population Division, July 10, 2003.

Appendix I Flight Speed Histograms

I.1 Jetway J42

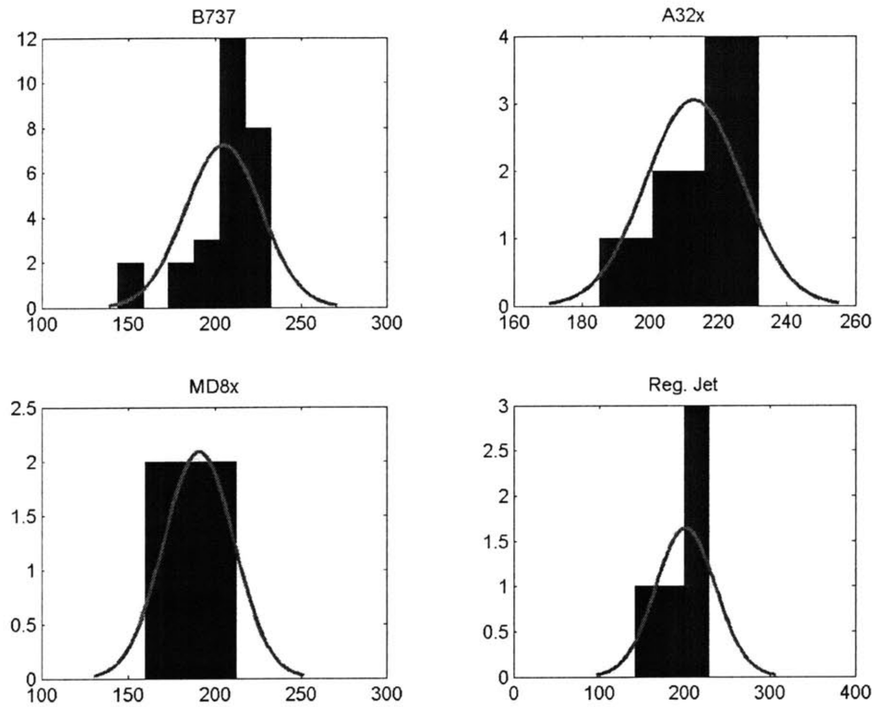


Figure I-1: J42 Flight Level 190 TAS Distribution (kts)

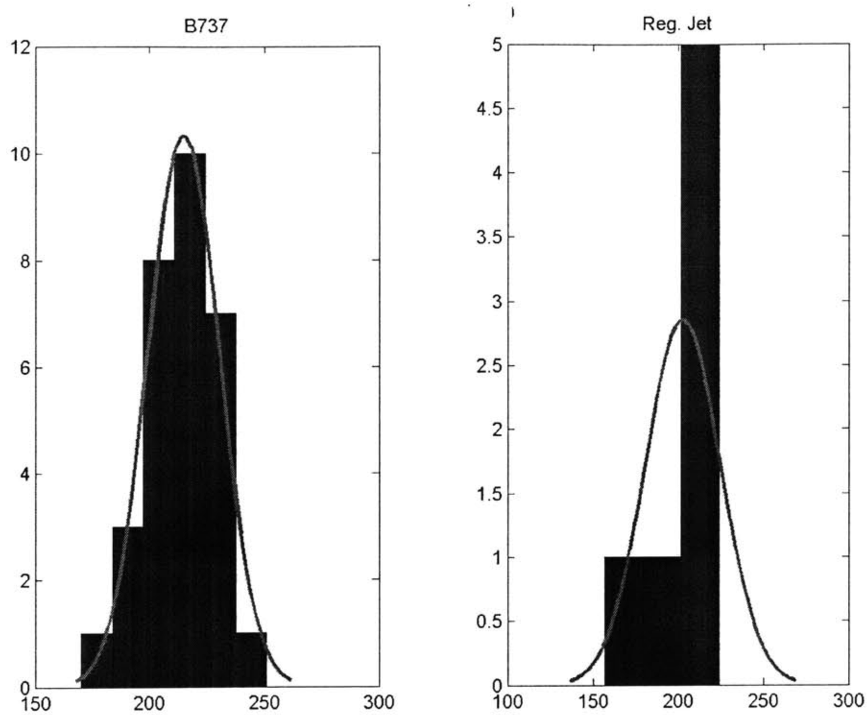


Figure I-2: J42 Flight Level 210 TAS Distribution (kts)

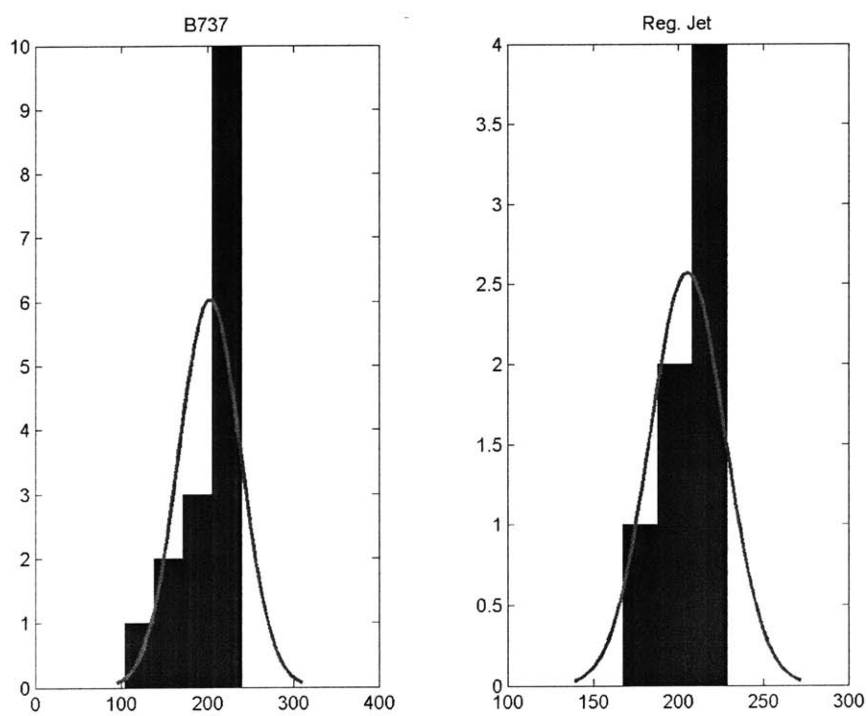


Figure I-3: J42 Flight Level 230 TAS Distribution (kts)

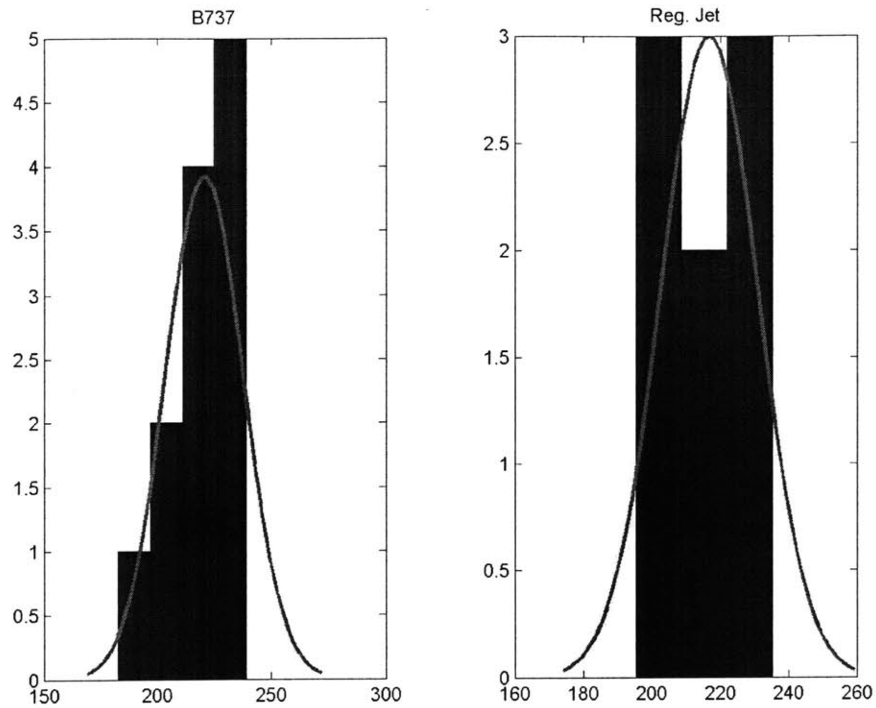


Figure I-4: J42 Flight Level 250 TAS Distribution (kts)

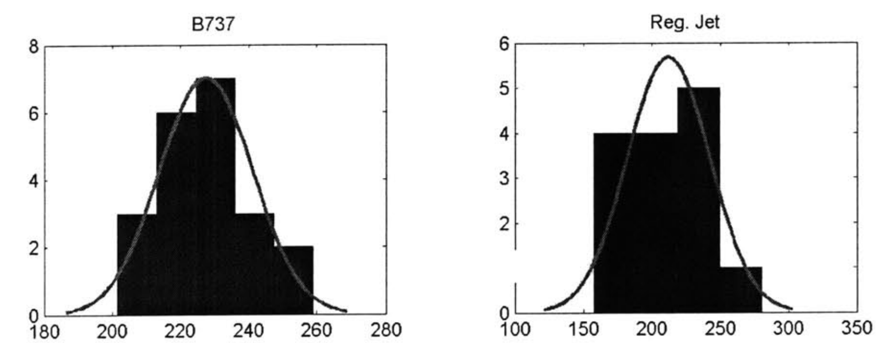


Figure I-5: J42 Flight Level 270 TAS Distribution (kts)

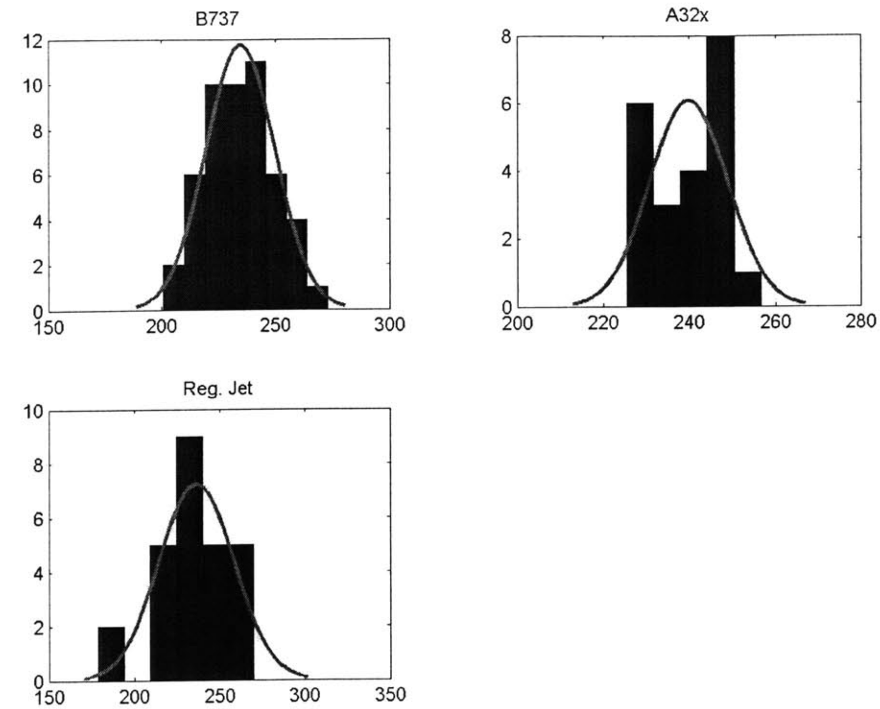


Figure I-6: J42 Flight Level 290 TAS Distribution (kts)

