ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

MSc THESIS

MUHAMMED ALİ ÜLKÜ

ON THE ELIMINATION OF THE AIR TRAFFIC CONGESTION : A NETWORK MODELING APPROACH TO STOCHASTIC STATIC SINGLE AIRPORT GROUND HOLDING PROBLEM

DEPARTMENT OF INDUSTRIAL ENGINEERING

ADANA, 2000

ÇUKUROVA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ON THE ELIMINATION OF THE AIR TRAFFIC CONGESTION : A NETWORK MODELING APPROACH TO STOCHASTIC STATIC SINGLE AIRPORT GROUND HOLDING PROBLEM

MUHAMMED ALİ ÜLKÜ

YÜKSEK LİSANS TEZİ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez 25/09/2000 Tarihinde Aşağıdaki Jüri Üyeleri Tarafından Oybirliği ile Kabul Edilmiştir.

İmza:	İmza:	İmza:
Yrd.Doç.Dr. Rızvan EROL	Doç.Dr. Nedret BİLLOR	Yrd.Doç.Dr. Funda YILDIRIM
DANIŞMAN	ÜYE	ÜYE

Bu Tez Enstitümüz Endüstri Mühendisliği Anabilim Dalında Hazırlanmıştır.

Kod No: 1756

Prof. Dr. Melih BORAL

Enstitü Müdürü

Bu çalışma Çukurova Üniversitesi Araştırma Fonu tarafından desteklenmiştir.
Proje No: FBE.2000.YL.20

ABSTRACT

MSc THESIS

ON THE ELIMINATION OF THE AIR TRAFFIC CONGESTION : A NETWORK MODELING APPROACH TO STOCHASTIC STATIC SINGLE AIRPORT GROUND HOLDING PROBLEM

MUHAMMED ALİ ÜLKÜ

DEPARTMENT OF INDUSTRIAL ENGINEERING
INSTITUTE OF NATURAL AND APPLIED SCIENCES
ÇUKUROVA UNIVERSITY

Supervisor : Assist. Prof. Dr. Rızvan EROL

Year: 2000, Pages:82

Jury : Assist. Prof. Dr. Rızvan EROL

: Assoc. Prof. Dr. Nedret BİLLOR

: Assist. Prof. Dr. Funda YILDIRIM

Ground Holding, a policy for eliminating air traffic congestion, is a mechanism devised to decrease the rate of the incoming flights into an airport when it is projected that arrival demand into the destination airport will exceed capacity. This thesis deals with an Air Traffic Flow Management (ATFM) mathematical model on Stochastic Static Ground Holding Problem (SSGHP) which considers a stochastic profile represented by a set of airport capacity scenarios and their probabilities and is represented in an aggregate level in terms of numbers of flights per unit time to allow the model to be used in the collaborative decision making setting thereby using fewer decision variables via post-processing.

Keywords: Air Traffic Flow Management, Ground Holding Problem, Computational Complexity, and Network Flows

ÖZ

YÜKSEK LİSANS TEZİ

HAVA TRAFİĞİ SIKIŞMALARINI AZALTMAK ÜZERİNE: STOKASTİK STATİK TEK HAVALİMANI YERDE BEKLETME PROBLEMİNE AĞ MODELLEMESİ YAKLAŞIMI

MUHAMMED ALİ ÜLKÜ

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ENDÜSTRİ MÜHENDİSLİĞİ BÖLÜMÜ

Danışman : Yrd. Doç. Dr. Rızvan EROL

Yıl: 2000, Sayfa:82

Jüri : Yrd. Doç. Dr. Rızvan EROL

: Doç.Dr. Nedret BİLLOR

: Yrd. Doç. Dr. Funda YILDIRIM

Yerde Bekletme, öncelikle hava trafiği sıkışmalarını azaltmak üzerine bir politika olup, varış havalimanının kapasitesinin üstünde geleceği kestirilen uçuşları önceden azaltan bir mekanizmadır. Bu tezde, Hava Trafiği Akış Yönetimi kapsamındaki bir matematiksel model olarak geliştirilen Stokastik Statik Yerde Bekletme problemi irdelenmektedir. Modelde havalimanı kapasite senaryoları stokastik bir profilde incelenip, varış havalimanına gelen uçaklar tekil değil tümel (toplam akış) olarak ele alınmakta ve böylece daha az karar değişkeniyle, ortak karar verme süreci geliştirilmektedir.

Anahtar Kelimeler : Hava Trafiği Akış Yönetimi, Yerde Bekletme Problemi, Hesapsal Karmaşıklık, Ağ Akışları

ACKNOWLEDGEMENT

First and foremost, I would like to thank my advisor Assist. Prof. Dr. Rızvan EROL, for his thorough guidance and belief in me and for his helping me in so many ways in the path of this thesis with his special generosity and inspiration. Also, I would like to thank to Assoc. Prof. Dr. Nedret BİLLOR for her "fussiness" in the revision of this thesis and her encouragement in these fledgling steps of my academic life. I also thank to Assist. Prof. Dr. Funda YILDIRIM, as of being a member of my dissertation committee and for her hearty support in the contemplation of my thesis. I would like to express my appreciations to Assoc. Prof. Dr. Ömer BENLİ, Assoc. Prof. Dr. Mustafa AKGÜL, Assoc. Prof. Dr. Ülkü GÜRLER and Assist. Prof. Dr. M. Selim AKTÜRK for they helped me much in the formation of this thesis and my academic career. Also I would like to acknowledge Assist. Prof. Dr. Oya H. YÜREĞİR and Assist. Prof. Dr. Ali KOKANGÜL for their valuable supports.

I would like to acknowledge my friends Didem ÇARMAN, Şeyda SÖNMEZ, Alper KALİBER, Alper YENTÜRK and Gürcan GÜLEŞİR for their moral and constituent efforts.

My deepest gratitude goes to my family, my mother Fatma ÜLKÜ and my sisters Esin KILIÇ, Cansen ÜLKÜ and Nursen ŞAHİNOĞLU for they helped and gave me the cord of power through difficult times in my life.

I thank you all.

DEDICATION

I love you so much MOM !

This thesis is dedicated to YOU.

TABLE OF CONTENTS **PAGE**

ABBREVIATIONS

AAR	: Arrival Acceptance Rate
ADR	: Automated Demand Resolution
ATC	: Air Traffic Control
ATCS	: ATC System
ATFM	: Air Traffic Flow Management
ATFMP	: Air Traffic Flow Management Problem
ATM	: Air Traffic Management
ATMS	: Advanced Traffic Management System
B&B	: Branch and Bound (a solution technique in IP)
CDI	: Capacity-Demand Inequity
CDM	: Collaborative Decision Making
CDT	: Controlled Time of Departure
CPU	: Computer Unit Time
CTA	: Controlled Time of Arrival
DM	: Decision Maker
DGHP	: Deterministic GHPP
EDCT	: Earliest Departure Clearance Time
EDT	: Expected Departure Time
ETA	: Expected Time of Arrival
ETC	: Expected Total Cost
ETGHC	: Expected Total Ground Holding Cost
FAA	: Federal Aviation Association
FCFS	: First-Come First-Served (a queuing discipline)
GDP	: Ground Delay Program (likewise GHP)
GHB	: GHP with Banking Constraints
GHP	: Ground Holding Policy
GHPP	: Ground Holding Policy Problem
ICAO	: International Civil Aviation Organization
IFR	: Instrument Flight Rules

IP : Integer Programming
LP : Linear Programming
MAS : Multiple Airport Scheduler
MCNFP : Minimum Cost Network Flow Problem
MIP : Mixed Integer Programming
MT : Ministry of Transportation
NLP : Nonlinear Programming
OC : Operations Coordinator
OR : Operations Research
PAR : Planned Arrival Rate
RHS : Right Hand Side
RM : Richetta Model
SSGHP : Static Stochastic Ground Holding Problem
SSGHP* : Proposed SSGHP
SSM : Stochastic Static Model
STN : Space Time Network
SWAP : Severe Weather Avoidance Program
TAM : Time Assignment Model
TFM : Traffic Flow Management
TFMP : Traffic Flow Management Problem
TU : Totally Unimodular
VFR : Visual Flight Rules

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

In recent years air traffic has dramatically increased without a corresponding development of airports. Therefore the limited capacity of the airports causes air traffic congestion and consequent expensive delays.

Strategies for reducing or eliminating air traffic congestion may be grouped into three classes (Odoni, 1987):

1. *Long-term strategy*, which consists of building new airports or enlarging existing ones or extending the utilization of wide body aircraft; such a strategy requires a term of some years and large investments.
2. *Medium-term strategy*, comprising of getting an even distribution of demand both in space (over different airports) and in time (over different time intervals of the day); such a strategy requires medium time (months) and low investments, but it encounters many organizational difficulties.
3. *Short-term strategy*, which consists of the optimal management of present resources, respecting airline schedules as much as possible; such a strategy can be implemented in the short term (hours), requires very low investments, and encounters few difficulties for its application.

Assigning ground-holding times to flights, i.e., determining how much to delay the take-off of a particular aircraft headed to a congested part of the Air Traffic Control (ATC) system has become one of the most effective short-term approaches to diminishing the impact of airport delays and this subject embodies the focus of this thesis.

1.1. Outline of the Thesis

In this thesis, the so-called Stochastic Static Ground Holding Problem (SSGHP) will be elaborated in detail. The setting of the thesis comprise of 6 chapters. In Chapter 1, an introduction to the ATC system and its operations, the interface points of Ground Holding Policy (GHP) with ATC and the current data on air transportation statistics in Turkey, are given, to the scope that SSGHP tackles with.

Chapter 2 (Recent Studies), delves with the recent studies on the Air Traffic Flow Management (ATFM) and gives basic comparison of the most recent mathematical models and exemplifies these models. Chapter 2 is to strengthen the background for Chapter 3, in which, Ground Holding Policy Problem (GHPP) is first introduced as a perturbation problem in ATFM, and then it is examined in detail for its various cases like deterministic, or probabilistic and the like. In Chapter 4, the problem is exactly specified and SSGHP is introduced and possible improvements and sensitivity analysis are explicitly divulged both by theoretical means and computational results. Chapter 5 delves with the research findings and discusses the results, especially concentrating on the computational efficiency aspects. Finally, in Chapter 6, the thesis is closed with emphasis on the results attained and future research areas, initially giving an overall research summary. The appendices comply with the data sets, OPL codes, optimization results and some theoretical proofs. Also the abbreviations in the thesis are tabulated to ease the reader.

1.2. Motivation

The ATC has become one of the most challenging and demanding research areas in the Operations Research. Luo and Yu (1998) report that in the US, delays due to ATC and TFM factors have costed the industry $1.5 billion per year in fuel, crew, and other direct operating costs. An article in The Economist (Nov., 1989) gives an estimate of total yearly delay costs for European airlines $8 billion. Indirect costs associated with missed passenger and crew connections, decreases in costumer satisfaction, and equipment under utilization are more difficult to measure but are likely to increase this total. This situation is not likely to improve in the near future as demand for air transportation areas is predicted to continue to grow. Moreover the problem is exacerbated by the use of 'hub and spokes'[1] scheduling system by almost all major airlines.

[1] This system consists of using, for reasons of operational and economic efficiency, a small number of designated airports as transit points for a large proportion of all flights of a given airline, thereby generating a large number of artificial operations at these airports.

Besides its economic and safety considerations, the focus of this thesis forms a challenging subject in computational optimization and has a promising academic value in current operations research field.

1.3. A Brief Note On the Nature of the Airline Systems

The overall efficiency of the ATC system is directly affected by the efficient control/management of the air traffic as well as adequacy of the landing/takeoff areas or airports. It is obvious that the airports must be improved in pace with the enhancements in ATC systems regarding the development of new aircraft and expanding air traffic volume should the airports not become serious bottlenecks for safe and efficient air transportation (Hall, 1985).

An airline system can be divided into six components as:

1. Airspace
2. Runway
3. Taxiway
4. Apron-gate
5. Terminal building
6. Ground-access facility

The first four items in the list above are involved in the air traffic system while the last two items, theoretically, must not affect the air traffic. The layout of the airport terminal buildings and the location of the airport have no influence on the aircraft movements, but they are likely to impose delays for the passengers. Among the components of an airport system, the airspace and the runway are usually the critical ones, which limit the airport capacity.

There are two major factors restricting the capacities of airspace and runway components: minimum separation criteria for two consecutive landing or taking off aircraft and runway occupancy time. The delays imposed on the air traffic can be classified into two categories: the delays due to the composition of the traffic and the ones due to runway non-availability.

The minimum separation criteria are imposed on the landing and takeoff procedure to prevent aircraft mishap due to inadequacies in the ATC system and due to the wake vortices generated by the leading aircraft. The number of the runways in use directly affects the runway non-availability. For the sake of safety, an aircraft is not allowed to use runway if another aircraft remains on the same runway. Hence, the capacity of an airport would not increase proportionally as the minimum separation time criteria decreases, unless the runway occupancy time decreases too; where the runway occupancy time is defined as the time interval from the instant the landing aircraft passes over the runway threshold until it completely clears the runway.

Figure 1.1. signifies that there is a two-sided interaction between the aircraft air delay and aircraft ground delay as well as the one between the aircraft air delay and the passenger ground-access delay. It is signed out that the passenger ground-access delay is independent from the aircraft air delay.

An easy solution to the delay problem is to limit the volume of air traffic, which can use the airport under specified conditions, or periods of times; but this is not a remedy for the problem.

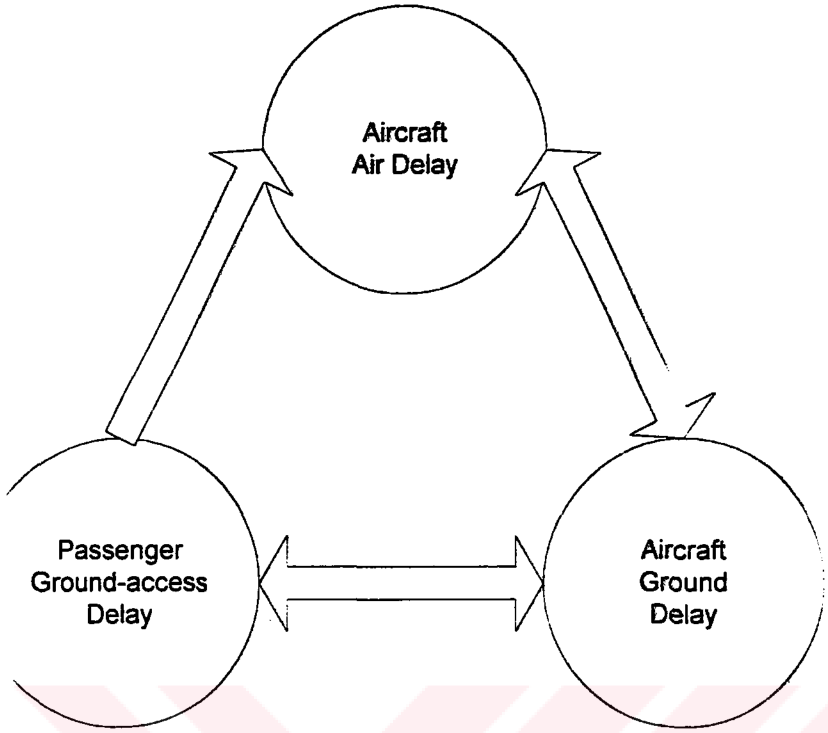

Figure 1.1. The classification of delays around an airport, showing the cause-effect relationships.

To meet public demand in a total transportation concept is to apply more effective methods to achieve the highest possible degree of efficiency in the use of airspace and airports. This objective depends to a large extent on the capability of the overall ATC system. The objectives of an air traffic service, as defined by the International Civil Aviation Organization (ICAO), are to:

- prevent collisions between aircraft in flight,
- prevent collisions between aircraft on the maneuvering area of an airport and obstructions on that area,
- expedite and maintain an orderly flow of air traffic,
- provide advice and information useful for the safe and efficient conduct of flights and notify appropriate organizations regarding aircraft in need of search and rescue aid, and assist such organizations as required.

The degree of air traffic control exercised by this system depends basically on the meteorological conditions in which an aircraft is flown. When an aircraft can be flown clear of clouds and the pilot has good visibility, the flight is conducted in accordance with "Visual Flight Rules", namely *VFR Flight.* VFR flights are subject to little or no control by the ground facilities. If a flight cannot be conducted via VFR, it must be conducted under "Instrument Flights Rules", or *IFR Flight,* and the ground facilities exercise positive separation control over all such flights. The traffic control procedures and parameters differ for these two types of flight conditions. Long-range radar, which provides position information, and computers, which perform many of the routine function of the controller, are utilized to assist the controller to achieve desired separation between aircrafts.

1.4. On The Processes of Airline System Operations Control

The process of Airline System Schedule development is unique to each airline, which has its own organizations names and time frames for these activities. In (Grandeau et al., 1998), these processes are separated in four distinct phases as Service Planning, Schedule Generation, Resource Assignment and Execution Scheduling. The four phases basically follow a sequential path, but there are iterations back to the previous phases to resolve conflicts. The process builds upon previous plans until the final timetable is generated and implemented as monitored in Figure 1.2.

The process of schedule generation begins with the service-planning phase. The goal of service planning is the production of the service plan, which is a set of services that the airline will be offering in each market. The level of detail varies from airline to airline, but generally, it is a complete schedule for a full cycle period. A cycle period is one day in domestic service and one week for international services. Tentative flight times may be specified, or simply time windows, with whatever level of accuracy the airline chooses for this purpose. The marketing group considers several factors in creating the service including marketing initiatives, expected competitive services, traffic forecasts, current schedules and a list of potential resources.

The production of the passenger schedule is the sole domain of the scheduling group. Presented with a service plan, a list of generic resources, and a set of operating constraints, the scheduling group generates the detailed optimal schedule that the airline will present to the travelers, and that the operations group will work to maintain against all irregularities. The service plan presents a general outline for market services for the airline during the scheduling period. There are many constraints that the schedulers must consider in making passenger schedule feasible. The network must exhibit balanced circulation flow. At every station, the number of aircraft of each type that flows in must be equal to the number that flows out. The network must also exhibit circulation, i.e., all flows repeat each other. There are different solution techniques but these constraints must be met. There are other constraints to form a contingent passenger schedule such as the size of each aircraft fleet, the number of available crew members and the training of the crew members on available operations, the number of available gates and the size of the apron at each airport and the like.

The scheduling group turns over the passenger schedule, along with the crew bid lines and the aircraft rotations to the operations group. Operations then begin the process of assigning the specific resources to the respective schedules. Finally the schedule is executed unless a rescheduling is required.

1.5. Air Transportation Status In Turkey

The share of the air transportation of the Ministry of Transportation (MT) of The Republic of Turkey in overall transportation has increased from %0.11 (in 1983) to %0.19 (in 1995) (MT Commission Report, 1998). With the same trend, the share in passenger transportation has increased from %0.79 to %1.63, in respective years. This increasing trend is projected to go on for near future.

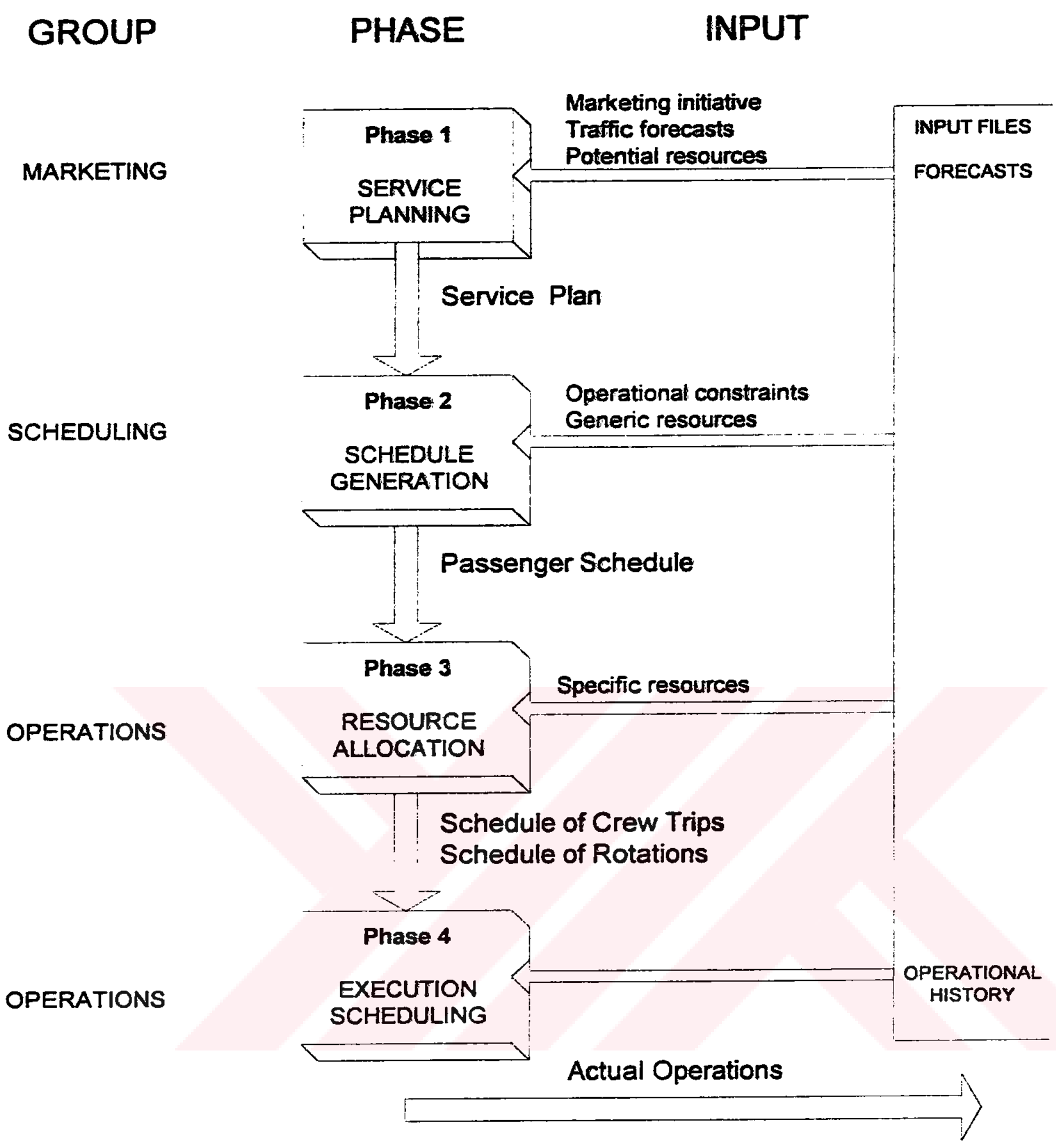

Figure 1.2. The Four Phases of Schedule Development, (Grandeau, et al., 1998)

Currently, there are 8 private airline agents besides THY (Turkish Airlines) and 124 big aircrafts serving with total 21526 seat capacity, overall sharing more than %50 of the international air transportation market. The Turkish air transportation companies can fly to 60 international airports. The passenger traffic loads in 1997 were 198.000 domestic and 207.000 international. The total number of the domestic and international passengers accumulated to 35 million in the same year.

The current Chief Executive Officer of THY claims that THY will do the best of its history and will exceed the goal of 12 million passenger load in 2000 (Radikal, 27 Aug. 2000).

Table 1.1. depicts the general structure of airline industry in Turkey, decomposing private and state companies with respect to their seat capacities. Just to give an insight, Table 1.2. exhibits capacity and load of İstanbul Atatürk Airport and that Table 1.3. gives annual estimates of passenger load in major airports in the world.

It is likely that GHPP is not applied in Turkey because there are no severe weather condition differences between the regions in the country, of course those regions that have airports. Another possible reason is that the flight load is not so much now and the management of the current air industry is unfortunately unable to project the future air transportation capacity requirements in Turkey. Or there are not enough technical staff and specialists to work on this matter.

Table 1.2. and Table 1.3. explicitly show the passenger load differences in Turkey and in the world via major airports. The load figure in the major airports of the world show how this problem gets to larger sizes as compared to that of İstanbul Atatürk Airport.

Table 1.1. General Structure of Airlines in Turkey (MT Report, 1998)

Name of Company	Date of Commence	Aircraft Type	# of Aircrafts	Seat Capacity	Total Capacity
THY	May 1956	A 340-311	5	271	1355
		A 310-303	7	225	1575
		A 310-304	4	210	840
		A 310-318	3	176	528
		B 737-400	28	150	4200
		B 737-500	2	117	234
		RJ 100	9	99	891
		RJ 70	4	79	316
		B727-200F	3	Courier	
TOTAL			**65**		**9939**
İHY	Dec. 1985	B 757-236	3	228	684
		B 737-400	8	168	1344
		B 737-300	2	148	296
		B 727-200	5	170	850
TOTAL			**18**		**3174**
KKTY	Dec. 1974	B 727-200	4	164	656
		A 310-200	1	226	226
TOTAL			**5**		**882**
ONUR	Apr. 1992	A 320	3	180	540
		A 321	4	220	880
		A 300	2	316	632
		MD 88	5	172	860
TOTAL			**14**		**2912**
SUN EXPRESS	Mar. 1990	B 737-300	3	148	444
		B 737-400	1	167	167
		B 737-400	1	170	170
TOTAL			**5**		**781**
PEGASUS	Jan. 1990	B 737-400	6	170	1020
TOTAL			**6**		**1020**
AIR ALFA	Jul. 1994	A 321	2	210	420
		A 300-B4	3	318	954
TOTAL			**5**		**1374**
TOP AIR	Jun. 1990	B 727-200	3	170	510
TOTAL			**3**		**510**
GTI	Sep. 1996	A 300-B4	2	310	620
		A 300-B2	1	314	314
TOTAL			**3**		**934**
GEN. TOTAL			**124**		**21526**
Private				**59**	**11587**
State				**65**	**9939**

Table 1.2. İstanbul Atatürk Airport Statistics (Yıldırım, 1995)

Year	Capacity (millions)	Passenger (millions)	% increase	Peak Day	Peak Hour
1991	7.5	5.2	-	21,188	1,674
1992	7.5	7.4	42	37,936	2,751
1993	8.5	9.4	27	47,675	3,212

Table 1.3. Passenger Loads in Major Airports in the World (Hall, 1985)

Airport	Passenger (millions)
O'Hare	64.4
Dallas	51.9
Los Angeles	46.9
London	45
Tokyo	42

2. RECENT STUDIES

In this chapter, the recent studies are explored from a general perspective beginning from the Air Traffic Flow Management (ATFM), then Ground Holding Policy (GHP) and so on. The reader should notice that the subject of this thesis is fledgling in the literature in the world and the first in Turkey's science literature. Thus it also gets the responsibility of introducing the problem in detail, from a wide scope to the finally narrowed one on Stochastic Static Single Airport Ground Holding Problem (SSGHP). Thus the recent studies for each level of scope are spread over the chapters but mainly are given in this chapter.

2.1. A General Evaluation of Optimization Models for Air Traffic Flow Management

Growth of air traffic demand with no matching growth in capacity causes congestion and delay. Moreover, bad weather conditions aggravate the capacity-demand imbalance/inequity (CDI). Congestion and delay are not only expensive but can also increase controller and pilot workloads, and may jeopardize safety in extreme cases. The Traffic Flow Management Problem (TFMP) is simply matching demand to capacity when the former exceeds the latter. Odoni (1987) classifies the solutions to TFMP as long, medium or short-term as mentioned in the first section. Generally accepted strategies to deal with TFMP can be categorized as follows:

- Miles-in-trails restrictions
- Severe Weather Avoidance Program (SWAP) plans
- Ground stops
- Ground delay or Earliest Departure Clearance Time (EDCT) programs

Miles-in-trail restrictions control the flow of aircraft in the en route environment to prevent sector overloading. SWAP involves rerouting flights to avoid bad weather. Ground stops/delays, as the name implies, are used to hold flights on the ground, usually as a last resort, during bad weather or any other condition under which an airport has no arrival capacity. The EDCT program is the major ATFM strategy, which is implemented when capacities are expected to decrease below demand, for example when inclement weather is predicted. Under this program, pre-departure delays are issued to aircraft whose destination airports anticipate reduced capacities. These decisions are based primarily on the capacity forecast for the affected destination airports and decisions affecting each destination are made independently of the decisions affecting the other destination airports.

2.1.1. Recent ATFMP Optimization Models

ATFM models focused on five types of operations research (OR) models (Helme et al., 1992): queuing, dynamic programming, linear, integer, and mixed integer programming, network flows, and nonlinear programming.

Queuing models are deemed inappropriate for two reasons. First, these models do not model the ATFMP with the best fidelity. Daily commercial air traffic does not attempt to enter the system according to a Poisson process: departure schedules are generated and known weeks ahead of time. Second, queuing methods do not optimize! While queuing models enable evaluating the impact of a particular strategy, they do not seek or generate improved strategies. Dynamic programming models are deemed useful for the multiple destinations; yet the other models outperform and have prohibitive execution times when compared. Linear, integer and mixed models are promising since these models are flexible, allowing a wide variety of constraints to be incorporated to the model as the model's requirements evolve.

Linear programming (LP) models can be solved for all but the largest of problems, which has not been historically true of integer programming (IP) and mixed integer programming (MIP) models. However, recent research has uncovered heuristics that, when used, make these models more computationally tractable. Linear network optimization models are a subset of linear IPs and are suitable candidates for TFMP optimization models. They are less flexible than the general LP and IP models yet their special structure often permits faster solution methods than the methods for solving generic IPs. Non-linear programming (NLP) models offer more flexibility than all other optimization methods but unfortunately existing algorithms can solve only problems of limited size. Thus NLP models do not appear to be suitable candidates for solving TFMP as these problems can be extremely large. If the ATFMP can be cast as a network optimization problem, timely solutions would be guaranteed.