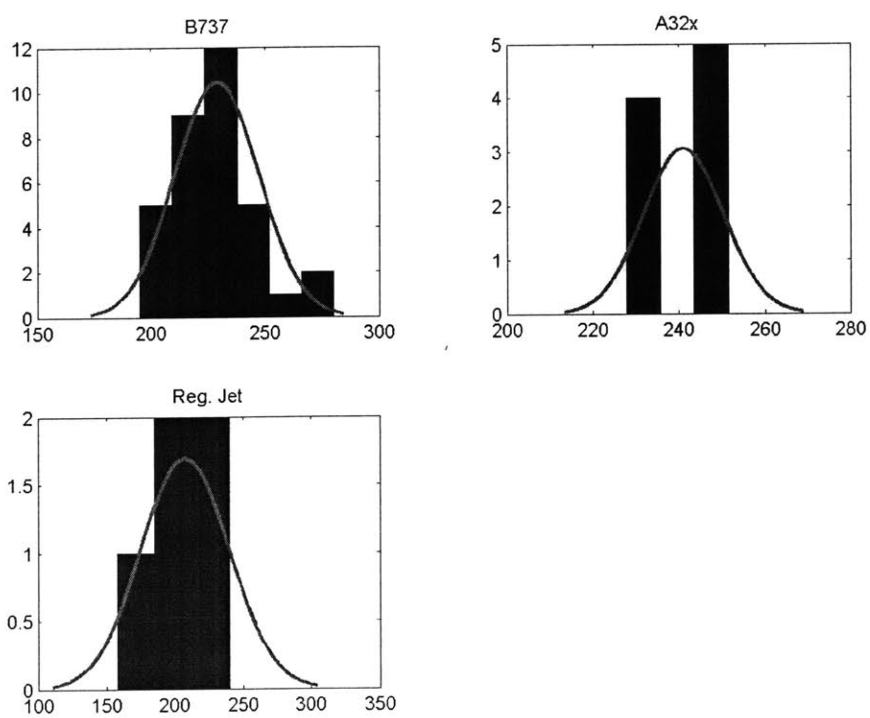


Figure I-7: J42 Flight Level 310 TAS Distribution (kts)

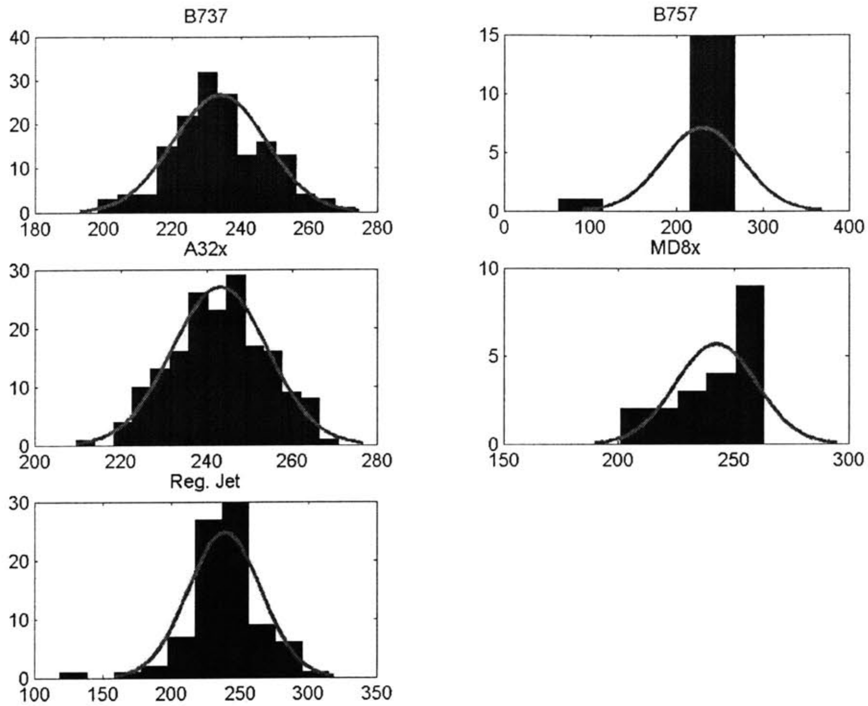


Figure I-8: J42 Flight Level 330 TAS Distribution (kts)

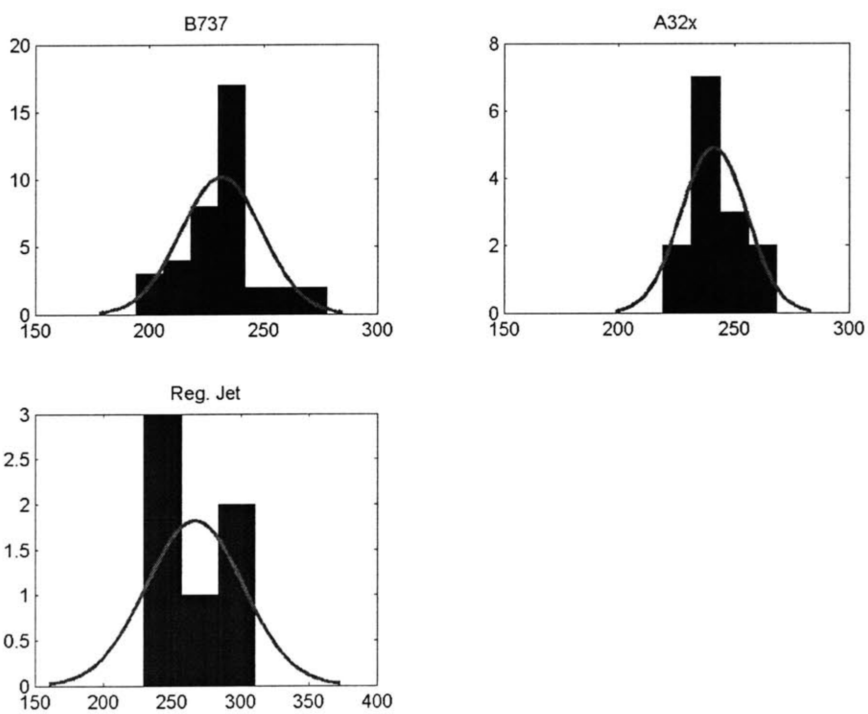


Figure I-9: J42 Flight Level 350 TAS Distribution (kts)

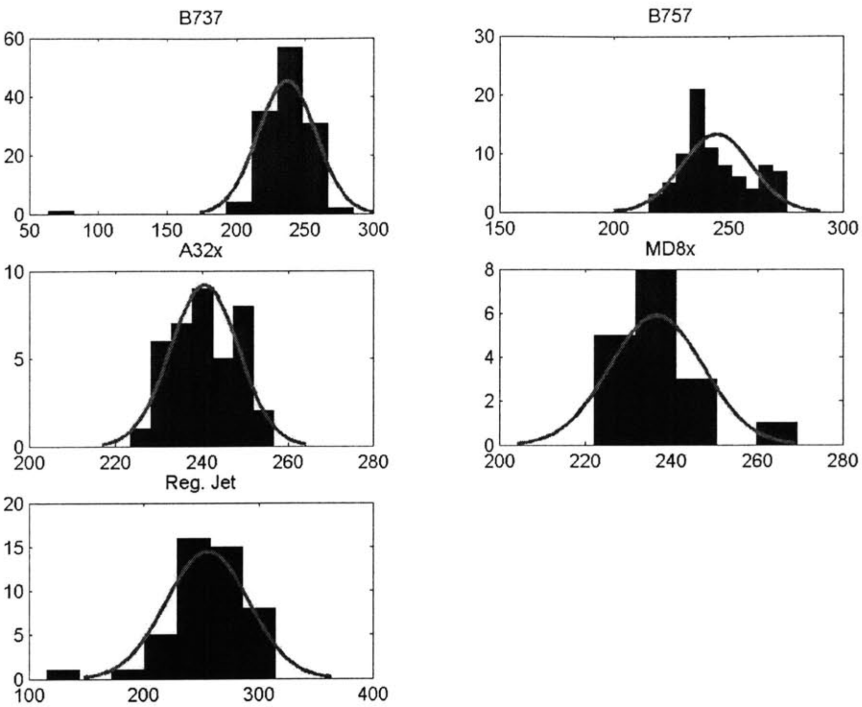


Figure I-10: J42 Flight Level 370 TAS Distribution (kts)

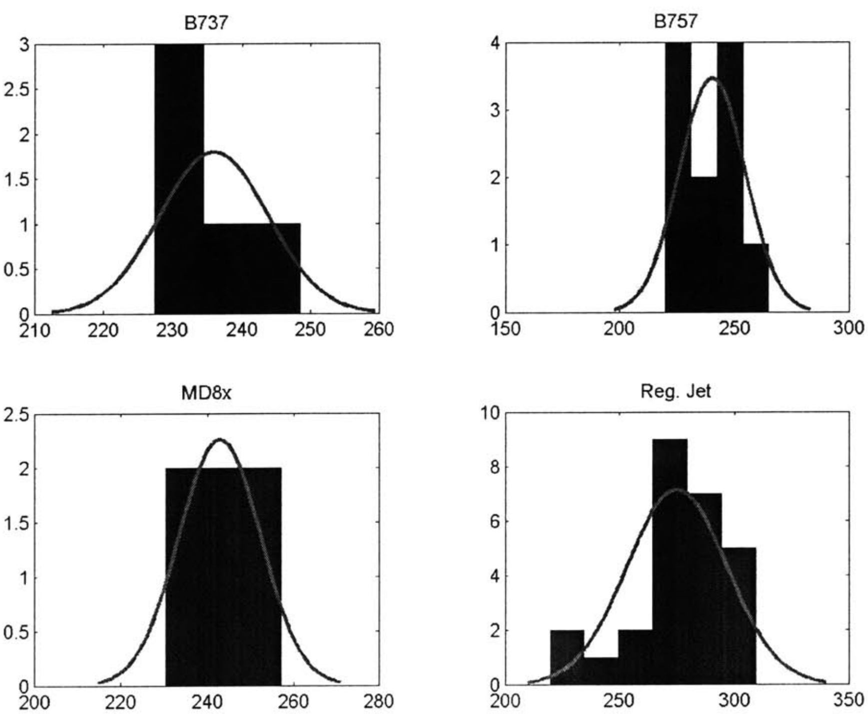


Figure I-11: J42 Flight Level 390 TAS Distribution (kts)