The recent TFMP models can be classified as:

- The Time Assignment Model (TAM)
- The Space-Time Network (STN)
- The MIT TFM Model
- Multiple Airport Scheduler (MAS)
- OPTIFLOW

It is noteworthy to give brief explanations of the model before comparing them. The TAM (Lindsay, 1993; Burlingame et al., 1995) was developed specifically to solve TFMP and is formulated as a 0-1 IP that consists of an objective function and five types of constraints. TAM generates individual aircraft decisions, which are selected over aggregate flows because the latter must ultimately be disaggregated before they can be implemented. The TAM represents scheduled demand deterministically allowing dynamic capacities. The TAM also represents restrictive en route capacities and aircraft itineraries. The model features multiple flights from origins bound for multiple destinations to allow any interaction among flights that may be caused by restrictive capacities and aircraft itineraries. At present, uncertainty is not explicitly represented in the model.

The STN (Helme et al., 1992), like the TAM, was developed to solve the TFMP but through the use of an alternative to the more generic IP formulations. The model is formulated as a multi-commodity minimum cost flow network. The objective function is to minimize the delay cost and the currently modeled constraints consist of departure, en route, and arrival capacity constraints. STN decision variables are flows from one space-time point to another. As the model's decisions are aggregate flows, some post-processing is necessary before the decisions can be implemented. The STN model is deterministic supporting dynamic capacities.

Itineraries are not currently modeled by STN since STN decision variables represent flow while itineraries are relevant to individual aircraft. The model does represent multiple destination airports that are genuine feature of the TFMP. To solve STN, software is developed to generate a heuristic solution. The solution entails starting with a feasible solution - specifically, the EDCT solution- and using the network to identify a sequence of improving loops, which are equivalent to less costly solutions within an impressive solution time.

The MIT TFM Model (Bertsimas and Stock, 1998) was also developed specifically for TFMP and is formulated as a 0-1 IP problem, which consists of an objective function and six types of constraints. The objective function is the weighted sum of the air and ground delay costs over a set of specified airports during the problem time horizon. The time horizon is partitioned into discrete intervals. A 0-1-decision variable is specified for each flight for each time interval for each sector in the flights route. Each decision variable specifies whether or not the indicated flight should arrive at the indicated sector by the indicated time interval. A value of 1 specifies affirmative and that of 0 specifies negative.

The MAS (Epstein et al., 1992) was developed to perform automated demand resolution (ADR) for the Federal Aviation Association's (FAA) Advanced Traffic Management System (ATMS). The MAS uses a collection of heuristics and optimization techniques to solve the ATFMP. The final MAS outputs are decisions for the individual aircraft. The MAS methodology designs the airspace as a network with the goal of maximizing flow in the network subject to departure, sector and arrival capacity constraints, flight connectivity (itinerary) constraints, and fairness-to-user constraints. The network is decomposed into sub-networks, each of which consists of a single arrival airport, to avoid computational difficulties associated with multi-commodity networks. OPTIFLOW is also another TFMP optimization model, which is adaptation of the early version of the MIT Model. The model was developed to generate TFM strategies that are better that the current EDCT ground delay strategies. OPTIFLOW inherits its 0-1 IP formulation from the early MIT Model. The model's decision variables specify whether or not each flight is to land in each time interval. The current version of the OPTIFLOW represents a single arrival

Table 2.1. Comparison of Fidelity with which Models Represent ATFMP (Lindsay et al., 1998)

TFMP Feature	TAM	STN	MIT Model	MAS	OPTIFLOW
Multi-Airport	+	+	+	+	-
En Route Airspace	+	+	+	+	-
Itineraries	+	-	+	+	-
Flight Banks	-	-	-	-	-
Uncertainty	-	-	-	-	-
Dynamic Capacities	+	+	+	+	+

airport and assumes all delay is taken on the ground. The model's objective function, delay cost is minimized subject to several constraints such as airport arrival capacity, earliest departure time, flight order preservation, and bank reservation constraints. Yet overall, it may be argued that OPTIFLOW is currently being implemented and tested. Future improvements are projected to the model as airport departure capacity, en route capacity and itinerary constraints.

2.1.1.1. Comparison of the Models

Table 2.1. reveals that the TAM, MIT, and MAS models represent an identical set of ATFMP features and therefore have the same degree of fidelity. The STN is not included in this group of three models because it does not represent itineraries. OPTIFLOW stands out from the other models because the current version does not represent multiple arrival airports, en route airspace capacity and itineraries.

Overall, two conclusions can be drawn. First, the degrees of commonality in the features of the problem that are modeled suggest agreement among the researchers that these are the essential features of the problem. Second, the fact that none of the models currently represent flight banks or uncertainty is an indication of the difficulty of modeling the feature or perhaps that the feature is not essential. Uncertainty is clearly essential to the problem and all of the researchers admit this directly or indirectly. Recently, Ball et al. (1999) tries to find a solution to uncertainty in a sense assuming a stochastic arrival nature at the destination (congested) airport and Hoffman (1997) tries to improve an IP model regarding the banking constraints faced in 'hub and spokes' operations.

Table 2.2. compares and contrasts the formulation features of ATFMP optimization models as they currently exist. For each model, the table presents the optimization type, the objective, the optimization sense, and a brief characterization of the constraints in the model. All of the models are static. Except for the MAS, all of the models minimize a delay cost objective function. The MAS maximizes flow through a network. The TAM and MIT model formulations are identical except for the definition of the decision variables and the manner in which each represents the en route airspace. Both the TAM and MIT models are formulated as 0-1 IP problems and both produce individual aircraft decisions. Both models feature airport arrival and departure capacity constraints, earliest departure time constraints and fix capacity constraints, where a *fix* is a point on an aircraft's flight path that is used as a conceptual representation of a sector of airspace.

Although the STN and the MAS are both network models, they are no closer to each other in formulation features than they are to any of the other models. The models are difficult types of networks; the STN being a minimum cost multi-commodity flow network and the MAS is a single-commodity maximum flow network. The STN generates aggregate flow decisions as an intermediate result and individual aircraft decisions as the final result. The formulation features are important in that they have computational implications. Networks have historically been easier to solve than generic IPs and this may motivate such a formulation. However, this is only true for single-commodity networks and a multi-commodity network best presents the ATFMP. Generic IP formulations are more flexible than network formulations and can provide a more realistic representation of ATFMP. This, coupled with the fact that preprocessing techniques and stronger formulations can now make IPs computationally tractable may motivate a 0-1 IP formulation.

Table 2.2 Comparisons of Models by Formulation Features
(Lindsay, et al., 1998)

Formulation Feature	TAM	STN	MIT Model	MAS	OPTIFLOW
Model Type	0-1 IP	Multi-Commodity Network	0-1 IP	Network	0-1 IP
Objective Function	Delay cost	Delay cost	Delay cost	Flow	Delay cost
Optimization Sense	Minimize	Minimize	Minimize	Maximize	Minimize
Airport Capacity Constraints	Arrival / Departure	Arrival / Departure	Arrival / Departure	Arrival / Departure	Arrival
En Route Capacity Constraints	Fix	Sector / Fix	Sector	Sector	None
Feasible Flight Time Constraints	Inter-Fix	Inter-Fix	Inter-Sector	None	None
Departure Time Constraints	Earliest Departure Time	Earliest Departure Time	Earliest Departure Time	Earliest Departure Time	Earliest Departure Time
Feasible Flight Route Constraint	Route	Route	None	None	None
Decision Time Ordering Constraints	None	None	Itinerary	Itinerary	None
Flight Connectivity Constraints	Itinerary	None	Itinerary	Itinerary	None
Decisions	Individual Aircraft	Aggregate Flows	Individual Aircraft	Aggregate / Individual	Individual Aircraft
Static / Dynamic	Static	Static	Static	Static	Static

2.2. A Review of Basic ATFMP Mathematical Models

Whether for single or multi airport, the basic seminal models that stand to evolve GHPP will be explained briefly in this section, mainly concerning the mathematical formulations and some discussions on the solution methodologies of these models.

2.2.1 The Time Assignment Model (TAM)

The TAM is formulated as a 0–1 IP problem. The objective function is to minimize the sum of the ground and air delay costs for selected flights between selected airports during some period of interest. The decision variables are the times at which aircraft are to be at selected fixes in their routes. The time period is partitioned into S time intervals T_j, $1 \leq j \leq S$ of equal length, for example 15 minutes. A 0–1 decision variable is specified for each flight i for each time interval j for each fix k in the flight's route. The decision variable, x_{ijk}, specifies whether or not the flight is to be at the fix in the time interval. A value of 1 specifies affirmative and of 0 does negative:

The IP formulation of the TAM is as follows:

$$\min \sum_i \sum_{k \in f_i} C_{ik} (\sum_j T_j (x_{ijk} - x_{ijk'}) - t_{ik'k}) \tag{2.1}$$

subject to

$$\sum_i x_{ijk} \leq c_{jk} \qquad \forall j, k \in f_i \tag{2.2}$$

$$\sum_j T_j (x_{ijk} - x_{ijk'}) \geq t_{ik'k} \qquad \forall j, k \in f_i \tag{2.3}$$

$$\sum_i x_{ijk} = 1 \qquad \forall j, k \in f_i \tag{2.4}$$

$$\sum_j T_j x_{ijk} \geq t_{ik}^0 \qquad \forall j, k \text{ the first element in } f_i \tag{2.5}$$

$$\sum_j T_j (x_{ijk} - x_{ijk'}) \geq t_{\mathrm{ink}} \qquad \forall k, (i,n) \in g_k \tag{2.6}$$

$$x_{ijk} = 0 \text{ or } 1 \tag{2.7}$$

The objective function (2.1) is minimized for the first fix k and each pair of consecutive k', $k \in f_i$; $x_{ijk} = 0$ and $t_{ik'k} = t_{ik}^0$ when k is the first fix.

where

f_i : ordered set of fixes in the route of flight i.

C_{ik} : cost per unit time of delaying flight i at fix k.

c_{jk} : capacity at fix k' during the j^{th} time interval.

$t_{ik'k}$: minimum time required for flight i to fly from fix $k' \in f_i$ to the next fix $k \in f_i$

when k is not the first fix in f_i.

t^0_{ik} : scheduled time that flight i is to be at the first fix $k \in f_i$.

t_{ink} : minimum turnaround time between the arrival of flight i and the departure of flight n at fix k.

g_k : set of ordered pairs of flight legs (i,n) using the same airframe for which flight i terminates at fix k and flight n starts subsequently.

The objective function to be minimized is the total cost of delaying the selected flights between the selected airports during the time period of interest. When k is the first fix in the route of flight i, C_{ik} is the cost of one unit of ground delay.

2.2.2. The Revised Richetta Model

Richetta makes some important simplifications to the problem that allowed him to solve it optimality (Richetta and Odoni, 1993). The crucial idea is the attempt to treat planes arriving at the capacitated airport as a *flow* rather than as individual flights. More specifically, one can aggregate the originally scheduled arrivals at each time period t into a vector of demands D_t, $1 \leq t \leq T$, where $D_t = |f \in F : F_f = t|$. Once this is done, the problem can be formulated using the decision variable $X_{i,j}$, $i \leq j$ where $X_{i,j}$ is the number of planes originally scheduled to arrive at time i that is rescheduled to arrive at time j. $W_{q,t}$, for $1 \leq q \leq Q$ and $1 \leq t \leq T$, is the number of planes experiencing airborne delay at time period t under scenario q.

In his original formulation, Richetta uses slightly super-linear ground holding costs (i.e. slightly more than twice as expensive to ground hold planes for two periods than for one), in order to avoid solutions where some planes are ground held for a very long time while other planes land immediately. Yet it is not necessary if it is involved that all planes originally scheduled to land at time i must land before all planes originally scheduled to land at time j, for $i \leq j$. Using a linear ground holding cost function with the understanding that the constraints above can be recovered by post-processing, the following IP is proposed (Rifkin, 1998).

$$\min \sum_{i=1}^{T} \sum_{j=i+1}^{T+1} c_g (j-1) X_{i,j} + c_a (\sum_{q=1}^{Q} p_q \sum_{t=1}^{T} W_{q,t} \tag{2.8}$$

$$\sum_{j=i}^{T+1} X_{i,j} = D_i \qquad i = 1,...,T \tag{2.9}$$

$$W_{q,t} - W_{q,t-1} - \sum_{j=1}^{t} X_{j,t} - S_{q,t} = -M_{q,t} \qquad t = 1,...,T+1, \;\; q = 1,...,Q$$

$$(W_{q,0} = W_{q,T+1} = 0) \tag{2.10}$$

$$\sum_{t=1}^{T+1} S_{q,t} = \sum_{t=1}^{T} M_{q,t} \qquad q = 1,...,Q \tag{2.11}$$

$$X_{i,j} \in Z_+, W_{q,t} \in Z_+, S_{q,t} \in Z_+ \tag{2.12}$$

In the IP above, the objective function (2.8) is the sum of the (fixed) ground holding costs and the (expected) air delay costs. Constraint set (2.9) stipulates that all planes scheduled to arrive at time t must arrive between times 1 and time T+1, inclusively. Constraint set (2.10) sets the air delays $W_{q,t}$ for each scenario- in words, during a given period, under a given scenario, the number of planes being air held is at least as large as the number of planes being air held from the previous time period, plus new planes that arrive during that time period, minus the capacity at that time period under that scenario. Constraint set (2.11) is an artifact of the way in which Richetta poses the deterministic version of this problem as a network flow problem; in this formulation, it is redundant.

The key feature of this model is that by treating arrivals to the airport as a flow, associating variables with individual flights is avoided. This in turn greatly reduces the size of the problem, making solution to optimality plausible for reasonable problem sizes. Additionally, empirically, Richetta observes that the LP relaxation of the model always yields integer solutions.

3. METHODOLOGY

3.1. Overview of Airline Schedule Perturbation Problem

Airlines expend tremendous effort in designing their flight schedules. In these expensive and dedicated processes, monetary, intellectual, and computational resources are put together to accommodate various factors for efficient utilization of each scarce resource and to take advantage of consumer demands. The scheduling process is complex due to such factors as market demand, fleet size, available resources, governmental regulations on aircraft maintenance requirements, and union contracts on crew work hours, etc. which must be accounted for (Luo and Yu, 1998).

Operations research plays an important role in this challenging environment. A key feature of products generated by operations researchers, in this case a flight schedule, is that resources are tightly coupled together, and even minor perturbations could have severe impact on smooth and successful execution of the established schedule. Perturbations could be attributed to various factors such as:

- air traffic congestion,
- severe weather conditions,
- airport congestion,
- mechanical problems,
- crew lateness/absence,
- tightly coupled schedule and
- propagation effects.

Traditionally and currently, airlines rely upon Operations Coordinators (OC) with access on-line operational data to manage the day-to-day situation as perturbation occurs. After analyzing the impact of the perturbation, the OC provides what s/he sees as the most appropriate response. The responses usually take the form of

- delaying,
- canceling, and/or,
- creating new flights.

Table 3.1. An Example of GDP, (at DFW airport, on Nov. 8., 1993)

Flight #	Origin	EDT	CTD	ETA	CTA	Delay
AAL1836	MCI	S2349	C0039	100	A0150	50
AAL1148	MSP	S2306	C2359	112	A0205	53
AAL1527	CLT	S2348	C0030	205	A0247	42
AAL515	CMH	S2345	C0027	208	A0250	42
AAL428	LAX	S0018	C0059	233	A0314	41
AAL639	OKC	S0205	C0254	240	A0329	49
AAL2257	ORD	S0219	C0248	417	A0446	29
AAL2359	ORD	S0249	C0249	449	A0449	O

These responses must be quickly generated. Because of the large volume of flights that need to be considered and the need to respond in real time, it is difficult to decide among many others in airline operations. Lack of robustness in a flight schedule is unfortunate, because perturbations do occur during execution of the schedule.

In the macro level, the remedial programs for these kind of perturbations are managed through various programs, most popularly, Air Traffic Control (ATC), Air Traffic Flow Management (ATFM) and Ground Delay Program (GDP). For example, ATFM decisions are constrained by their responsibility to endure that traffic flow levels do not exceed established capacity thresholds, enabling air traffic controllers to ensure safe aircraft separation.

Simply, all the approaches / programs above are to get the perturbed schedule to its established form, to yield efficient and equitable use of scarce airspace and airport capacity. All try to re-establish the balance of the air system.

Table 3.1. above depicts a real GDP instance from DFW airport, USA. EDT, CTD, ETA, and CTA refer to Expected (Scheduled) Departure Time, Controlled Time of Departure, Expected Time of Arrival and Controlled Time of Arrival. To read, the first row of the table indicates that flight AAL1836 originating from MCI was scheduled to arrive at 1:00 pm but was given a slot 1:50 pm. To note, this is an extracted exemplary data, only 8 schedules of the 71 ones in that day's actual data. And also note that the schedule's main ordering characteristic is on ETA.

3.1.1. Possible Objectives to Absorb Perturbations

Many objectives are legitimate in the context of managing airline schedule perturbation some of which are listed here with their appropriateness:

1. Minimizing the number of cancellations: Due to the profound effect of canceling a flight (such as the burden of putting passengers on other flights, creating a lack of resources in down-line stations, etc.), an airline's first priority is to keep the number of cancellations to a minimum.
2. Minimizing the number of delayed flights: Dependability statistics of an airline is one of the most visible performance measures. It is calculated as the number of flights arriving within 15 minutes of their scheduled arrival times, divided by the total number of arrivals. Canceled flights are not considered for dependability statistics[1]. By minimizing the number of flights delayed more than 15 minutes, the dependability statistics is optimized.
3. Minimizing maximum delay: To ensure that no flights will be delayed to an unreasonable length when optimizing other criteria, the maximum delay among all affected flights is to be minimized.
4. Minimizing total passenger delay: This objective measures the total impact on all passengers caused by schedule perturbation.
5. Minimizing total costs of delays: This objective is similar to the last one, except that the cost of the delay is substituted for the passenger delay for each flight which may be a nonlinear function of delay length and number of passengers.

[1] It is estimated that every percentage point of dependability increase generates an extra $17 million in revenue for American Airlines. (ATAA, 1992)

3.2. The Ground Holding Policy Problem (GHPP)

GHPP is a specific schedule perturbation problem and tries to minimize delay costs when a reduced arrival capacity is forecasted at the destination airport. The coming sections delve with this problem in detail.

3.2.1. Overview of Recent Studies in GHPP

The study (publicized) list on GHPP is actually, by no means exhaustive. Odoni (1987) is the first to describe GHPP systematically. In (Andreatta and Romanin-Jacur, 1987), the stochastic version of GHPP is treated but for a single airport with constrained arrival capacity in at most one time period. In (Terrab and Odoni, 1993) a dynamic programming formulation of stochastic GHPP is developed with some heuristics to handle the larger cases. Using stochastic linear programming with recourse, Richetta and Odoni (1993) expend this work to include the dynamic case, in which ground holdings are updated as time progress. This paper deals with stochasticity by assuming that a probabilistic distribution of 'scenarios', or possible realizations of capacity, is known. By treating arrivals as flows rather than as individual flights, and making use of the fact that empirically, the linear programming relaxation always yields integer solutions, the model defined in this paper is practically solvable for reasonable problem instances. Although their dynamic solution yields considerable savings over the static solution; the speed of solution proves to be too slow for realistic cases. The GHPP is explored with banking constraints in (Hoffman, 1997). In this study, a particularly strong, facet-inducing model of the banking constraints is presented which allows one to solve large instances of ground holding problem with banking constraints (GHB) in less than half-an-hour of CPU time. A substantial simplification of the Richetta model and a new model that solves a closely related problem, the Maximum Air Delay model are introduced in (Rifkin, 1998).

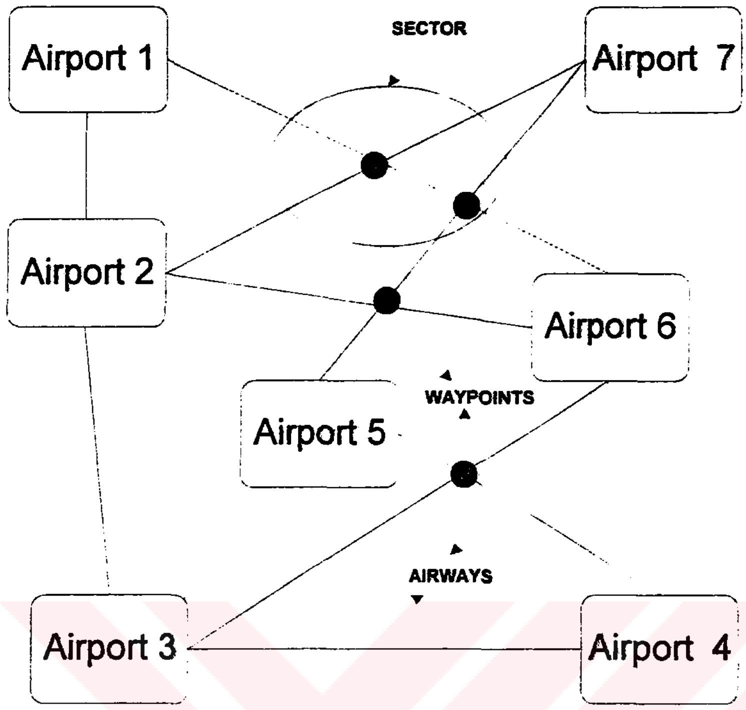

Figure 3.1. Configuration of ATC Network (Terrab, 1990)

3.2.2. Nature of GHPP in ATC System

To describe the ATFMP, a basic representation of an Air Traffic Control (ATC) network is figured as above. There are four types of ATC elements in the figure as shown that are:

- Arcs in the network represent airway elements. These correspond to the physical paths aircraft use.
- Airport elements are the sources and sinks for air traffic in the system; nodes in the network represent them.
- Nodes in the network also represent waypoint elements. They correspond to points in the network where airways intersect.
- Sector elements are defined to be a set of waypoints and segments of airways that is treated as a unit for ATC purposes.

Congestion can occur at any one of the elements of this network when the capacity of these elements is reduced. If the capacity of each one of the elements were known and invariant of time, there might be no delays in ATC system. Yet the uncertainty of the physical conditions of the system like the weather, these capacities are subject to influenced changes. Airports constitute the principal bottlenecks of the ATC system. Weather conditions (visibility, precipitation, wind, cloud ceiling, etc.) determine the capacity of a given airport (in terms of the number of landings and take-offs that can occur during a time interval) since they affect minimum landing separation rules and can dictate the runway configuration to be used. Moreover, the reductions in airport capacity can be as high as %50 in some extreme weather conditions (Terrab, 1990). Thus there is a need to adjust the aircraft flows through the various elements of the ATC network on a short-term basis using available information and forecasts concerning the capacities of the elements of the ATC network. It is noteworthy to mention that if an incoming flight is delayed due to air traffic congestion or weather conditions at the station where the flight originated, and the aircraft has been scheduled to take a flight departing from this station, it is likely that the second flight will be delayed. This delay may propagate through the network.

The TFMP can then be defined as the problem of adjusting flows in the ATC network so as to minimize the cost of delays given the available information pertinent to the present and future status of the ATC network elements. Odoni (1987) classifies two types of such flow management actions as “tactical” and “strategic”. Tactical actions are to be exercised when the aircraft is already airborne and include:

- High altitude holdings, path stretching maneuvers, or modifying en route flight plans in order to avoid more costly low altitude delays.
- The control of en route speeds to time the interval of an aircraft at a congested point of the ATC network.
- The sequencing of aircraft for landing to maximize runway acceptance rate.

Strategic actions include:

- The modification of flight plans of some flights before take-off in order to bypass congested areas of the network.
- Delaying the departure time of some flights. These delays are referred as "ground holds" or "gate holds" and correspond to delaying the actual departure time of an aircraft beyond its scheduled take-off time. These delays are to be taken before the aircraft starts its engines on the apron area (either at the gate or in a parking area) even if the aircraft is otherwise ready to taxi to the runway.

As can be observed, the tactical actions are clearly microscopic in nature and can help control the flow of aircraft through specific areas of the ATC network but are limited in the extent of control they permit by the very fact that the aircraft is airborne and is therefore subject to fuel and safety constraints. On the other hand, the strategic actions have a macroscopic nature with greater potential for regulating aircraft flows. However, the first type of strategic action above has a limited potential since there are plain limits to the scope of modifications of flight plans like fuel constraints which actually brings this type of action out to a "fine tuning" nature. The second type of strategic action, on the other hand, provides greater flexibility in flow management adjustments and can lead to greater savings in delay costs for the following two reasons:

- Because of the increased ability in control and regulation of the aircraft flows with respect to the exclusion of some constraints (e.g. fuel constraint) of airborne control.
- Because important savings in delay costs besides the improved safety can be expected if some of the ground delays are absorbed. Savings in ground holding delays are for sure less costly than of the airborne ones that comprise fuel consumption as well as depreciation and maintenance costs. Also ground holds are preferable for safety considerations. Since, in general, it is not possible to predict airport capacities exactly at the times of departure of the aircraft, the basic trade-off is between the cost of airborne delays that result from *optimistic* strategies, which impose little

ground holds, and the costs of ground holds from *pessimistic* strategies, which impose excessive ground holds and may result in unused capacity.

Ground Holding strategy should thus be considered as the essential component of ATFM system and it should nevertheless be complemented by the tactical actions aforementioned but those are only of a fine tuning nature. It is also important to note that the ATFMP is not a short-term problem in the sense that it is to be addressed only until the longer term issues are resolved. It, in fact, is to be considered a permanent part of an efficient ATC system because TFMP is already motivated by the fact that the capacity conditions in the ATC system can change over time and may be difficult to predict yet may be ameliorated if significant improvement in weather prediction technology is attained.

As to sum up, the Ground Holding Policy Problem (GHPP) constitutes the focus of an efficient ATC system and it should be kept in mind that the boundaries between tactical versus strategic actions are not clearly defined in practical situations. There are some obvious interdependencies such as, for example, the fact that airport capacity depends not only on weather conditions but also on sequencing of landing aircraft. A complete flow management system has to deal with interfacing these various approaches.

3.2.3 General GHPP Description

The GHPP is concerned with timing the arrival of aircraft coming from several origin airports into a single arrival airport (actually a congested one), Airport Z and is described, referring to Figure 3.2, with the problem characteristics, as follows:

1) Arrival operations at a given airport Z (for which some amount of congestion is expected) during a time interval $[0,T]$ are considered.

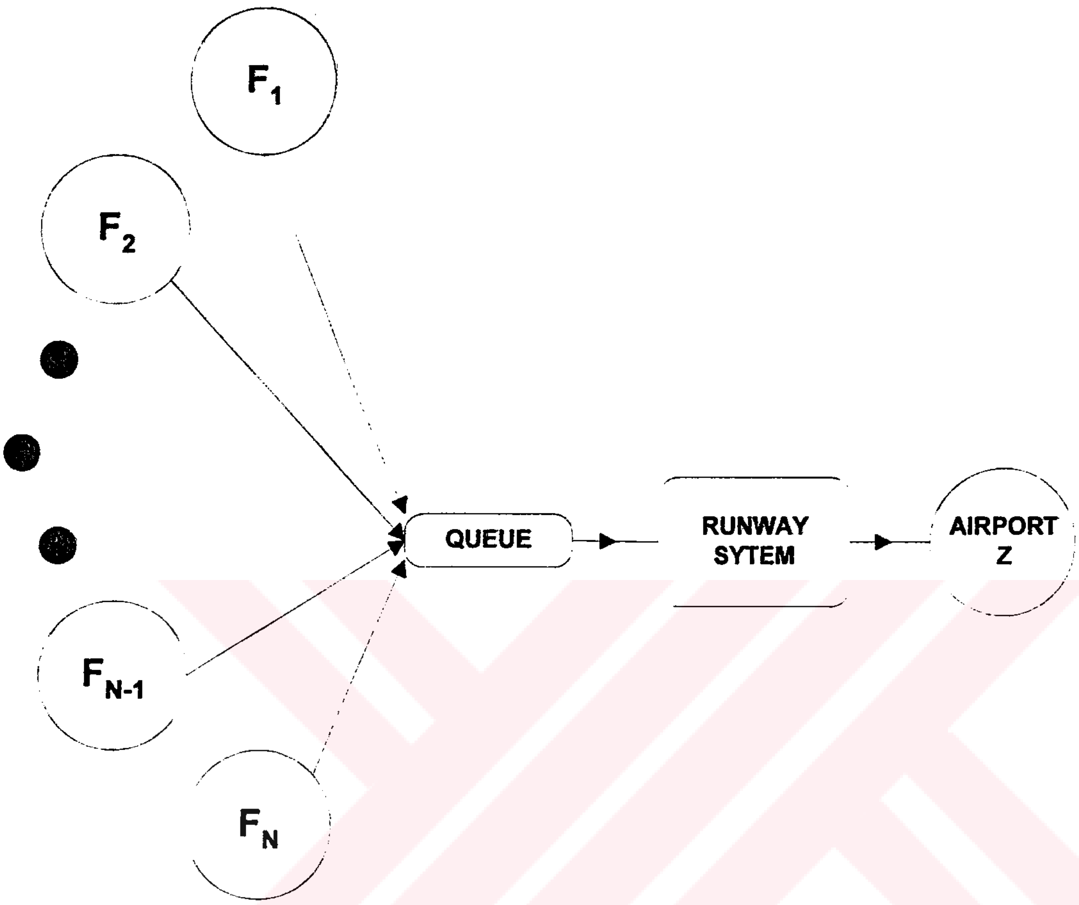

Figure 3.2. Network Representation of GHPP

2) On hand, there is a complete list F_1, F_2, ...F_N of flights departing from a number of origin airports and scheduled to land at airport Z during $[0,T]$. For each flight F_i, the scheduled departure and arrival times of F_i are known. Furthermore it is assumed that the travel times are deterministic and known in advance.
3) For each $t_1 \in [0, T]$ and any $t_2 \geq t_1$ a probabilistic forecast for $C_p\ (t_2 \mid t_1)$, the arrival capacity of airport Z at time t_2 is given.