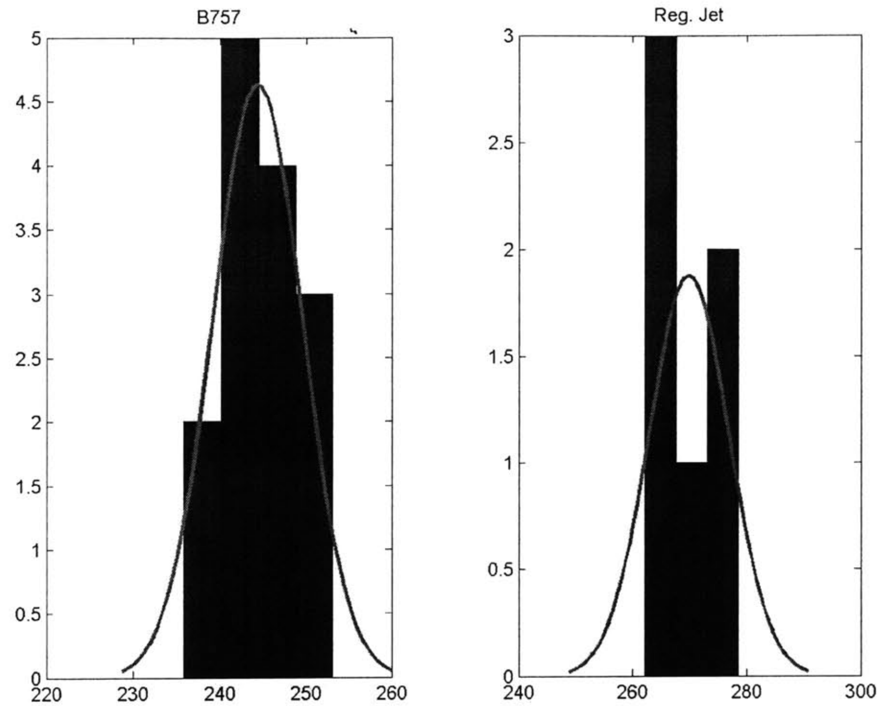


Figure I-12: J42 Flight Level 410 TAS Distribution (kts)

I.2 Jetway J191

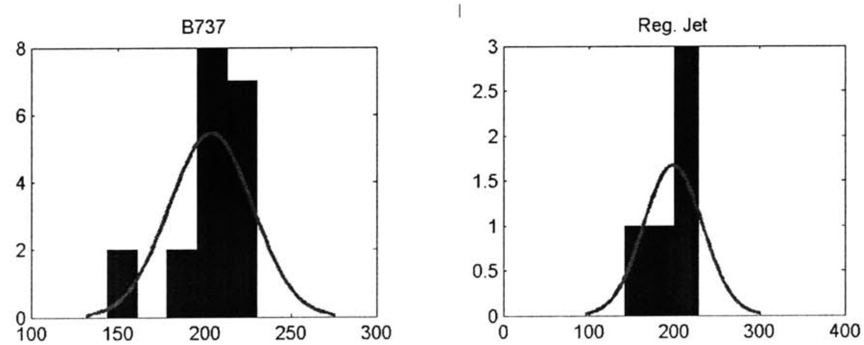


Figure I-13: J191 Flight Level 190 TAS Distribution (kts)

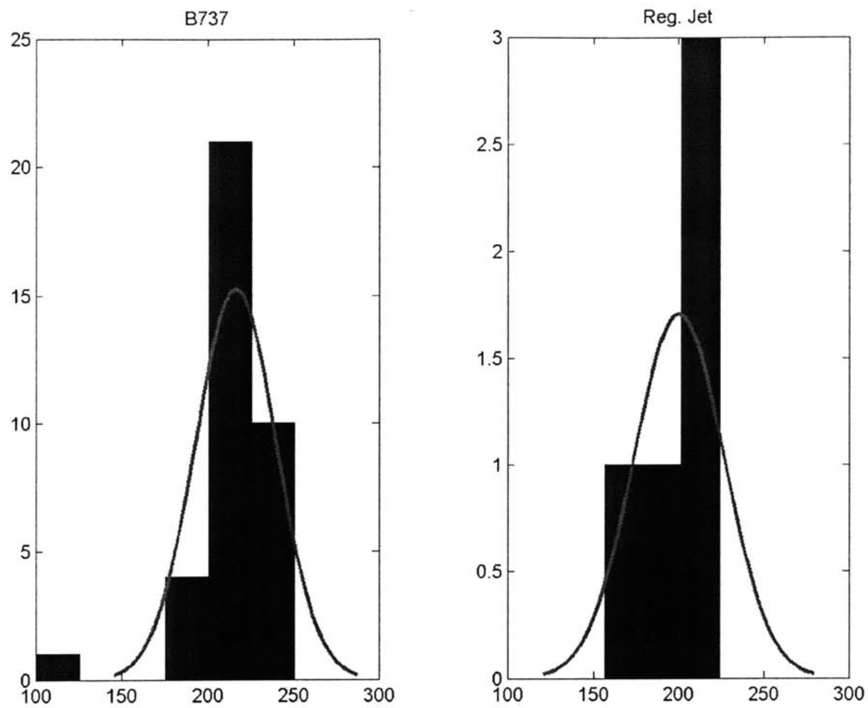


Figure I-14: J191 Flight Level 210 TAS Distribution (kts)

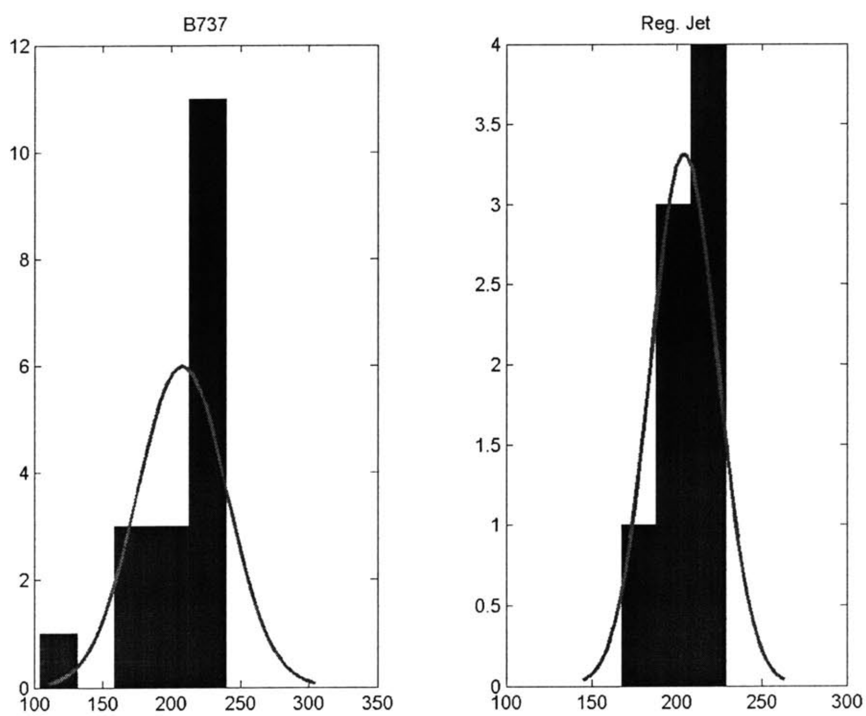


Figure I-15: J191 Flight Level 230 TAS Distribution (kts)

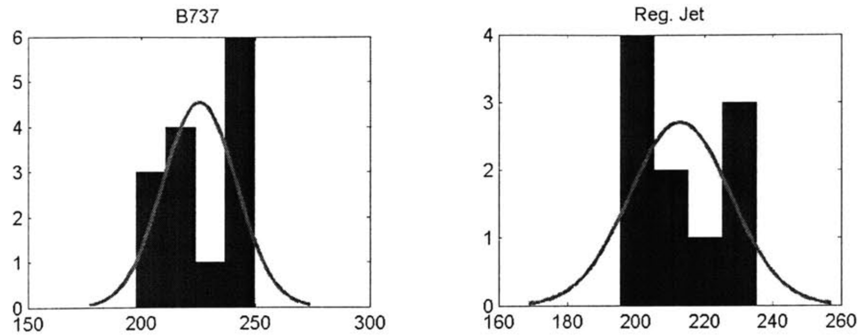


Figure I-16: J191 Flight Level 250 TAS Distribution (kts)

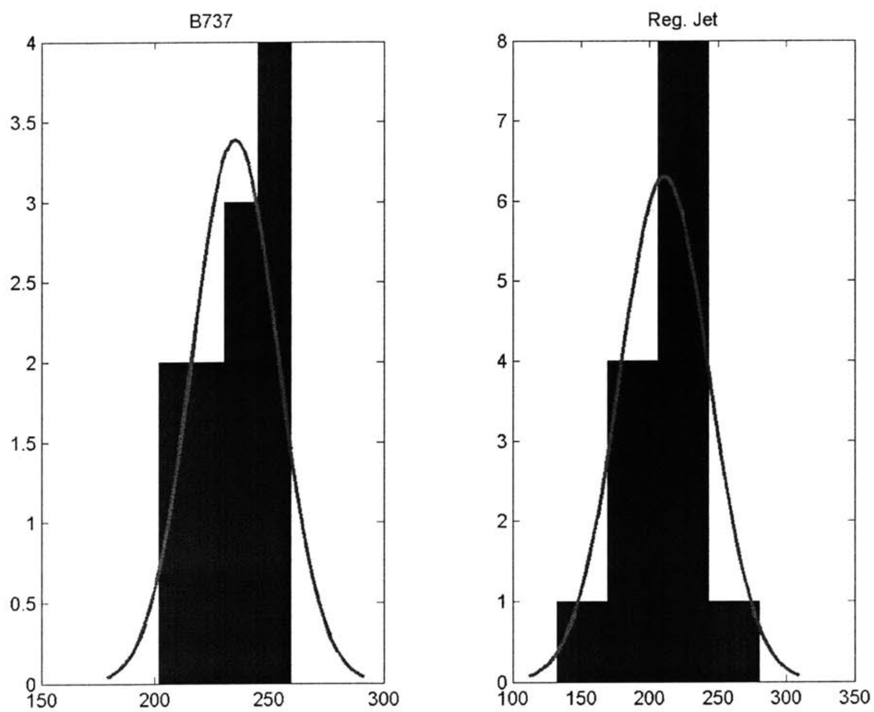


Figure I-17: J191 Flight Level 270 TAS Distribution (kts)

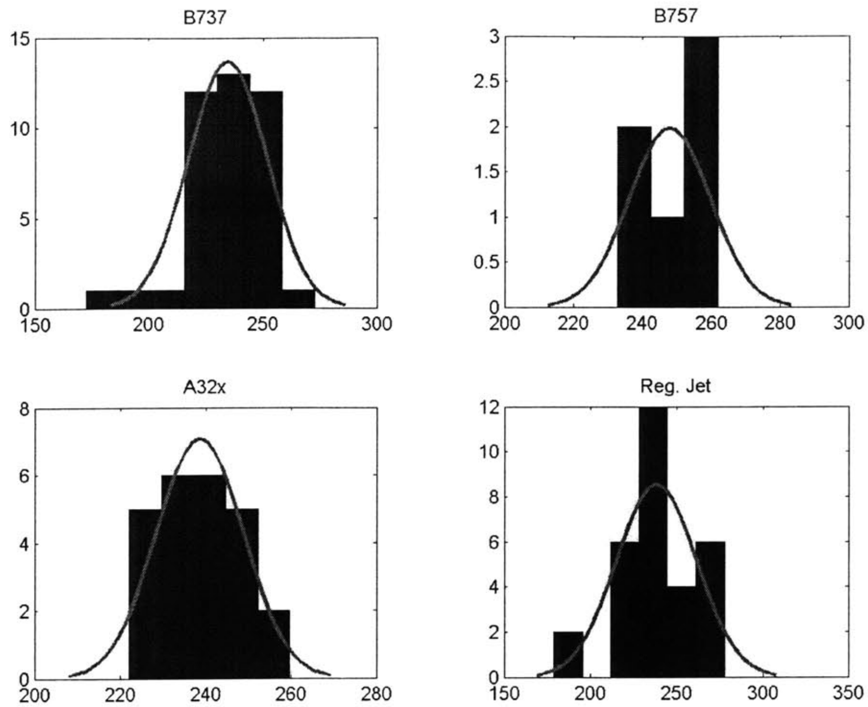


Figure I-18: J191 Flight Level 290 TAS Distribution (kts)