The GHPP is then defined as the problem of finding, for each flight F_i, the optimal amount of ground hold to be imposed on flight F_i so that the overall expected total delay cost (ground + airborne queuing delays) is minimized. If the queuing delays were predictable, one would take them entirely as ground delays before departure so as to minimize costs. Because of the uncertainty concerning the capacity of airport Z during [0,*T*], however, the amount of delays that each flight would incur if it left on time cannot be predicted. The GHPP is therefore concerned with finding the amount of ground delay to impose on each flight so as to strike an optimal balance between the costs of these known ground delays and future unknown airborne delay costs based on available information concerning the capacity of airport Z. The most general version of the GHPP has thus the following characteristics:

a) It is a stochastic problem. There are many circumstances for which it is not possible to exactly predict the capacity of an airport even a few hours in advance. The reason is that capacity depends strongly on local weather conditions, which are subject to high uncertainty. In these circumstances one may have to settle for a probabilistic capacity forecast. The object of the GHPP is then to try to reach an optimal balance between *present-known* ground delays and *future-expected* airborne delays.

b) It is a dynamic problem. For any time t_2, an assumption is needed to make a decision to access to a whole family of random variables (a stochastic process) $[C_p\,(t_2 \mid t_1)\colon t_2 \geq t_1]$. This formulation allows that, for any specific t_2, the forecast $C_p\,(t_2 \mid t_1)$ can change with t_1.

c) It is a combinatorial problem. Since the individual delay costs for aircraft can be very different, it is not enough to ponder this problem solely in terms of controlling the flow of aircraft arriving at airport Z but one has to determine the optimal composition of these flows. The combinatorial aspect is evident, regarding the case for the deterministic capacity case; one can consider this situation as one for which the stochastic and dynamic aspects have been resolved but it is obvious that large cost-savings can be expected from assigning individual aircraft to available capacity in an optimal manner.

The probabilistic and dynamic aspects of GHPP are difficult to deal with. At first the solution methods are developed for the deterministic version which is a purely combinatorial problem, and then the static probabilistic version of GHPP is focused which consists of determining ground holds based on a unique probabilistic forecast for the airport capacity. Thus solution methods for the dynamic probabilistic version of the GHPP are derived from the analysis of the static version and can be classified as follows:

- The deterministic GHPP complies with the assumption that the capacities K_1, K_2, ...K_P are fixed and known. The GHPP is then concerned with assigning each flight F_i to a time period T_j so as to minimize a total ground delay cost (and not to violate the capacity constraints).
- The static-probabilistic GHPP assumes that the capacities K_1, K_2, ...K_P are random variables and that, at some time t before the earliest departure time of any of the flights F_1, F_2, ...F_N, a probability distribution $P_{K_1,K_2,\dots K_P}(t)$ that represents a forecast for these capacities is given and that is *assumed to remain fixed after t*.
- The dynamic-probabilistic GHPP also assumes the capacities to be random variables yet moreover it is assumed that one has a model for the evolution of the probabilistic distribution $P_{K_1,K_2,\dots K_P}(t)$ with time; i.e., it is assumed that a family of distributions $P_{K_1,K_2,\dots K_P}(u)$ for all times after the earliest departure time is obtained.

Onwards the deterministic model will simply be touched without the scope.

3.2.3.1. The Deterministic Case

The deterministic version of GHPP (will be abbreviated as DGHP onwards) assumes that the future capacity at airport Z can be predicted *exactly*. This assumption reduces the GHPP to a purely combinatorial problem for which standard solution methods are available. One of the motivations for looking at the deterministic case is that there are cases for which it is reasonable to assume that capacities can be forecasted with little error. This is the case for airports located in areas where there is little variability in weather conditions or for which changes in weather conditions are predictable and weather patterns remain stable for a long time period once established.

As to formulate the DGHP, consider arrival operations at a given destination airport, say Z, during time interval [0, *T*]. The interval [0, *T*] is subdivided into several consecutive time periods $T_1, T_2, \ldots, T_P$ with corresponding fixed deterministic capacities $K_1, K_2, \ldots K_P$. N flights $F_1, F_2, \ldots F_N$ are scheduled to land at airport Z during these time periods. The objective function can be stated as minimizing the total ground delay as

$$TC = \sum_{i=1}^{N} Cg_i(X_i) \tag{3.1}$$

with X_i being the number of time periods flight F_i is delayed on the ground. The underlying assumptions in this approach are simply:

a) For each flight F_i , the flight's scheduled landing period and the cost of delaying F_i ($Cg_i(t)$) t time periods on the ground before take-off ıs known.
b) It is assumed that all flights that were not able to land during one period of the time periods $T_1, T_2, \ldots, T_P$ can do so during a final time period T_{P+1} with $K_{P+1}=\infty$.
c) Imposing a ground delay of X_i on flight F_i will make that flight arrive at airport Z during time period T_{Pi+Xi}. This requires that the travel times be deterministic and the airport Z be the only congested element of the air traffic network. (i.e. no delays can occur at the origin airports or en route to airport Z).
d) It can be determined in advance how available capacity at airport Z will be allocated between arrivals and departures.

The final DGHP model can be formulated as follows:

$$\min \sum_{i=1}^{N} \sum_{j=P_i}^{P+1} C_{ij} X_{ij} \tag{3.2}$$

subject to :

$$\forall (i,j): X_{ij} = 0 \text{ or } 1 \tag{3.3}$$

$$\forall i \ \ (=1, N): \sum_{j=P_i}^{P+1} X_{ij} = 1 \tag{3.4}$$

$$\forall j \ \ (=1, P): \sum_{i=1}^{N} X_{ij} \le K_j \tag{3.5}$$

where X_{ij} is defined as the assignment variable as $X_{ij} = 1$ if flight F_i is assigned to land during period T_j and $X_{ij} = 0$ otherwise. The X_{ij}'s are only defined for $j \ge P_i$. C_{ij} is the quantity $C_{gi}(j-P_i)$, the cost of assigning flight F_i to land during time period T_j for $j \ge P_i$.

The constraint matrix of this IP is totally-unimodular and therefore the relaxation of the integrality conditions may be used to solve with the simplex method. (see (Terrab, 1990) for further details).

3.3. Some Background for Integer Programming

An integer program (IP) is a problem in which one wishes to optimize a linear functional $f(x)$ subject to the conditions that

(i) the vector x satisfies a system of linear inequalities $Ax \le b$ (or $Ax \ge b$),

(ii) each component of x be non-negative and

(iii) each component of x be integer.

If this last condition is relaxed, then one obtains a linear program (LP), which is called the LP relaxation of IP. Consider the LP below.

$$min \ \{f(x): \ Ax \ge b \ and \ x \ge 0\} \tag{3.6}$$

Let P be the polyhedron defined by the system $Ax \ge b, x \ge 0$. Using the fact that P is convex, (3.6) can be solved very efficiently by algorithms such as the simplex method.

If the constraint that x must be integral is added to (3.6), then one obtains an IP with a set F of feasible points. F is a discrete set of lattice points within P but, unless

F consists of a single point or no points at all, F is not convex. This renders useless many traditional optimization procedures, thereby, integer programming is, in general, more difficult than linear programming.

In any practical applications of an IP, one can find some upper bound on the integer values that can be assumed by the variables. Therefore, there is always an algorithm that will find the optimal value to the bounded IP in a finite number of steps: simply enumerate all the feasible points and select the one with the optimal function value. However, this is highly impractical for all but the smallest problems. Consider that a problem with just 100 binary variables could have 2^{100} feasible solutions!

The algorithms employed by the commercial solvers for solving IP or MIPs are based on a branch-and-bound (B&B) strategy. In B&B, the set of feasible points is enumerated implicitly by exploring the tree of solutions in which each branch represents a restriction of the set of feasible points. At each node, a solution is evaluated and a decision is made as to whether or not the optimal solution could lie in the subtree below the current node. The pruning criteria are based on the solution to the LP relaxation that is obtained after imposing the restrictions implied by the current path through the B&B tree. When minimizing, for instance, if the LP relaxation value is higher than the current best integer value, then the optimal solution could not possibly lie in the subtree and the subtree is pruned. In practice, the success of any B&B algorithm lies in its ability to do a great deal of pruning.

When solving an IP, relaxing the integer constraint on some of the variables can greatly reduce the number of integer solutions that need to be explored. Often, much can be done to simplify an IP after it is formulated but before it is solved. It might be possible to fix some of the variables a priori or even eliminate variables altogether. Bounds can be examined for potential tightening and constraints can be examined for elimination. Any such steps fall under the category of *preprocessing*, which eases greatly the computational complexity of an IP.

Every polytope (bounded polyhedron) P has a minimal description meaning that, up to multiplication by constants, there is unique set of linear inequalities that describe it. Given an inequality $\pi x \geq b$ or ($\pi x \leq b$), the set of points in P that satisfy

π at equality is called a *face* of P. A *facet* of P is a face of dimension n-1, where n is the dimension of P. If π is one of the inequalities necessary to the description of P, then the face defined by π is a facet of P. Since the convex hull of integer points is feasible to an IP is a polyhedron, it has minimal description.

A major technique in integer programming is to produce a formulation in which as many as possible of the inequalities represent the facets of the convex hull of integer solutions, P_C. In the best of circumstances, the solution to the IP can be obtained directly from its LP relaxation. This is possible whenever each of facets of P_C is represented by at least one of the inequalities. Since a minimal description of P_C is rarely known, a greater number of constraints might be preferable to a lesser number so that P_{LP} (P_{LP} be the set of points feasible to the LP relaxation of the integer program, IP)is as close to P_C as possible; this increases the chances of solving the problem quickly.

If an integer point is feasible to IP, then it is feasible to the LP relaxation and, therefore, P_C is contained in P_{LP}. If P_{LP} fits tightly around P_C, then the formulation at hand is said to be a *strong formulation*. Let IP^* be an equivalent formulation of IP, meaning that the set of integer points feasible to IP^* is the same as those integer points feasible to IP. Let P^*_{LP} be the set of points feasible to the LP relaxation of IP^*. If P^*_{LP} is contained in P_{LP}, then $P_C \subseteq P^*_{LP} \subseteq P_{LP}$ and IP^* is said to be a stronger formulation than P_{LP}. If IP^* is the strongest formulation possible if $P_{LP} = P_C$.

The ability to solve an integer program can vary enormously with the choice of the formulation. Therefore, it is important to compare both analytically and empirically a number of alternative formulations and choose the strongest one. A valuable experimental metric used for determining the stronger of the two models is *value gap*, meaning that the difference between the LP relaxation optimal function value and the optimal integer function value. A lower value gap generally indicates a stronger model.

In the event that each of the corner points of (non-empty) P_{LP} is integral, there will always be an integral optimal solution x^* to the LP relaxation. Such a polyhedron is called *integral polyhedron*. Every integer point feasible to IP is contained in P_{LP}. Since x^* yields the best function value of all points in P_{LP}, in particular, it yields the best function value of all integer points in P_{LP}, so x^* must be the optimal integer solution. The ideal formulation of an IP is one for which the LP relaxation is integral for, then, the IP can be solved by applying an LP optimization algorithm such as the simplex method to the LP relaxation of the IP.

Let $Ax \geq b$ be the linear system that describes the set P_{LP}. One way to ensure that P_{LP} is integral is to show that the matrix A is *totally unimodular* (TU) meaning that every square submatrix of A has determinant 0, 1, or -1.

4. THE STOCHASTIC STATIC APPROACH

4.1. The Stochastic Static Single-Airport Ground Holding Problem (SSGHP)

In Chapter 3, it is shown that GHP can be formed for a single airport given the schedule of incoming flights and a deterministic arrival acceptance rate, or simply the capacity for each time period in the planning horizon. It is obvious that when the stochasticity in future capacities is enforced to the model, things get more involved. In this chapter, Stochastic Static Single-Airport Ground Holding Problem (SSGHP) is introduced: it is 'static' because it requires all decisions over a given time horizon to be made in advance; it is 'stochastic' because it explicitly takes the stochastic nature of future capacity into account. Finally it is 'single-airport', as the name suggests, because all decisions are accumulated at a single airport rather than multiples. A detailed study on the multiple-airport case can be found in (Navazio and Romanin-Jacur, 1998). The effectiveness of GHP is totally dependent upon the ability of the specialist who formulates the GHP to predict the AAR's (Arrival Acceptance Rate) for each of the time periods.

To illustrate, consider a scenario in which a storm is predicted to hit an airport, say airport Z, at 12:00 hours and in response, a GHPP has been implemented based on the prediction that the arrival capacity for first hour will drop from 60 to 40 flights per hour. Then the GDP would assign to each of the flights scheduled to arrive during the first hour a ground delay sufficient to ensure that it will arrive without airborne delay. But if the storm is delayed by several hours or bypasses the airport altogether, then many of these flights will have incurred unnecessary delay. For instance, a flight that was scheduled to leave another airport, say airport Y, at 8:00 and arrive at 13:00 may have been held at its departure airport for one hour so that it would not arrive until after 14:00. Aside from a slight adjustment in en route air speed, there is no way for this flight to offset its one-hour delay even though, in hindsight, it could have landed on time. In this event, the GDP has assigned too much ground holding.

There is also the opposite scenario in which the arrival capacity of an airport is overestimated and flights incur airborne delay that could have been absorbed on the ground, had a more aggressive GDP been enforced. In this instance, the GDP has not assigned enough ground delay.

Almost always there will be some uncertainty in the prediction of the arrival capacity at a given airport. AAR's are dependent upon airport configurations, which are, in turn, dependent upon meteorological conditions such as visibility, wind velocity/direction, and precipitation. Thus, arrival acceptance rates are stochastic in nature, rather than deterministic.

The SSGHP can be improved substantially under a first-come first-served (FCFS) discipline i.e. large amounts of structural information can be recovered via post-processing. (Rifkin, 1998) introduces a new model for the static stochastic case as follows:

$$\min \sum_{t=1}^{T} c_g G_t + \sum_{q=1}^{Q} \sum_{t=1}^{T} c_a p_q W_{q,t} \tag{4.1}$$

$$\sum_{t=1}^{j} A_t \le \sum_{t=1}^{j} D_t \qquad j = 1,\ldots,T \tag{4.2}$$

$$\sum_{t=1}^{T+1} A_t = \sum_{t=1}^{T+1} D_t \tag{4.3}$$

$$W_{q,t} - W_{q,t-1} - A_t \ge -M_{q,t} \qquad t = 1,\ldots,T \quad q = 1,\ldots,Q \quad (W_{q,0} = 0) \tag{4.4}$$

$$G_t + \sum_{t=1}^{j} A_t = \sum_{t=1}^{j} D_t \qquad j = 1,\ldots,T \tag{4.5}$$

$$A_t \in Z_+, W_{q,t} \in Z_+, G_t \in Z_+ \tag{4.6}$$

The decision variables in the model above is A_t, $1 \le t \le T$, is simply the number of planes allowed to arrive in the airspace of the capacitated airport at time t. The objective function (4.1) is the sum of the (fixed) ground costs and the (expected) air delay costs. Constraint set (4.2) stipulates that no plane may arrive before it is

Table 4.1. Complexity Size Comparison Of RM and SSM

	Richetta Model (RM)	Static Stochastic Model (SSM)
Rows	$QT+2Q+T$	$QT+2T+1$
Columns	$\frac{(T+1)(T+2)}{2}+Q(2T+1)$	$2QT+3T-Q+1$
Non-zero Elements	$(T+1)((Q+1)(\frac{T+2}{2})+4Q)-2Q$	$T^2+4T+4QT-Q+1$

originally scheduled to do so, while constraint set (4.3) stipulates that all planes must arrive by time T+1. Constraint set (4.4) sets the air delays $W_{q,t}$ for each scenario. The G_t (auxiliary) variables represent arrival time adjustments based on planned ground delay whereas $M_{q,t}$ is the arrival capacity of the airport during time t if scenario q were realized with probability p_q. And c_g stands for the unit ground delay cost and c_a does so for air delay cost.