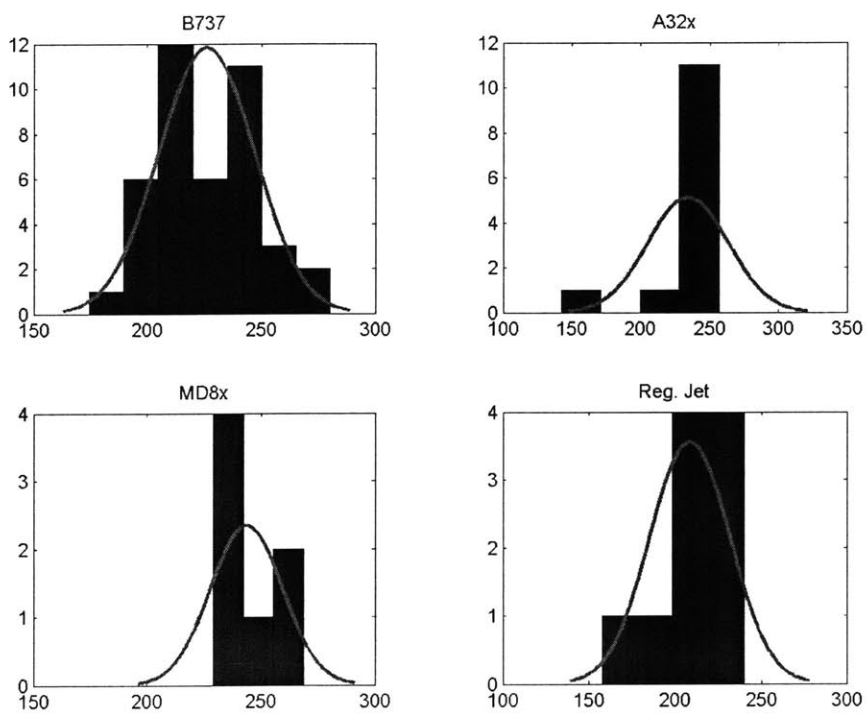


Figure I-19: J191 Flight Level 310 TAS Distribution (kts)

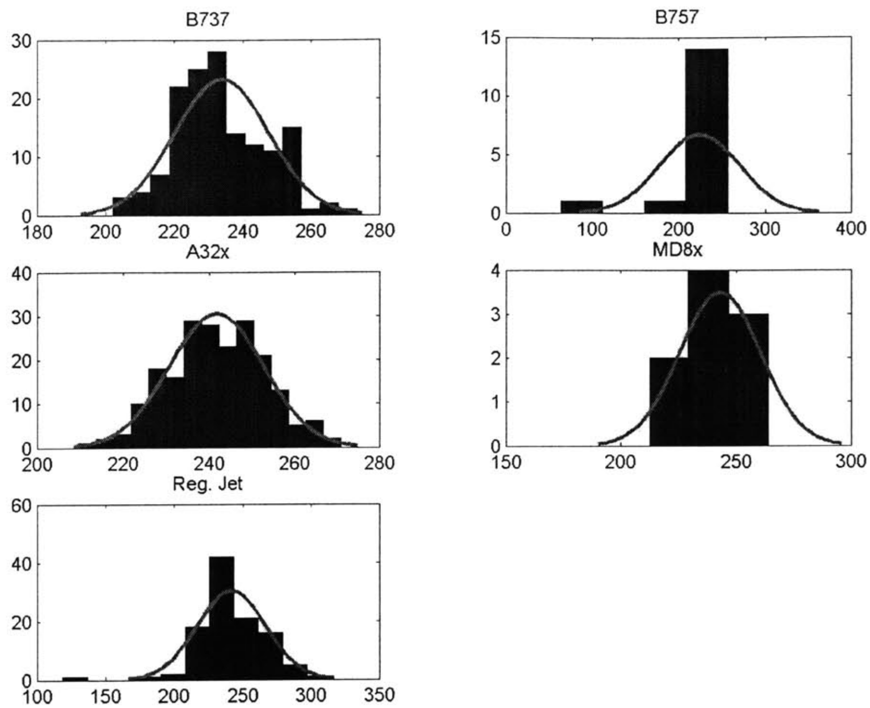


Figure I-20: J191 Flight Level 330 TAS Distribution (kts)

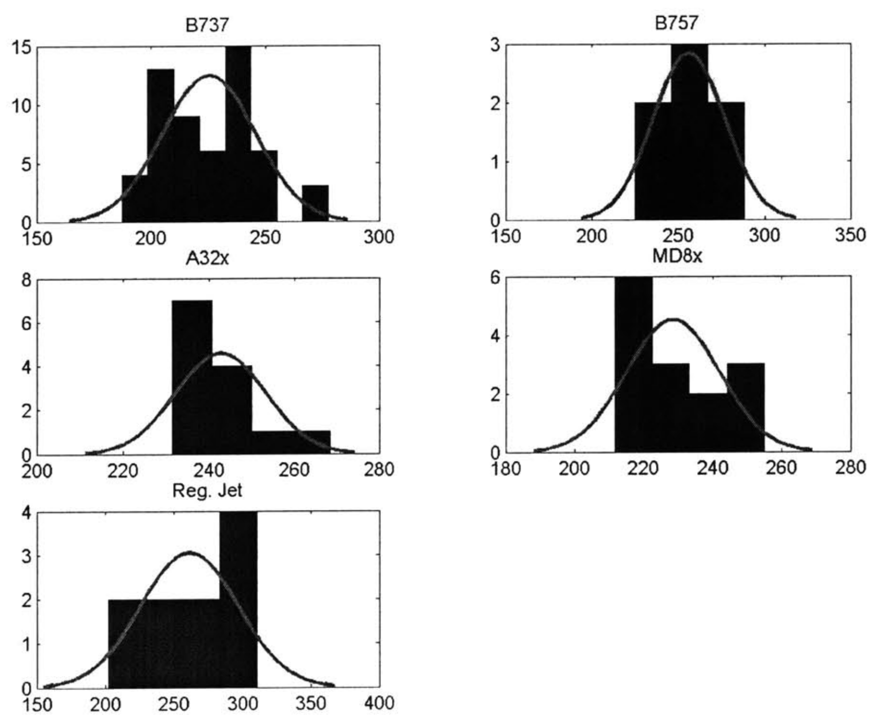


Figure I-21: J191 Flight Level 350 TAS Distribution (kts)

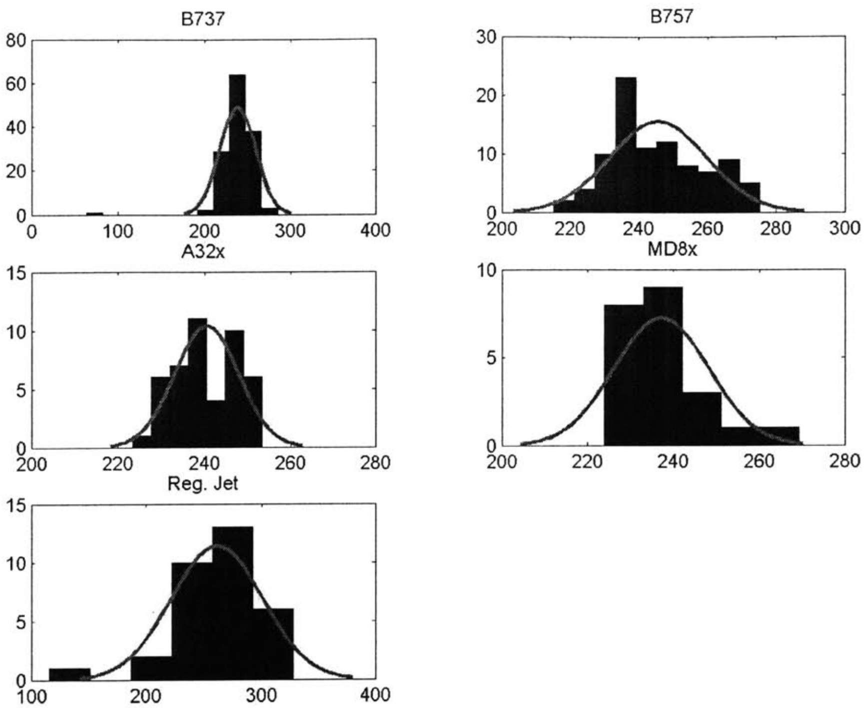


Figure I-22: J191 Flight Level 370 TAS Distribution (kts)

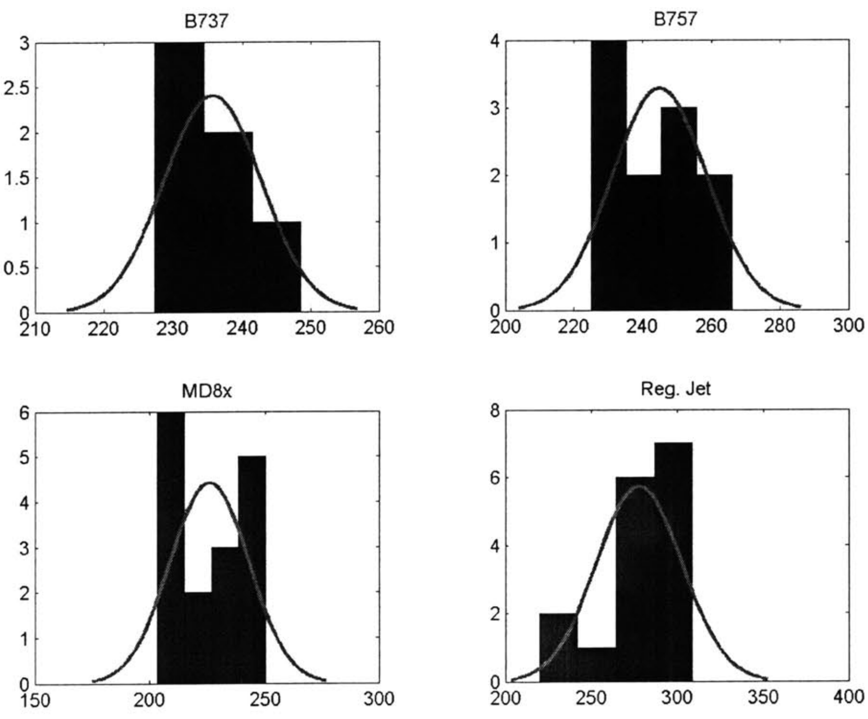


Figure I-23: J191 Flight Level 390 TAS Distribution (kts)