Rifkin (1998) compares the Richetta Model (RM), which is the seminal model for the last ATFMP models, with SSGHP in terms of model size and simplicity. The result is tabulated as follows, given Q scenarios and T periods.

To clarify, for 10 scenarios and 30 time periods, the Richetta model has 350 rows, 1106 columns and 6676 non-zero elements, while the SSM has 361 rows, 681 columns and 2211 non-zero elements. Since these models are solved as LP relaxations, the number of non-zero elements is the most important measure of the size of a model. However, it is observed that SSM is solved three times faster than RM with CPLEX.

4.2. A New SSGHP Model

Getting the Rifkin's model as a seminal one, a new SSGHP model is developed which is as follows (SSGHP*):

This model is a substantial simplification of Rifkin's model. The only cost factor is c_r which is c_a/c_g. The decision variables and parameters are:

T : number of time periods

S : number of scenarios

A_t : the number of flights to land during t in the absence of airborne delays.

G_t : number of flights whose arrival time is adjusted from t to t+1 (or later) imposing ground delay at their point of origin

$W_{s,t}$: number of flights air held from t to t+1 (or later) under scenario s.

$M_{s,t}$: the arrival capacity of the destination airport at t if scenario s is realized.

D_t : the number of flights predicted to arrive at the airport in t, if there were no capacity restrictions.

$$\min \sum_{t=1}^{T} G_t + \sum_{s=1}^{S} \sum_{t=1}^{T} c_r p_s W_{s,t} \tag{4.1}$$

$$A_t - G_{t-1} + G_t = D_t \qquad t = 1,2,\ldots T+1 \tag{4.2}$$

$$(G_0 = G_{T+1} = 0) \tag{4.3}$$

$$-W_{s,t-1} + W_{s,t} - A_t \geq -M_{s,t} \qquad t = 1,2,\ldots T+1 \tag{4.4}$$

$$s = 1,2,\ldots S \tag{4.5}$$

$$(W_{s,0} = W_{s,T+1} = 0) \tag{4.6}$$

$$A_t \in Z_+, W_{s,t} \in Z_+, G_t \in Z_+ \tag{4.7}$$

The objective function (4.1) is the sum of the (fixed) ground delay and the expected airborne delay costs. Note that there is no cost constant associated with G_t since already the ground delay cost is normalized (i.e. c_g = 1 unit cost) Constraint set (4.2) says that all flights originally wishing to land in the current time period (D_t) or whose scheduled arrival time has been pushed by the previous time period (G_{t-1}) must be scheduled to arrive either in the current time period (A_t) or later (G_t). Constraint set (4.3) puts that $G_t = 0$ at times $t = 0$ and $t = T$+1. Constraint set (4.4) stipulates that, under scenario s, all flights scheduled to arrive in the current time period (A_t) or air delayed from the previous time period ($W_{s,t-1}$) must be air delayed until a later time period ($W_{s,t}$) or allowed to land. This inequality is necessary (rather than equality) because there may not be enough planes available to fill up the airport capacity. Unused capacity is represented by a slack variable. Constraint set (4.5) is again an initialization constraint. (4.6) imposes integrality on all of the decision variables. It is trivial to say that the probabilities of scenarios should sum up to 1.

Onwards in this thesis, SSGHP* is used as the driving model.

To clarify the model, a visualization of small size of SSGHP is presented here. Assume that D_1 = 15 and the airport can land only 10 flights at time t = 1 (i.e. $M_{1,1}$ =10), then 5 of the 15 flights scheduled to arrive at time t =1 will be held in the air by the controllers at the destination airport. Assume that D_2 = 8 and the capacity of the airport is, say 14 for the next period so that the five airborne-held flights are able to land in the next time interval. Then the airborne delay under this single AAR (Arrival Acceptance Rate) scenario with two Planned Acceptance Rate (PAR) values can be depicted as the flow diagram in Figure 4.1. Note that there is 1 more flight that can land at t = 2 (i.e. $M_{1,2} - A_2$ = 14-13 =1).

Already it is known that ground holding is cheaper than airborne holding (else there is no need for GDP!), it would have been cheaper to hold 5 flights (refer to Fig. 4.1.) on the ground for one unit time than of imposing them airborne delay. Adding a horizontal arc at the top of Fig. 4.1 can represent thus imposing ground delay to those flights as shown in Fig. 4.2.

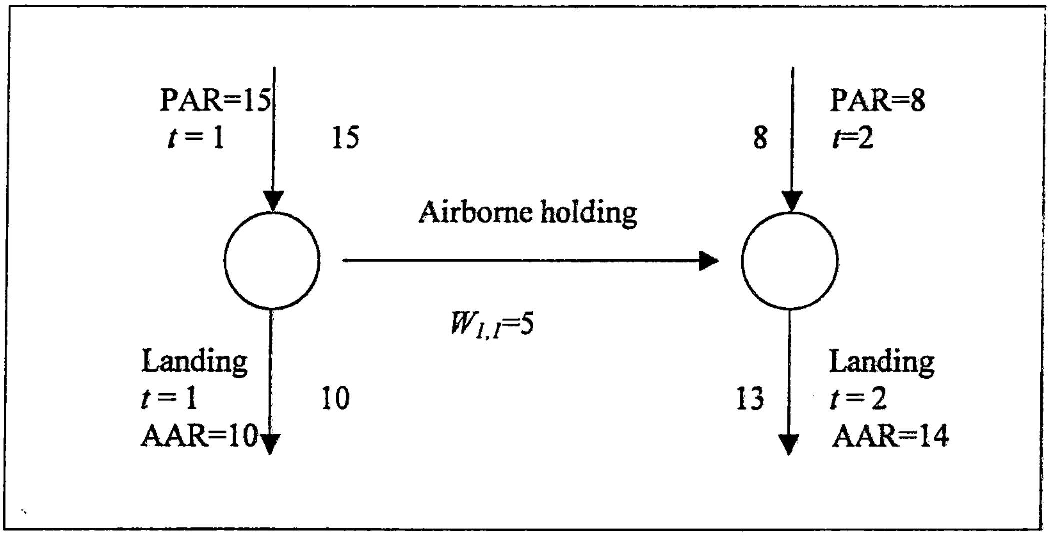

Figure 4.1. Network Representation of Example Without Ground Holding

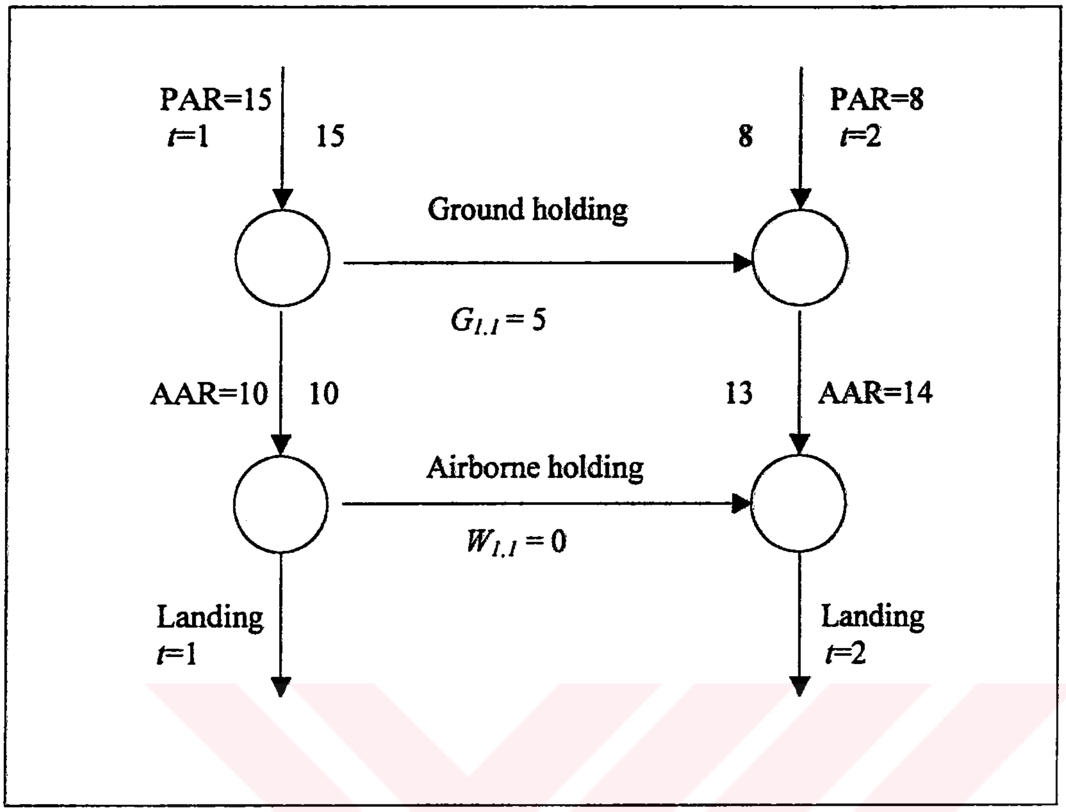

Figure 4.2. Ground Holding of 5 Flights

Note that the only change is that those five flights are subjected to GHP rather than airborne delay, yielding lesser cost and safer conditions. To note, this is a deterministic case in which there is only one scenario. If the AAR were different than the one given, this might cause a waste ground hold if for example say, AAR were actually 20 for t=1, in which there is no need for GDP!

In the real life situation, of course, the problem aforementioned is not so easy. Including the possible scenarios and the time horizon, the problem is complicated by means of computation. To visualize, an illustration (a network representation) of SSGHP for $S = 1$ and $S = 2$ with T+1 time periods is depicted in Fig. 4.3.

It should be noted that in SSGHP model, constraints (4.2) and (4.4) have the structure of flow conservation constraints. Thus this model is almost a network flow problem. Yet, the presence of a common in-flow variable, A_t, in each of S in constraint (4.4) destroys the network structure except in the case of S =1. Figure 4.3

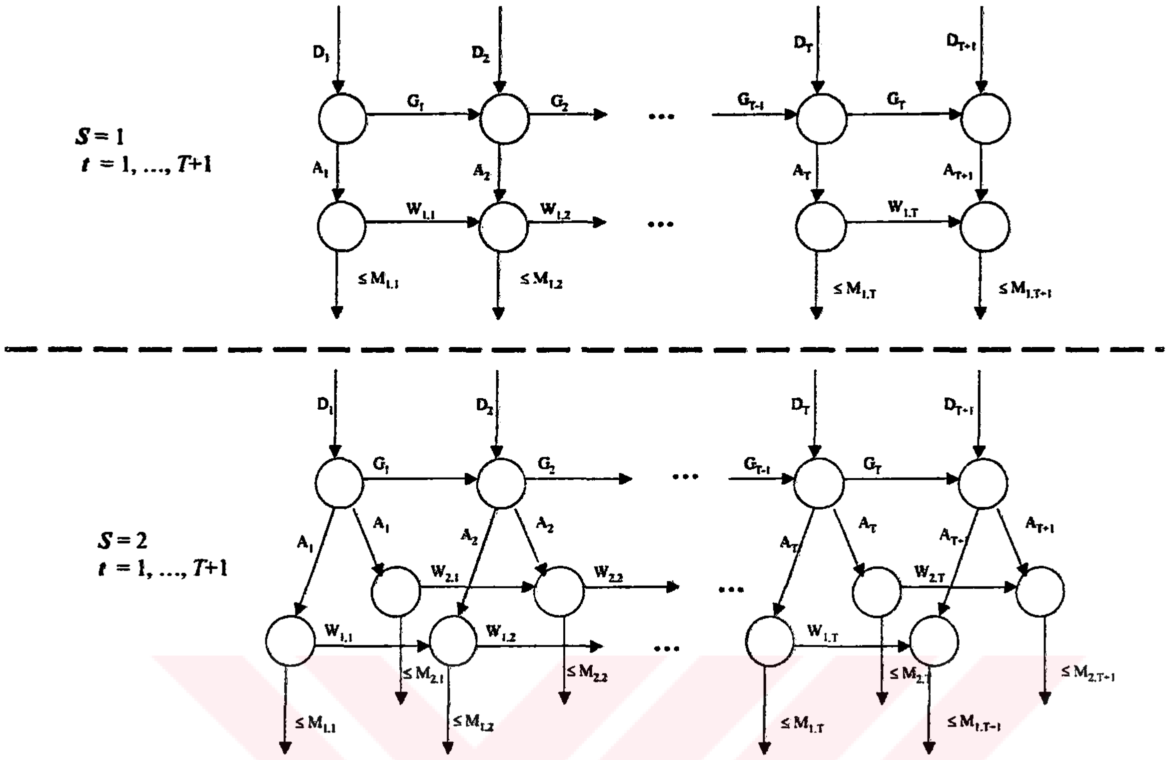

Figure 4.3. Network Illustration of SSGHP for S = 1 and S = 2.

reveals that SSGHP is a network flow problem for S =1 (which, actually, is the deterministic case) and it is not for $S \geq 2$.

4.3. An Intuitive Approach to SSGHP

There are many approaches to stochastic programming. In this study, it is adopted with respect to discretized time horizon and the consideration of arrival capacities as random variables with a discrete probability distribution. This is consistent with the manner in which airports operate (Hoffman, 1997). An intuitive approach, which may strengthen to understand the motivation behind the operating system of SSGHP, is given on an example.

EXAMPLE 1 : Consider a fictitious airport whose normal AAR is 80 flights per hour. Figure 4.4 displays again fictitious possible multiple AAR scenario forecasts. There are 3 scenarios and 6 time periods.

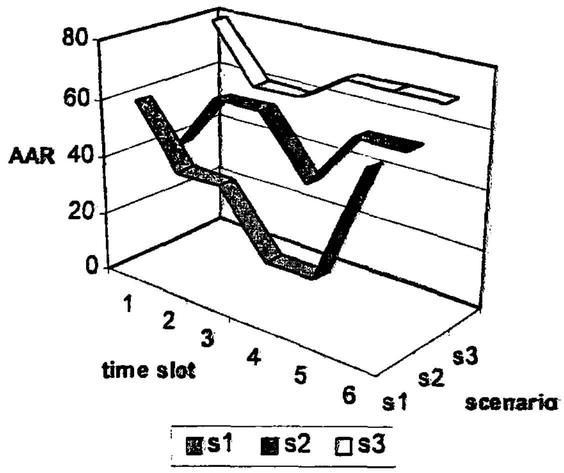

Figure 4.4. Multiple AAR scenarios

The pessimistic forecast, scenario 1 (s_1), predicts that the AAR will plummet to 40 and then 20 flights per unit time slot (say 1 hour) and will not recover until the 6^{th} hour. Scenario 2 (s_2) is similar to Scenario 1 in that both indicate similar trends between 2^{nd} and 5^{th} hours. Yet s_2 signals that (say for a stormy weather) the weather is less severe. Scenario 3 (s_3) is an optimistic one predicting at least 60 flights per hour.