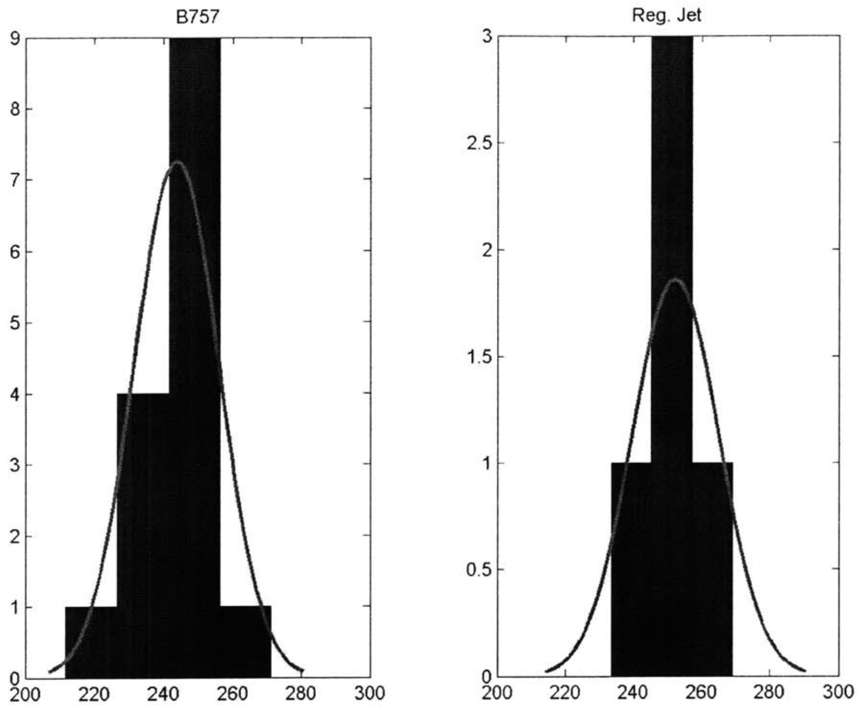


Figure I-24: J191 Flight Level 410 TAS Distribution (kts)

Appendix II Aircraft Performance Summary

II.1 Boeing 737

$h_{MO} = 45,000\,ft$

$CAS_{min} = 340\,kts$

$M_{max} = 0.82$

Table II.1: Boeing 737 Flight Envelope			
Flight Level	Min TAS	Max TAS	Min Fuel Burn TAS
ft	kts	Kts	kts
19000	181	506	290
23000	193	498	310
27000	207	490	335
29000	222	481	355
33000	239	473	380
37000	260	470	415
41000	285	470	455

II.2 Boeing 757

$h_{MO} = 42,000\,ft$

$CAS_{min} = 350\,kts$

$M_{max} = 0.86$

Table II.2: Boeing 757 Flight Envelope			
Flight Level	Min TAS	Max TAS	Min Fuel Burn TAS
ft	Kts	kts	kts
19000	206	530	285
23000	220	522	305
27000	235	513	325
29000	252	505	350
33000	271	496	375
37000	295	493	410
41000	323	493	450

II.3 Boeing MD80/MD88/MD90/717

$h_{MO} = 37{,}000\,ft$

$CAS_{min} = 340\,kts$

$M_{max} = 0.84$

Table II.3: Boeing MD80/MD88/MD90/717 Flight Envelope			
Flight Level	Min TAS	Max TAS	Min Fuel Burn TAS
ft	kts	kts	kts
19000	215	518	270
23000	230	510	287
27000	245	501	307
29000	262	493	330
33000	280	484	355
37000	304	482	385

II.4 Airbus A318/A319/A320

$h_{MO} = 39,000\, ft$

$CAS_{min} = 350\, kts$

$M_{max} = 0.82$

Table II.4: Airbus A318/A319/A320 Flight Envelope			
Flight Level	Min TAS	Max TAS	Min Fuel Burn TAS
ft	kts	kts	kts
19000	197	506	255
23000	210	498	270
27000	225	490	290
29000	241	481	315
33000	259	473	340
37000	282	470	370

II.5 Embraer EMB-145

$h_{MO} = 41{,}000\,ft$

$CAS_{min} = 335\,kts$

$M_{max} = 0.85$

Table II.5: Embraer EMB-145 Flight Envelope			
Flight Level	Min TAS	Max TAS	Min Fuel Burn TAS
Ft	kts	kts	kts
19000	188	524	263
23000	201	516	280
27000	214	507	303
29000	230	499	325
33000	246	490	347
37000	268	488	365
41000	292	488	377