Now suppose that s_2 is to occur with a high probability say, p_2 = 0.90. The decision-maker (DM) would do well to set the acceptance rates in accordance with, if not exactly equal to, this scenario. That is DM should perhaps accept 40 flights in the 1^{st} time period, 50 in the 2^{nd}, 60 in the 3^{rd}, and so on. However, as the likelihood shifts toward scenario 3, say p_1 = 0.05, p_2 =0.55, p_3 =0.40, considerable weight should be given to the optimism of s_3 and a very little to the pessimism of s_1. A GDP based on s_2 would probably prove to be overly aggressive and lead to unacceptable levels of unnecessary ground holding, thus pushing airport demand well below that of capacity. It seems that in the formulation of the GDP, the DM should hedge toward s_3 and perhaps disregard s_1 altogether.

This type of intuitive reasoning can produce close-to-optimal results for cases in which there are a small number of AAR scenarios with simple patterns. However, there could be a great number of scenarios to consider and it is not uncommon for storm intensity to peak twice or more in an affected region, thereby leading to several peaks and valleys in each AAR scenario. This greatly obscures the GHP that minimizes overall delay costs and renders almost useless any simple, intuitive approach for finding it.

4.4. A New Approach to Penalize SSGHP

One problem that airlines have with the current DGHP has been a feeling that the policy is overly conservative. That is a large number of delays are being taken on the ground, when planes could have been arriving and successfully landing. This observation sets the motivation for explicitly penalizing the waste, unused capacity at the destination airport.

4.4.1. A Model for SSGHP with Waste Penalties

Rifkin (1998) introduces new penalty variables, $P_{q,t}$, the 'waste' penalty at time t under scenario q, given c_w, the cost associated with wasting one unit of capacity. It is desired to penalize only the situations in which planes that *could be landing* are ground held. If planes are imposed ground delay, yet those planes would not have been able to land, no penalty is assessed (of course!): similarly, if there is excess capacity, but all planes have arrived, no penalty counts for. This logical reasoning grounds that $P_{q,t}$ should be the minimum of the number of planes being ground held at time t, and the excess(or slack) capacity at time t under scenario q. The following model is proposed then:

$$\min \sum_{t=1}^{T} c_g G_t + \sum_{q=1}^{Q} \sum_{t=1}^{T} p_q (c_a W_{q,t} + c_w P_{q,t}) \tag{4.7}$$

$$\sum_{t=1}^{j} A_t \le \sum_{t=1}^{j} D_t \qquad j = 1, \dots, T \tag{4.8}$$

$$\sum_{t=1}^{T+1} A_t = \sum_{t=1}^{T+1} D_t \tag{4.9}$$

$$W_{q,t} - W_{q,t-1} - A_t - S_{q,t} = -M_{q,t} \qquad t = 1,\ldots, T \quad q = 1,\ldots, Q \quad (W_{q,0} = W_{q,T} = 0) \tag{4.10}$$

$$G_t + \sum_{t=1}^{j} A_t = \sum_{t=1}^{j} D_t \qquad j = 1,\ldots,T \tag{4.11}$$

$$P_{q,t} = \min(G_t, S_{q,t}) \qquad q = 1, \ldots, Q \quad t = 1, \ldots, T \tag{4.12}$$

$$W_{q,t} = 0 \text{ OR } S_{q,t} = 0 \qquad q = 1, \ldots, Q \quad t = 1, \ldots, T \tag{4.13}$$

$$A_t \in Z_-, W_{q,t} \in Z_+, G_t \in Z_+, S_{q,t} \in Z_+ \tag{4.14}$$

It is seen that the penalty term is added to the original SSGHP model. Constraint sets (4.8), (4.9) and (4.11) remain unchanged, and the only change to constraint set (4.10) is that the slack variables ($S_{q,t}$) are explicitly represented to vanish inequalities. Constraint set (4.12) is new and represents the assignment of the penalty values. To note, (4.12) is shorthand that is minimums are not a part of the language of IP. Constraint set (4.13) is also shorthand and needs to be expanded into two disjunctive constraints, each involving a new 0-1 variable. This constraint set is necessary in order to avoid situations where, under extremely high penalty costs, the system decides to force arriving planes to take air holds under certain scenarios, even though the capacity exists to allow those planes to land immediately, in order to avoid waste penalties at later times.

Under the assumption that air holds are costlier than the ground holds, without waste penalties, the system always automatically minimizes the $S_{q,t}$, the number of air holds taken. But with waste penalties, this is not necessarily the case, as expected. Rifkin underlines that the constraint matrix of this extended model is clearly *not* TU. In fact, the LP relaxation of this model does not yield integer solutions and for even moderate sized instances, the problem is intractable under the current formulation (Rifkin, 1998).

4.4.2. A New Simplified Approach to SSGHP

The problem delved with waste penalties can require robust optimization techniques with some allowable amount of loss given a priori. Define the loss, φ, as the cost of the waste penalties. The allowable loss φ can be introduced to the original objective function of SSGHP to penalize waste capacity as follows:

$$\min \sum_{t=1}^{T} G_t + \sum_{s=1}^{S} \sum_{t=1}^{T} c_r p_s W_{s,t} + \varphi_s \tag{4.15}$$

$$\text{s.t} \qquad \varphi_s \le \alpha \tag{4.16}$$

$$\varphi_s \ge 0, \quad \alpha \ge 0 \tag{4.17}$$

Ceteris paribus for original SSGHP* constraint sets.

where α is a managerial decision parameter. This can be set to be a scalar value directly or may be an allowable percent of the expected total ground hold cost (*ETGHC*) divided by the expected total cost (*ETC*) for all periods and scenarios. More explicitly in mathematical terms,

$$\alpha = \frac{ETGHC}{ETC} = \frac{\sum_{s=1}^{S} \sum_{t=1}^{T} p_s G_t}{\sum_{t=1}^{T} G_t + \sum_{s=1}^{S} \sum_{t=1}^{T} c_r p_s W_{s,t}} \tag{4.18}$$

This is logical since the realization of a scenario with the minimal (that is allowable) discrepancy is 'forced' to derive the optimal solution. Note that the solution of the problem becomes harder in this case because the computation of α requires a backward solution, i.e. the separate cost computations of each scenario for all possible schedules. This in fact sounds intractable but may yield a new insight for further research. The motivation in this approach may be expressed as follows:

Define, for each schedule i, $EGHC_i$ as the expected ground hold cost if that schedule is realized and d_i be the absolute value of the (expected) discrepancy between these cost figures. A rough algorithmic approach to find the optimal schedule is to choose i^* such that

$i^* = \min\{d_i : |EGHC_i - ETGHC|, \forall i \in (1..s)\}$ for all possible values of i.

EXAMPLE 2: To clarify, consider an example consisting of two scenarios and two time periods (S=2, T=2). Assume that the demand is 1 during both periods ($D_t = 1$, $t = 1, 2$). Under the first scenario, with probability 0.6, the capacity is 0 at times 1 and 2 ($p_1 = 0.6$, $M_{1,1} = 0$, $M_{1,2} = 0$). Under the second scenario, that occurs with probability 0.4, the capacity is 1 for both times ($p_2 = 0.4$, $M_{2,1} = 1$, $M_{2,2} = 1$). Also assume that $c_g = 1$ and $c_a = 2$ unit cost (i.e. $c_r = 2$). Table 4.2. lists the possible schedules, as well as the number of the ground holds, the expected number of air holds and the relevant costs.

Regarding no waste penalty, schedule 1 with the least ETC is optimal. To involve waste penalties, one has to find ETGHC, simply computing the weighted average of the ground holds (GHC, note that $c_g = 1$) which is (3+2+1+0)/4 = 2. With respect to the algorithm above, it is trivial to see that $i^* = 2$, that is optimal schedule is $A_1 = 0$ and $A_2 = 1$ regarding the waste penalties. Is it unusual that the optimal solution found without waste penalties (Schedule 1) does not match the one found with waste penalties (Schedule 2)? The answer is NO! Actually this is an open question and needs further investigation. To compare, the results for Rifkin's model with waste penalties, for the same problem instance with one more additional given, $c_w = 0.75$ unit cost, are exhibited in Table 4.3. It should be noted that the new robust approach and the Rifkin's approach yield the same result.

Table 4.2. Possible Schedules for Example 2

Schedule Number	Arrivals		Ground Holds	Expected Air Holds	Expected Total Costs	Expected d_i
	t=1	t=2				
1	0	0	3	0	3	1
2	0	1	2	0.6	3.2	0
3	1	0	1	1.2	3.4	1
4	1	1	0	1.8	3.6	2

Table 4.3. Comparison with Rifkin's Model for Example 2

Schedule Number	Arrivals		Ground Holds	Expected Air Holds	Expected Waste Penalties	Expected Total Cost
	t=1	t=2				
1	0	0	3	0	0.8	3.6
2	0	1	2	0.6	0.4	3.5
3	1	0	1	1.2	0.4	3.7
4	1	1	0	1.8	0.0	3.6

Table 4.4. Cost Sensitivity Analysis for Example 2

Schedule Number	Arrivals		Ground Holds	Expected Air Holds	Expected Waste Penalties	Expected Total Cost
	t=1	t=2				
1	0	0	3	0	0.8	3.6
2	0	1	2	0.6	0.4	4.1
3	1	0	1	1.2	0.4	4.9
4	1	1	0	1.8	0.0	5.4

Again, Schedule 2 is optimal with the minimal expected total cost. To note, Rifkin's model is more sensitive to the cost structure, thereby for varying cost structure it is expected to have varying optimal schedules! To verify, suppose that c_a is changed from 2 to 3 unit costs, letting the other parameters be unchanged. Referring to Table 4.4. the optimal solution is Schedule 1 now! That is, intuitively, depending on the cost ratio ($c_r = c_a/c_g$) effectiveness domain, i.e. depending on the cost structure, the optimal solutions are subject to change. This issue will be explored in Chapter 5 in detail.

4.4.3. A Few Closing Notes on Simplified Approach

The example above clearly shows that a more simplified (and robust in a sense) approach is better since it does not involve with waste penalties and related waste penalty cost, yielding a simple form. Yet the backward solution and the loss allowance equation complicate the structure of the model. However this is an open area to delve.

The Rifkin's model is verified for its sensitivity. Yet the Robust Approach is not affected with variations in c_g and c_a because it already deals with the discrepancy between the averaged ground holding cost and presuming an amount of loss. To clarify, d_{i^*} was used to find the optimal, not even tackling with α. Both, using α or d_i, should give the same results since α is already linear with d_i assuming that the costs are invariant of time. α is just a convenience for mathematical notation.

4.5. Some Theoretical Basis for SSGHP

The following results are basically obtained from (Nemhauser and Wolsey, 1998) and (Ball et al., 1999)

Definition 1: Given an instance I of SSGHP, the upper envelope of I is the sequence $U \in Z^T_+$ satisfying $U_t = \max_{s\in S} M_{s,t}$, $1 \le t \le T$, and the lower envelope is the sequence $L \in Z^T_+$ satisfying $L_j = \max_{s\in S} M_{s,t}$, $1 \le j \le T$.

Lemma 1: Given an instance of SSGHP, and an optimal schedule A^*, $A^*_t \le U_t$ for $1 \le t \le T$.

Proof Sketch: Assume that A_t^* is strictly greater than U_t for some t. Defining $d = A_t^* - U_t$, consider the new schedule $A^{\sim}$ constructed by setting $A^{\sim}_t = U_t$, $A^{\sim}_{t+1} = A^*_{t+1}+d$, and $A^{\sim}_{t^\wedge} = A^*_{t^\wedge}I$ for all other $t^\wedge$. Under all possible scenarios, this schedule will have d

additional ground holds and d less air holds at period t, and the same number of air and ground holds for all $t^\wedge \neq t$. Since $c_a > c_g$, Cost($A^\frown$) = Cost(A^*) – d*(c_a- c_g) < Cost (A^*), contradicting the assumption that A^* was optimal. □

Definition 2: An m row, n column (0, 1, -1) matrix M is a *network matrix* if there exists a directed tree R on m+1 nodes and a one-to-one mapping of the rows of the matrix onto the edges of R with the property that each column of the matrix corresponds to the characteristic vector of a path in R. The more familiar *node-arc incidence matrices*, with a single 1 and –1 in each column form a special case of this construction.

Definition 3: An m row, n column matrix A is totally unimodular (TU) if the determinant of each square submatrix of A is equal to 0, 1 or –1.

Network matrices are desirable because they are totally unimodular and give rise to integral polyhedra.

Theorem 1: Let A^T be the transpose of the constraint matrix associated with SSGHP. Then A^T is a network matrix.

The proof sketch can be found in (Ball et al., 1999).

Theorem 2: An m-row n-column matrix is TU if and only if for every subset of the rows $I \subseteq N$ there exists a partition of I into two subsets I_1 and I_2 such that

$$\left|\sum_{i \in I_1} a_{ij} - \sum_{i \in I_2} a_{ij}\right| \leq 1, \quad j = 1,2,\ldots N$$

In (Rifkin, 1998), it is shown that such a partition exists for the constraint matrix of SSGHP.

Corollary 1: The constraint matrix associated with SSGHP is totally unimodular. Its LP relaxation yields an integral solution, and the SSGHP can be solved in polynomial time.

The results obtained above, in sum, reveal that the dual tree of the constraint matrix of SSGHP is a network matrix and is TU, meaning that it can be solved in polynomial time (tractable). A proof of the above corollary is given in Appendix A. To note, the primal problem of SSGHP can not be recast a s network flow problem, which can be verified by applying standard network recognition algorithms. (see Nemhauser and Wolsey (1988) and Bixby and Cunningham (1980) for futher details.)

5. RESEARCH FINDINGS AND DISCUSSION

5.1. Implementation of the SSGHP Model

The SSGHP model, in its latest form, is in fact simple to implement. The model is coded in ILOG OPL Studio, a special optimization language of which solver is CPLEX. In this coding script, the model is divided into two parts: the model file (with *.mod extension) and the data file (with *.dat extension). These two parts are then linked together and form a whole modeling code (with *.prj extension). ILOG OPL Studio stores OPL statements in model files. A model file can be executed in the environment without any additional requirements. However, large problems are better organized by separating the model of the problem from the instance data. ILOG OPL Studio uses the concept of a project to associate a model file with a number of data files. The model file declares the data but does not initialize it. The data files contain the initialization of each data item declared in the model. The project file then organizes all the related model and data files. A project provides a convenient way to maintain the relationship between related files and runtime options for the environment (Hentenryck, 1999).