Appendix III Fuel Burn Analysis Figures

III.1 Boeing 737

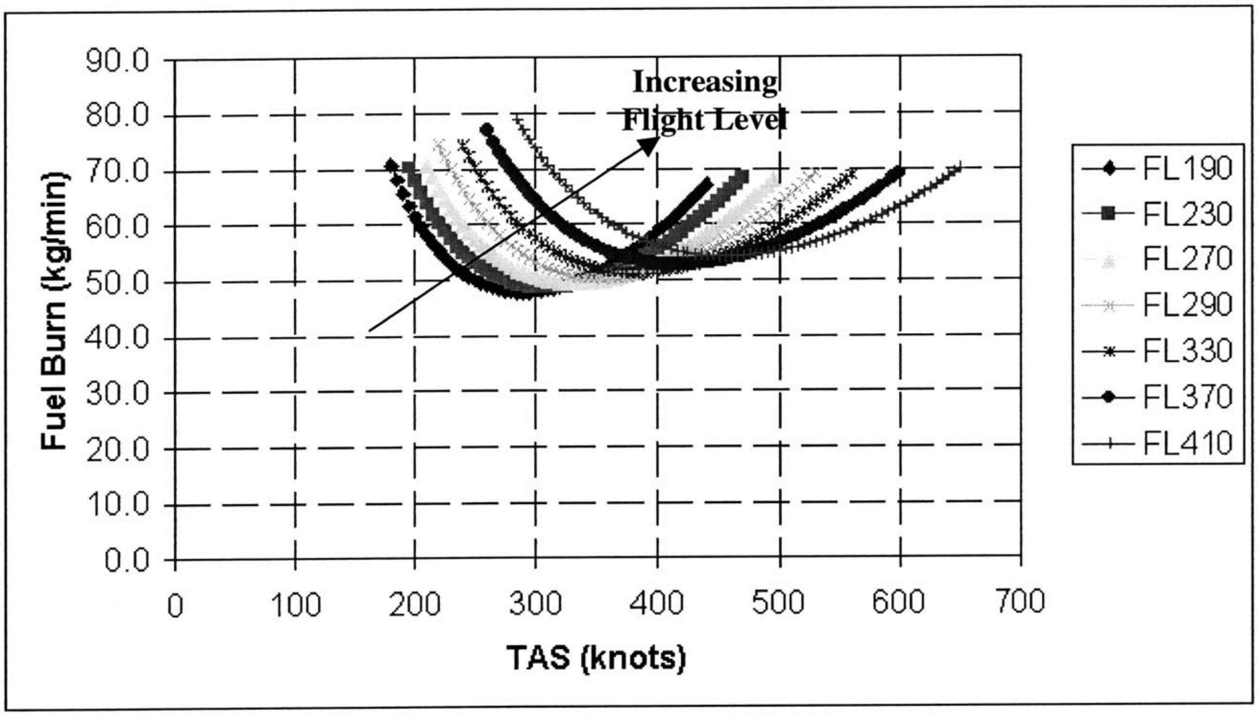


Figure III-1: Boeing 737 Fuel Burn Analysis

III.2 Boeing 757

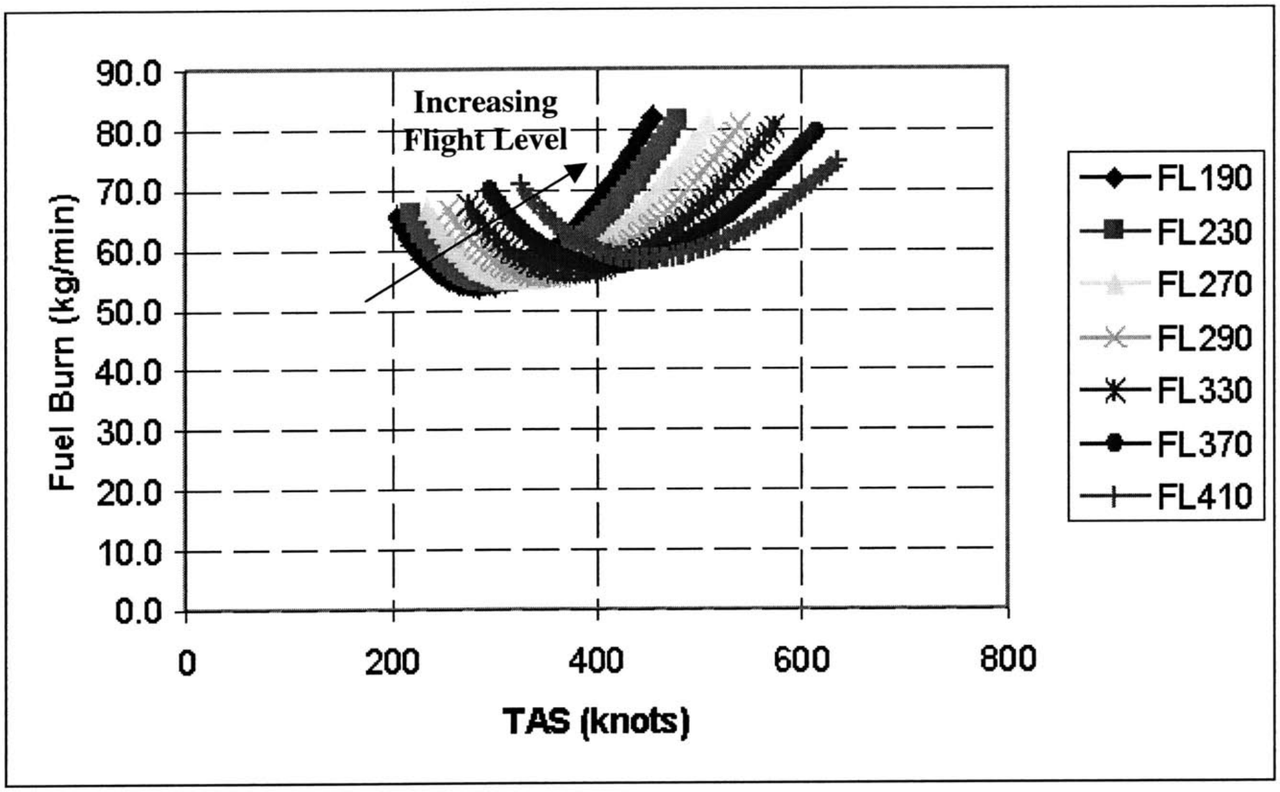


Figure III-2: Boeing 757 Fuel Burn Analysis

III.3 Boeing MD80/MD88/MD90/B717

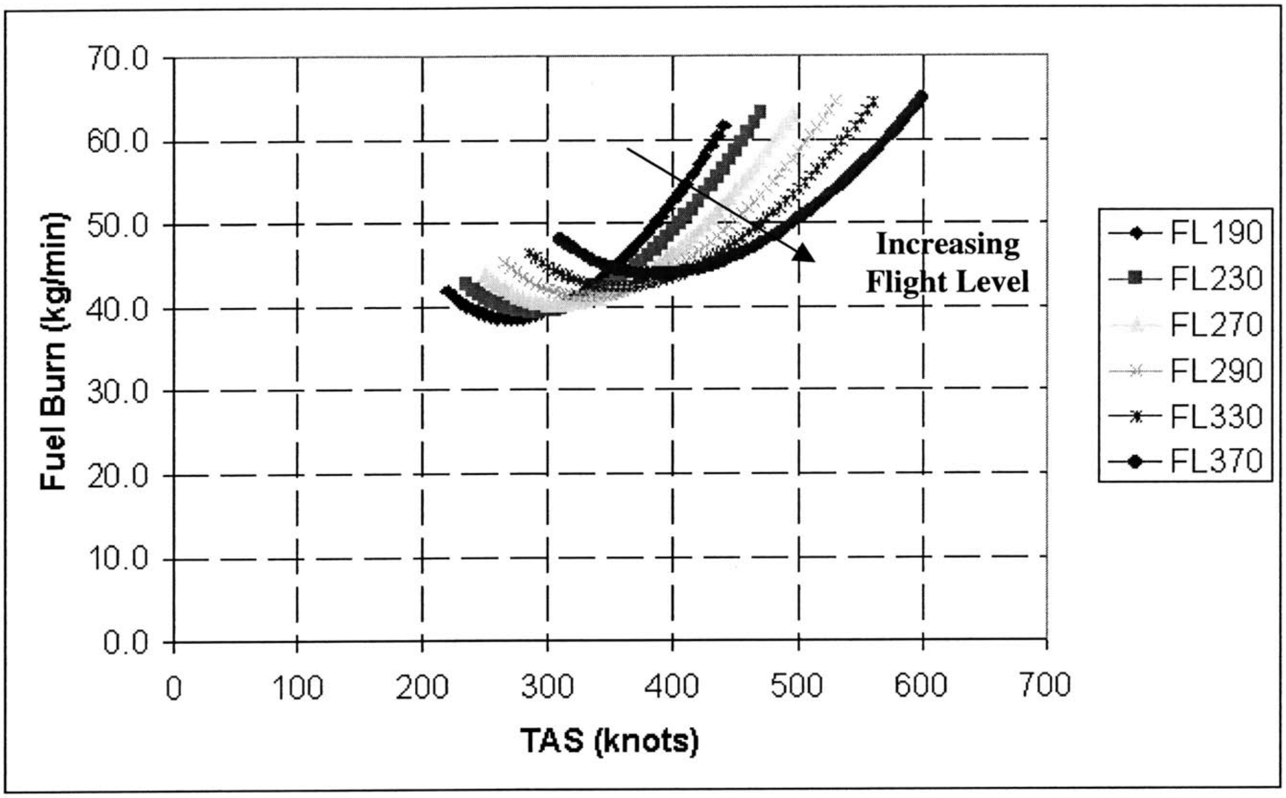


Figure III-3: MD80/MD88/MD90/B17 Fuel Burn Analysis

III.4 Airbus A318/A319/A320

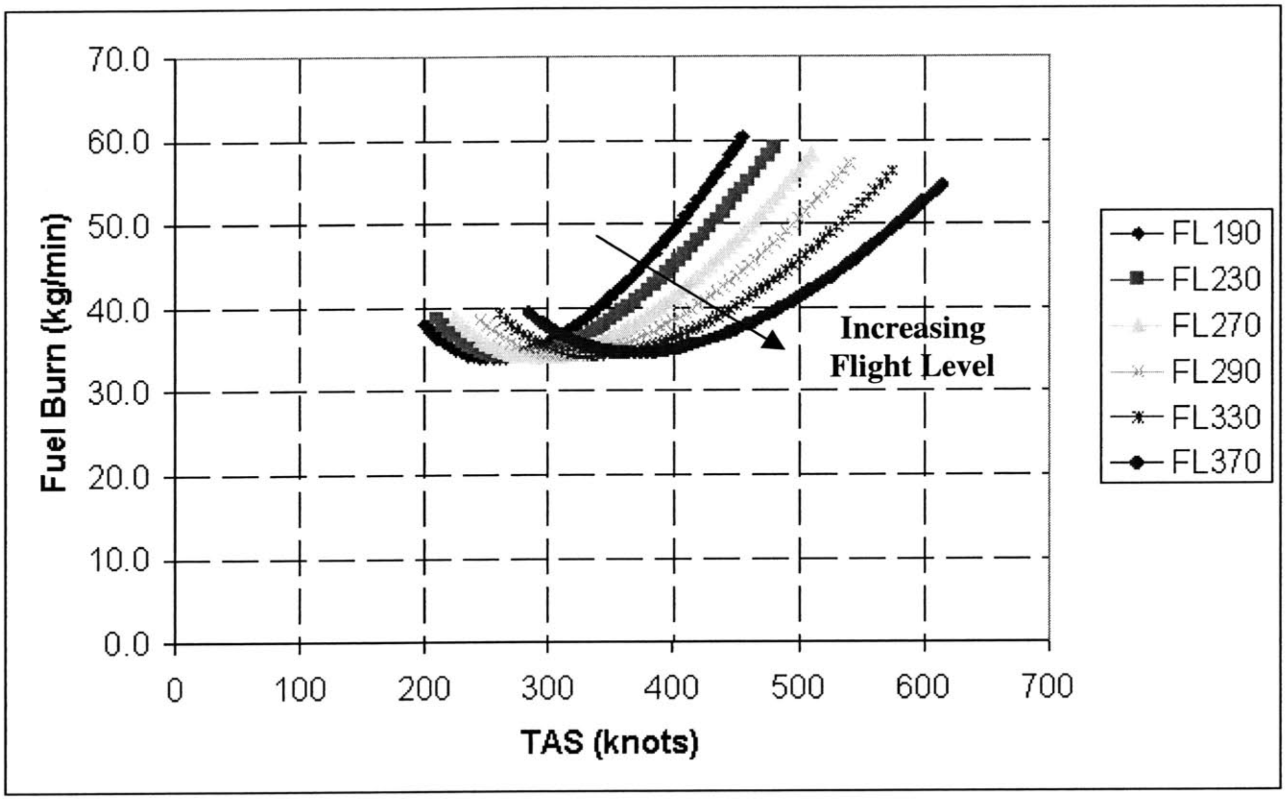


Figure III-4: Airbus A318/A319/A320 Fuel Burn Analysis

III.5 Embraer EMB-145

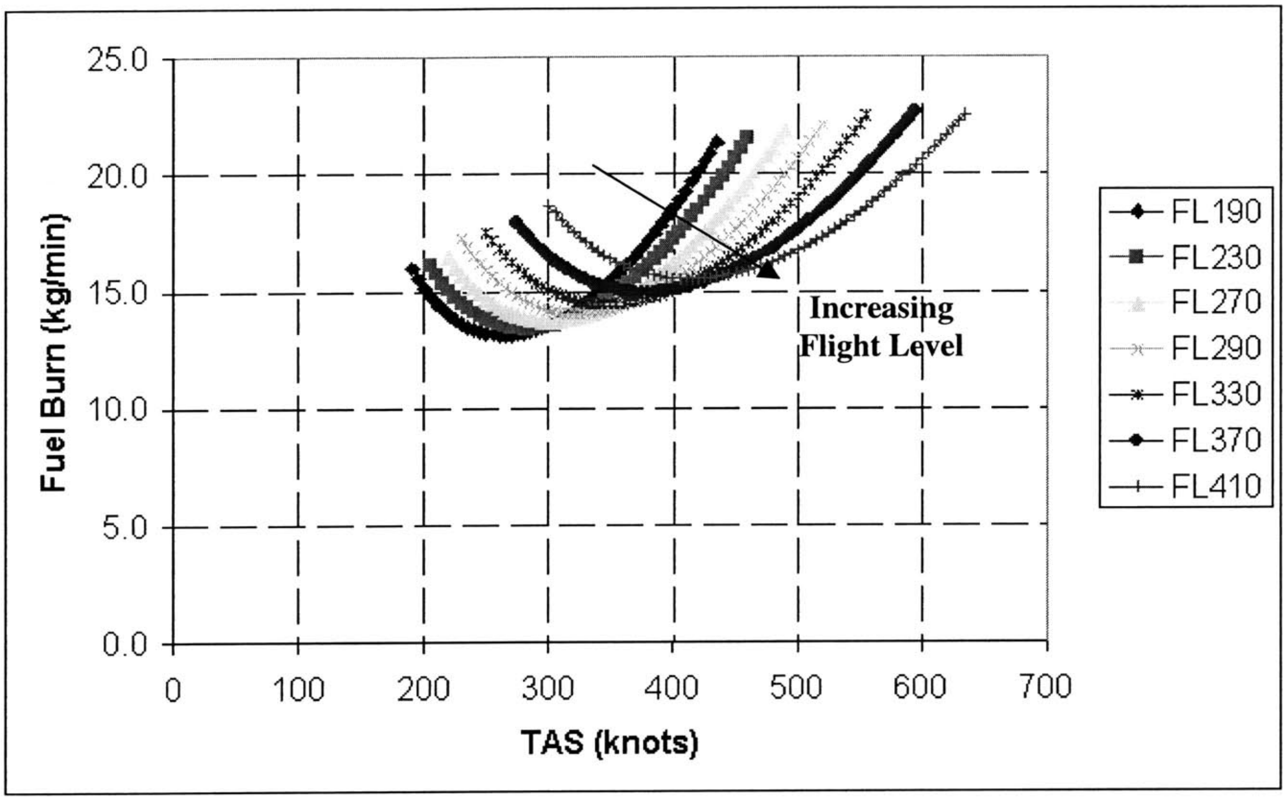


Figure III-5: EMB-145 Fuel Burn Analysis

Appendix IV Piecewise Linear Convex Fuel Burn Approximations

Four linear functions comprise the approximation of the fuel burn curves, each of the following form:

$$fuel\,burn = a_i \times TAS + b_i, i \in \{1,2,3,4\}$$

Figure IV.1 illustrates how each linear function relates to the different regions of the original fuel burn function.

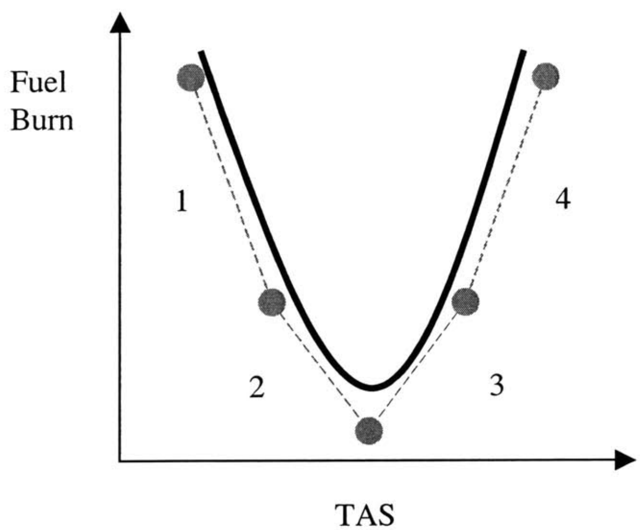
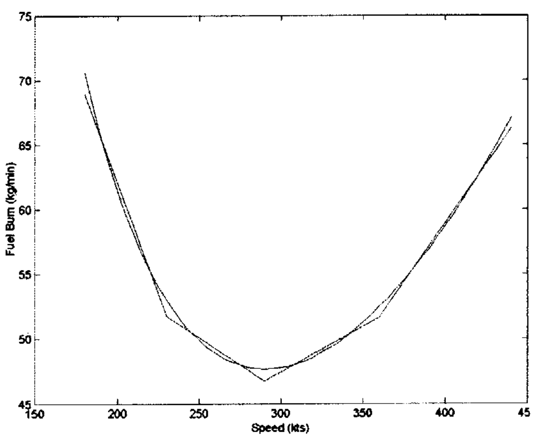


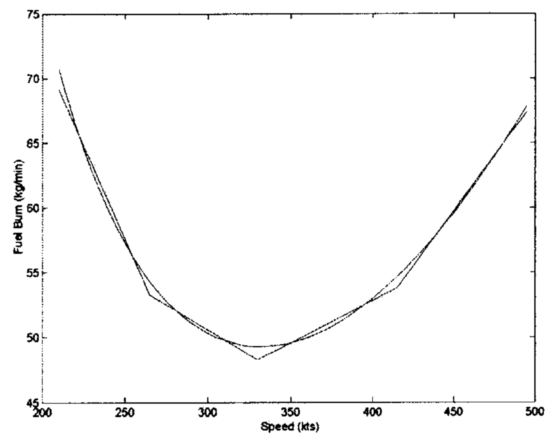
Figure IV.1 Generic 4 Segment Linear Approximation

IV.1 Boeing 737

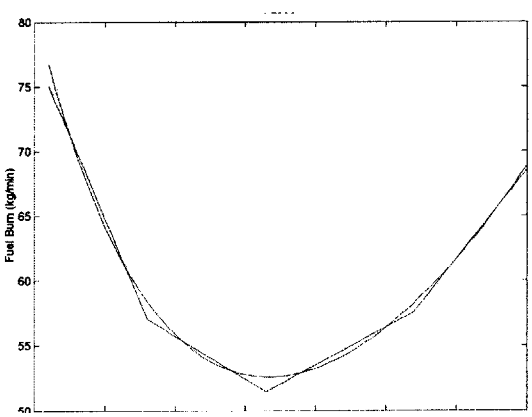
Table IV.1 Boeing 737 Linearized Fuel Burn Coefficients							
	FL190	FL230	FL270	FL290	FL330	FL370	FL410
A1	-0.3438	-0.322	-0.2879	-0.2959	-0.2626	-0.2574	-0.2286
B1	130.8125	131.6027	129.5795	137.5302	135.4557	141.9857	141.8255
A2	-0.0833	-0.0814	-0.077	-0.0743	-0.0686	-0.0657	-0.0593
B2	70.9167	72.6714	73.68	75.4857	76.2857	78.7286	80.0429
A3	0.07	0.0686	0.0651	0.0641	0.0583	0.0581	0.0543
B3	26.45	26.1714	26.7911	26.3653	28.08	27.3375	28.3429
A4	0.1825	0.1758	0.1701	0.1607	0.1491	0.1375	0.1295
B4	-14.05	-15.1209	-16.768	-16.6453	-15.0384	-13.9375	-14.5011
max error (%)	2.4155	2.2355	2.2907	2.7084	2.4316	2.3792	2.7752



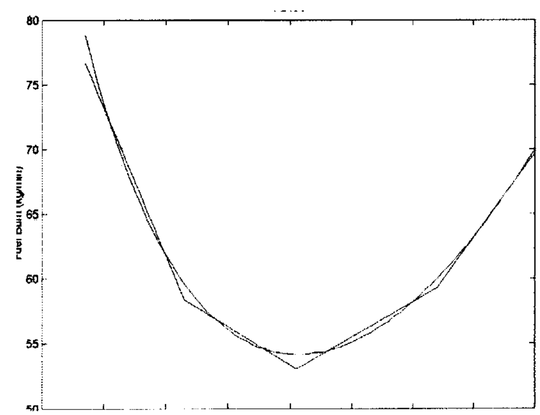
(a): FL190



(b): FL270



(c): FL330



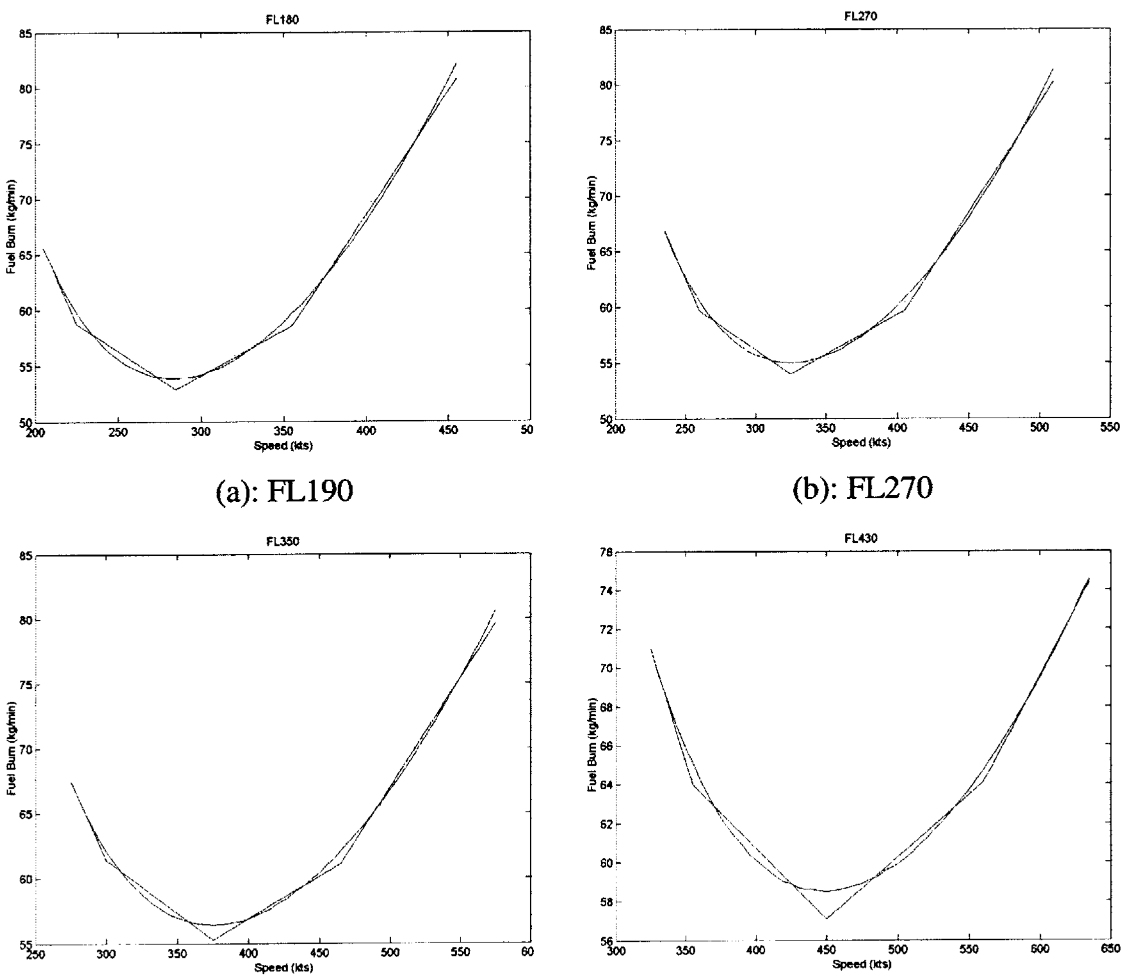
(d): FL410

Figure IV.2: Boeing 737 Linearized Fuel Burn

IV.2 Boeing 757

Table IV.2 Boeing 757 Linearized Fuel Burn Coefficients

	FL190	FL230	FL270	FL290	FL330	FL370	FL410
a1	-0.341	-0.329	-0.2889	-0.3169	-0.244	-0.2497	-0.2336
b1	135.5087	138.4207	134.7875	147.8952	134.6308	143.6517	146.9318
a2	-0.0985	-0.0965	-0.0878	-0.0831	-0.0819	-0.074	-0.0727
b2	80.9496	82.6322	82.5039	83.5976	85.9962	86.56	89.8091
a3	0.0813	0.0822	0.0714	0.0714	0.065	0.0624	0.0637
b3	29.7118	28.1255	30.7571	29.5143	30.9	30.636	28.4295
a4	0.2218	0.2073	0.1952	0.1814	0.1679	0.1555	0.1371
b4	-20.1818	-19.4091	-19.3616	-18.3357	-16.9676	-16.845	-12.7143
max error (%)	1.9404	2.3107	1.9243	2.1384	2.0715	2.0811	2.502



(a): FL190

(b): FL270

(c): FL330

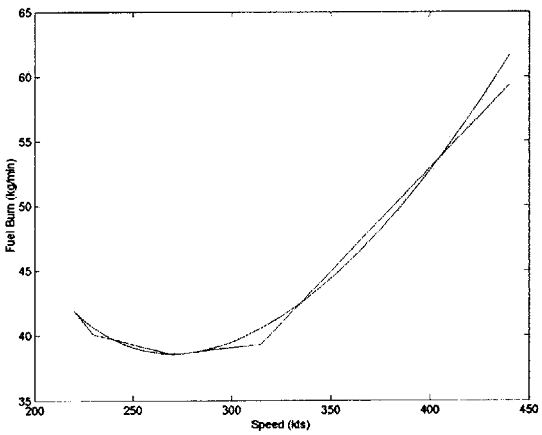
(d): FL410

Figure IV.3: Boeing 757 Linearized Fuel Burn

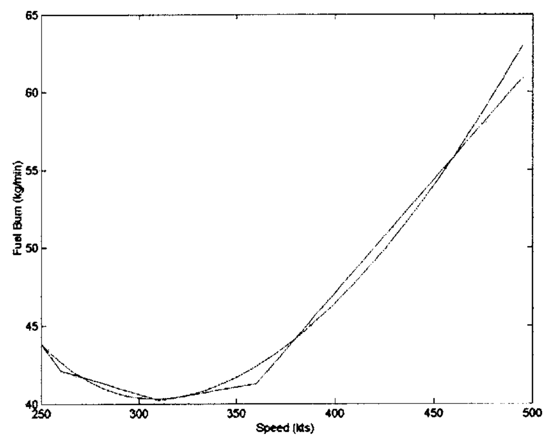
IV.3 Boeing MD80/MD88/MD90/717

Table IV.3 Boeing MD80/MD88/DM90/717 Linearized Fuel Burn Coefficients

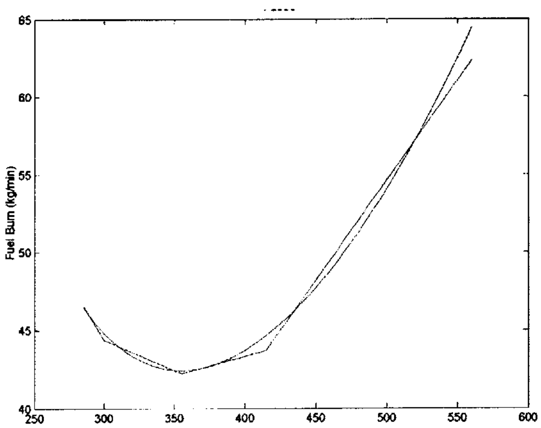
	FL190	FL230	FL270	FL310	FL350	FL390	FL430
a1	-0.1801	-0.1735	-0.1739	-0.1534	-0.1472	-0.1402	-0.1174
b1	81.5171	83.511	87.3519	86.0588	88.5149	91.6444	89.7826
a2	-0.0386	-0.0376	-0.038	-0.0387	-0.0389	-0.0429	-0.0356
b2	48.982	50.2096	52.015	53.935	56.0362	60.0314	60.3323
a3	0.017	0.025	0.021	0.0209	0.0246	0.0278	0.0282
b3	33.965	32.0661	33.7168	34.2765	33.4822	32.8128	33.2222
a4	0.1602	0.157	0.1454	0.137	0.1287	0.121	0.1084
b4	-11.1521	-12.8399	-11.0688	-10.4348	-9.7069	-9.5737	-6.5046
max error (%)	3.8864	3.5781	3.4093	3.7468	3.3751	2.9192	2.2226



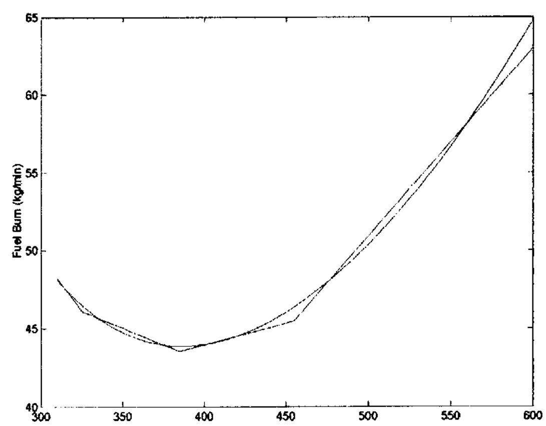
(a): FL190



(b): FL270



(c): FL330

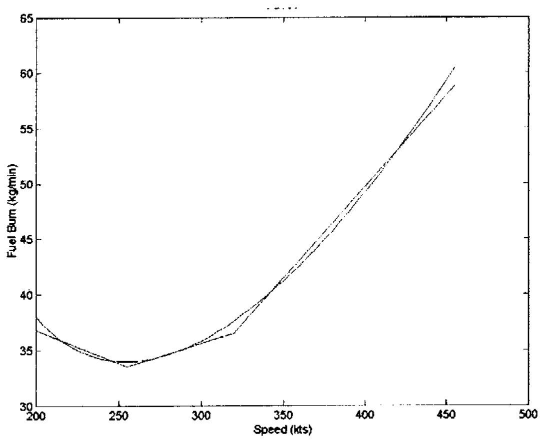


(d): FL410

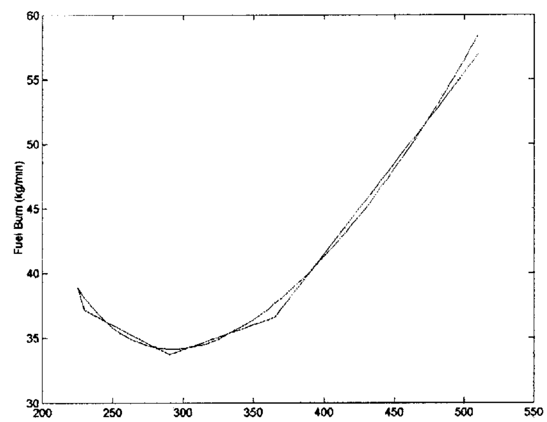
Figure IV.4: Boeing MD80/MD88/MD90/717 Linearized Fuel Burn

IV.4 Airbus A318/A319/A320

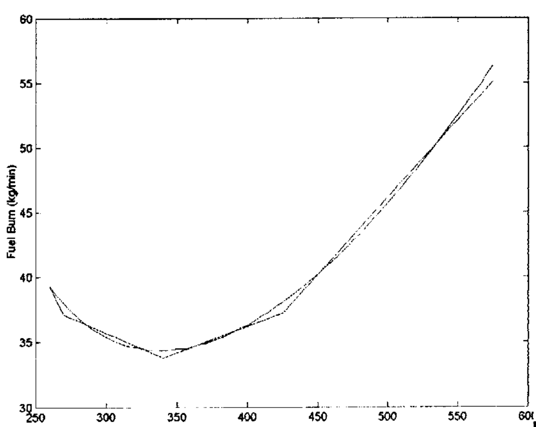
Table IV.4 Airbus A318/A319/A320 Linearized Fuel Burn Coefficients							
	FL190	FL230	FL270	FL310	FL350	FL390	FL430
a1	0.184	-0.3766	-0.3483	-0.305	-0.2188	-0.2022	-0.1644
b1	0.0009	117.8938	117.275	113.325	96.1995	97.0175	91.1273
a2	-0.06	-0.0604	-0.0572	-0.0517	-0.0474	-0.0452	-0.0467
b2	48.8	49.9025	50.3194	49.9917	49.9196	50.7189	52.8667
a3	0.0467	0.0403	0.0383	0.0417	0.0405	0.0357	0.0358
b3	21.6	22.7028	22.6083	20.5917	20.0371	20.79	19.448
a4	0.1647	0.1513	0.14	0.13	0.1188	0.1075	0.0923
b4	-16.16	-15.0063	-14.5	-14.3	-13.2706	-12.6125	-9.3462
max error (%)	3.2609	2.9689	3.0055	2.5641	2.394	2.2074	2.1766



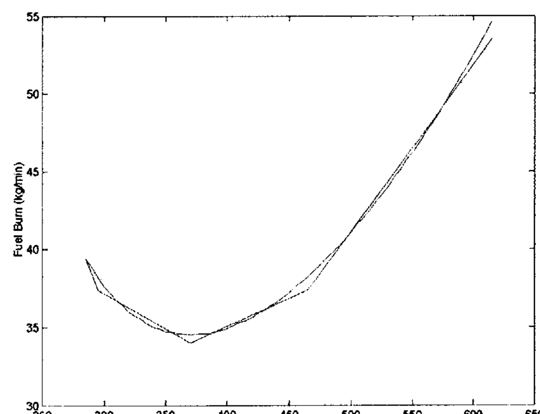
(a): FL190



(b): FL270



(c): FL330



[redacted]

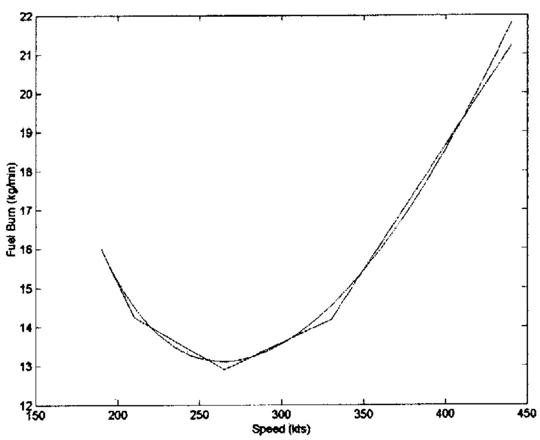
(d): FL370

Figure IV.5: Airbus A318/A319/A320 Linearized Fuel Burn

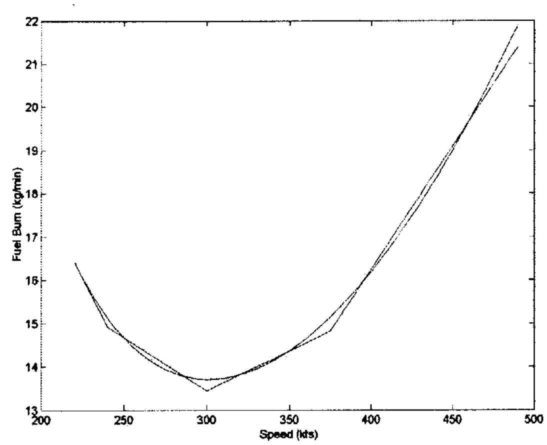
IV.5 Embraer EMB-145

Table IV.5 Embraer EMB-145 Linearized Fuel Burn Coefficients

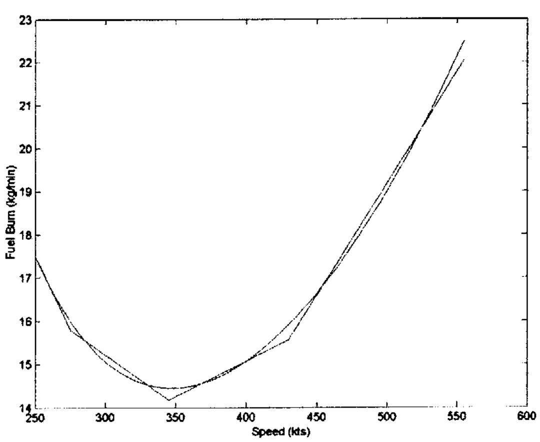
	FL190	FL230	FL270	FL310	FL350	FL390	FL430
a1	-0.0873	-0.0798	-0.0759	-0.084	-0.0695	-0.069	-0.0617
b1	32.5863	32.4955	33.1256	36.6851	34.8883	36.9182	37.179
a2	-0.0244	-0.0244	-0.0246	-0.0205	-0.0231	-0.0194	-0.0189
b2	19.3636	20.0186	20.8099	20.5035	22.1302	22.0423	23.0657
a3	0.0196	0.0174	0.0183	0.0177	0.0163	0.0169	0.0167
b3	7.6995	8.3316	7.9418	8.0898	8.5458	8.27	8.2699
a4	0.0639	0.0605	0.057	0.0538	0.0518	0.0485	0.0445
b4	-6.9099	-6.7561	-6.572	-6.3506	-6.7142	-6.5922	-6.0144
max error (%)	2.821	2.659	2.4161	2.3237	2.2376	1.9547	2.0399



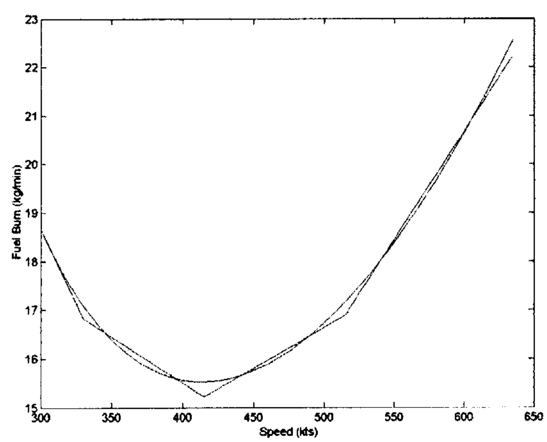
(a): FL190



(b): FL270



(c): FL330



(d): FL410

Figure IV.6: Embraer EMB-145 Linearized Fuel Burn