The OPL code for SSGHP is listed in Appendix B.1. As it can be observed thereof, the decision variables are A_t , G_t and W_st, directly standing for A_t, G_t and $W_{s,t}$. The parameters in the code are nbS (number of scenarios), nbT (number of periods), c_r (cost ratio), D_t (PAR, the period based demand), p_s (probability of scenario *s*), and finally M_st (AAR with respect to scenarios *s* in period *t*). The model user should enter / change these parameters in the data file and simply run. The results can be visually observed for each data component explicitly. This is an advantage of OPL Studio. In coding, the definitions of variables are enforced to resemble to the original mathematical formulation for a better understanding. Onwards, an experimental design to test the speed of the solver for varying sizes of scenarios and periods are analyzed.

5.2. Experimental Design on Computation Time Sensitivity

As of major parameters in the model, experimentation on the size of the model is to be included for a thorough study. Letting the number of scenarios fixed, the number of periods will be increased from 2 to 24 with an increment of 2 time slots. As to sensitize the model's computational efficiency on the number of scenarios, for a fixed number of time periods, the model's run time is analyzed for various number of scenarios, reasonably ranging from 2 to 10, with a marginal increase. The number of constraints and the number of variables will also be considered. The density of the model for each run should be as equal as possible to dispose the effect of density on solution time. For this reason, as a logical approach, for every period, PAR will be taken 1 and the scenarios will be formed with equal probabilities and with a special composition of 0's and 1's. The cost ratio is fixed to 2.25 depending on the estimation of (ATAA, 1995). To clarify, demand for each period is 1 and the scenario generation for 4 scenarios and 7 time periods is as follows:

Table 5.1. Exemplary Scenario Generation for Experimentation with S=4 and T=7

S	p_s	**Capacity**						
		$t=1$	$t=2$	$t=3$	$t=4$	$t=5$	$t=6$	$t=7$
1	0.25	0	0	0	0	1	1	1
2	0.25	1	0	0	0	0	1	1
3	0.25	1	1	0	0	0	0	1
4	0.25	1	1	1	0	0	0	0

Actually, in this fashion, if it is observed with a little care, it is tried to construct upper and lower triangular shapes in the data structure. In such a way, the experimenter suffices to block the bias of model density for each trial. To generalize, as a guiding rule in this experimental design, for this special problem instance, every scenario is given an equal weight of probability with $1/S$, where S is the number of scenarios. And for each scenario, there are T-S 1's and S-1 0's, in order to ornate upper and lower triangular forms, where *T* stands for the number of period, so that the number of non-zero elements in the model are enforced to be equal for an objective comparison. With this in mind, the experiments are designed as in Table 5.2 and Table 5.3.

Table 5.2. Computational Consideration on Number of Periods (S =2)

S = 2	T	2	4	8	12	16	20	24	28	32	36	40	50	60	70
	soln time	0,05	0,00	0,28	0,05	0,00	0,06	0,05	0,08	0,00	0,11	0,22	0,11	0,33	0,17
	rows	15	21	33	45	57	69	81	93	105	117	129	159	189	219
	columns	16	24	40	56	72	88	104	120	136	152	168	208	248	288

Table 5.2. shows the solution times (CPU solution time including the sparsing time), the number of rows in the extracted model (coincides with the number of constraints) and the number of columns (referring to the number of variables) for a fixed number of scenarios ($S = 2$) and for varying number of periods ($T = 2 \ldots 70$). This experiment is conducted to analyze the effect of increase in T to the complexity of the model. For sure, both for *S* and *T*, any increase in these parameters is to increase the solution time but the quantification is the concern here. The data sets for these experiments are established as aforementioned. To exemplify, for 2 scenarios and for 40 periods, the model is solved in 0,22 seconds, with 129 rows and 168 columns. Since *T* is sensitized in this experiment, S is taken to be 2 for simplicity and already to necessitate and suffice two kinds of scenarios: the pessimist and the optimist ones. To examine the effect of increase in *T* visually, Figure 5.1 and 5.2 are devised.

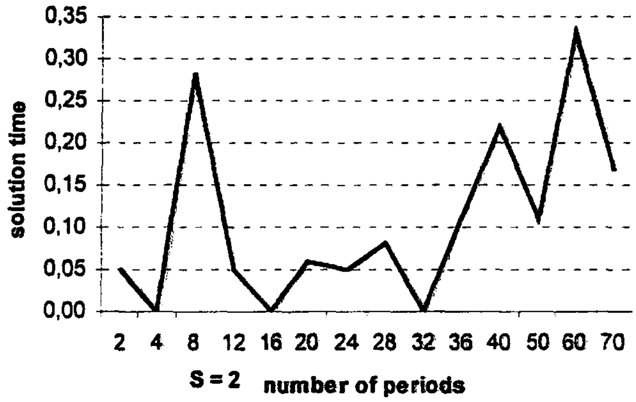

Figure 5.1. CPU Solution Time vs. T

For S=2 and for T in the range 2...70, the CPU solution times are shown above. The maximum solution time is 0.33, which is very satisfying. But it should be noted that OPL Studio is configured with CPLEX solver engine and CPLEX stores a sparse representation of the model matrices and unless the problem is large, CPLEX sparse times dominate the running time. Thus, conclusively, it can be claimed that for the experimental domain, the solution times are smaller than even 1 second and that is very satisfying.

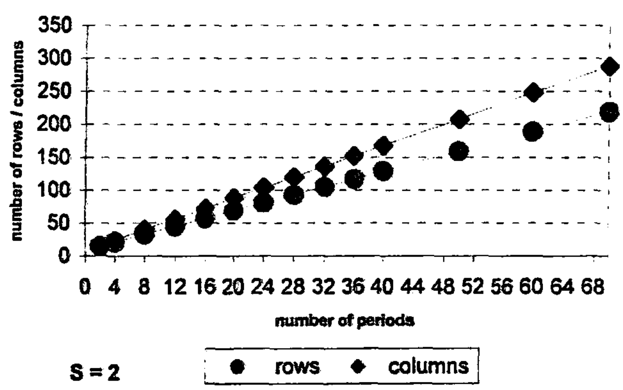

Figure 5.2. Number of Rows and Columns vs. T

Table 5.3. Computational Considerations on Number of Scenarios (T =16)

	S	1	2	3	4	5	6	7	8	9	10	12	14
T = 16	soln time	0,11	0,11	0,06	0,06	0,11	0,17	0,22	0,22	0,27	0,22	0,22	0,33
	rows	38	57	76	95	114	133	152	171	190	209	247	285
	columns	54	72	90	108	126	144	162	180	198	216	247	285

Figure 5.3. basically exhibits that as the number of periods are increased for 2 kinds of scenarios, both the number of rows and the number of columns increase linearly yet the latter dominates the former.

Table 5.3. is designed to analyze the effect of the increase in the number of scenarios for a fixed number of time periods. In this experiment, T is fixed to 16 (this is logical because, usually, the GDPs are re-executed for a planning horizon of 4 hours, and getting time slots as 15 minutes, 16 periods is sufficient), and S is sensitized over the range 1...14. Actually, as mentioned in the intuitive approach to SSGHP, the success in a GDP partly lies in accurate formation of possible scenarios. So, 14 scenarios for a GDP planning horizon shows that the forecast done is not good (not reasonable in real life situations!), but since the concern in this experiment is to analyze the effect, there seems no problem for now. The table is designed in similar fashion with Table 5.2. and shows the effect on solution times, the number of rows and the number of columns with respect to the increase in the number of scenarios. The results in this table are figured out in Figure 5.3. and Figure 5.4.

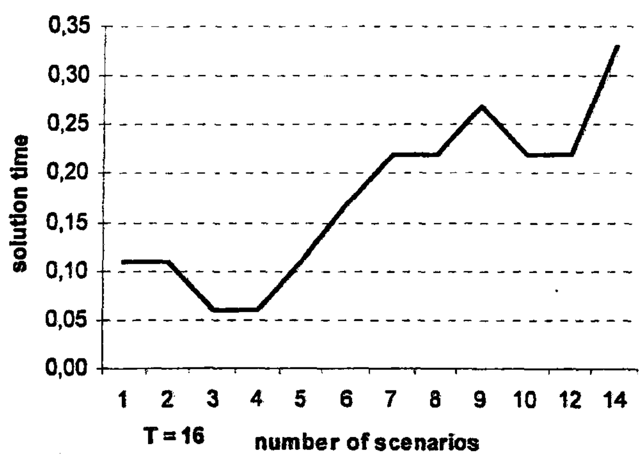

Figure 5.3. CPU Solution Time vs. *S*

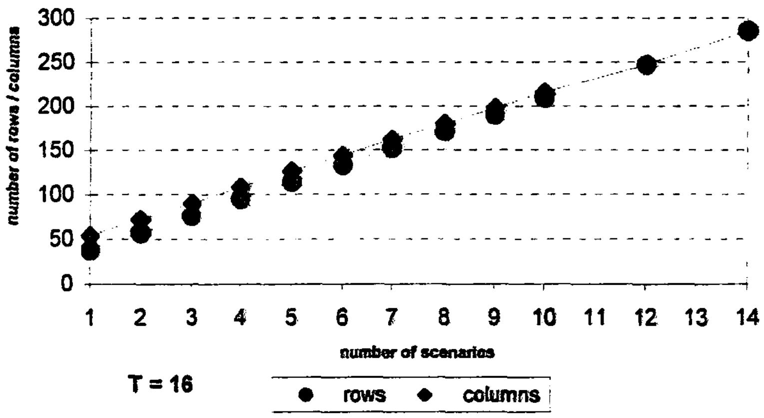

Figure 5.4. Number of Rows and Columns vs. *S*

Figure 5.3. extracts similar results to those of Figure 5.1. Again, for fixed T=16, and for varying S between 1 and 14, the solution times do not exceed 0.35 seconds. Figure 5.4. says that the number of rows and the number of columns are nearly in the similar incremental trend for increasing number of scenarios. It is interesting that the number of the rows and columns coincide for S=12 and S=14.

To sum up for this sub section, it is observed that the solution times for varying T and S values are very satisfying for the model implemented in OPL Studio. A sufficiently large number of scenarios and periods are designed and found to be at most for S=10 and T=23, with 286 rows and 300 columns, for at most 0.38 CPU time, as Table 5.4. exhibits.

Table 5.4. Computational Experiment for Extreme (S, T)

S = 10	T	20	21	22	23
	soln time	0,11	0,11	0,38	0,17
	rows	253	264	275	286
	columns	264	276	288	300

5.3. Benchmark with Recent Models via Computational Efficiency

As for large-scale optimization problems, computational efficiency, generally, becomes the first criteria. In this section, the new proposed SSGHP model in this thesis is benchmarked with those of Rifkin's and Richetta's, on the basis of the number of the columns and the number of the rows. The number of the non-zero elements in the models can be another criteria since all these models are solved with LP relaxations but it suffices to do so as here.

Table 5.5. Comparison via Fixed S=2

(S,T) = (2, 2...70)		*T*	2	4	8	12	16	20	24	28	32	36	40	50	60	70
	Rows	M.Ali	15	21	33	45	57	69	81	93	105	117	129	159	189	219
		Rifkin	9	17	33	49	65	81	97	113	129	145	161	201	241	281
		Richetta	10	16	28	40	52	64	76	88	100	112	124	154	184	214
	Columns	M.Ali	16	24	40	56	72	88	104	120	136	152	168	208	248	288
		Rifkin	13	27	55	83	111	139	167	195	223	251	279	349	419	489
		Richetta	16	33	79	141	219	313	423	549	691	849	1023	1528	2133	2838

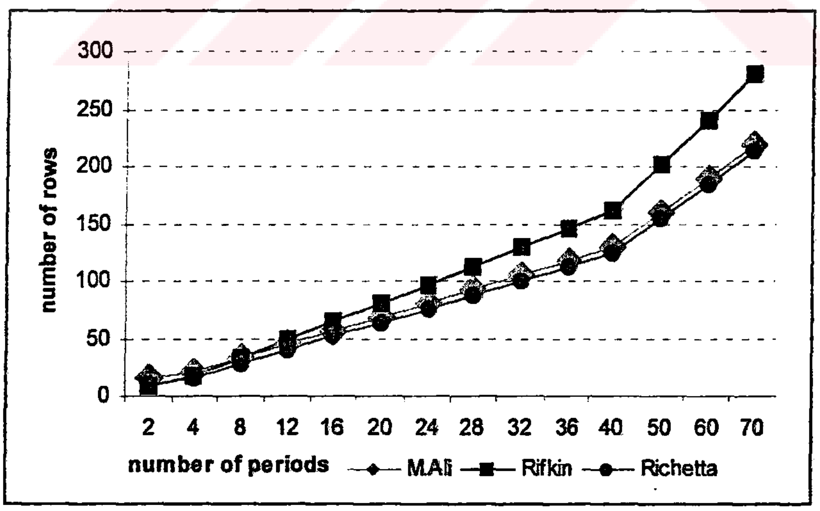

Figure 5.5. Benchmark on Number of Rows for Fixed S=2

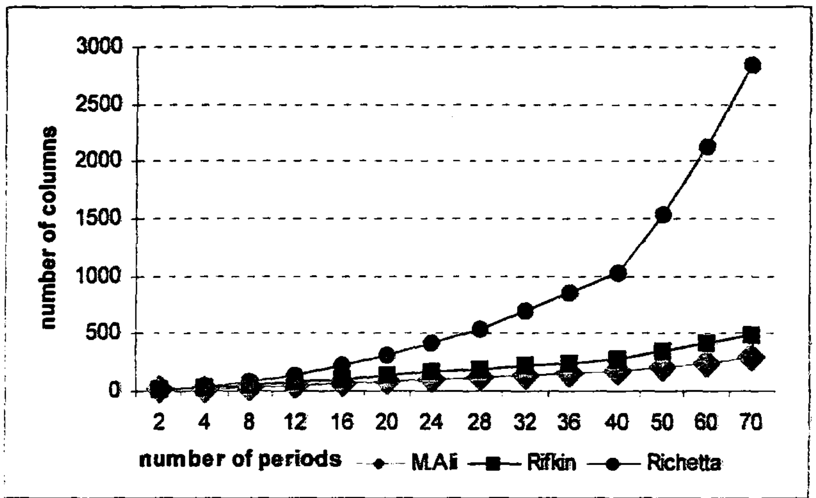

Figure 5.6. Benchmark on Number of Columns for Fixed S=2

Table 5.6. Comparison via Fixed T=16

(S,T) = (1...14, 16)		S	1	2	3	4	5	6	7	8	9	10	12	14
	rows	M.Ali	38	57	76	95	114	133	152	171	190	209	247	285
		Rifkin	19	37	55	73	91	109	127	145	163	181	217	253
		Richetta	34	52	70	88	106	124	142	160	178	196	232	268
	columns	M.Ali	54	72	90	108	126	144	162	180	198	216	247	285
		Rifkin	80	111	142	173	204	235	266	297	328	359	421	483
		Richetta	169	202	235	268	301	334	367	400	433	466	532	598

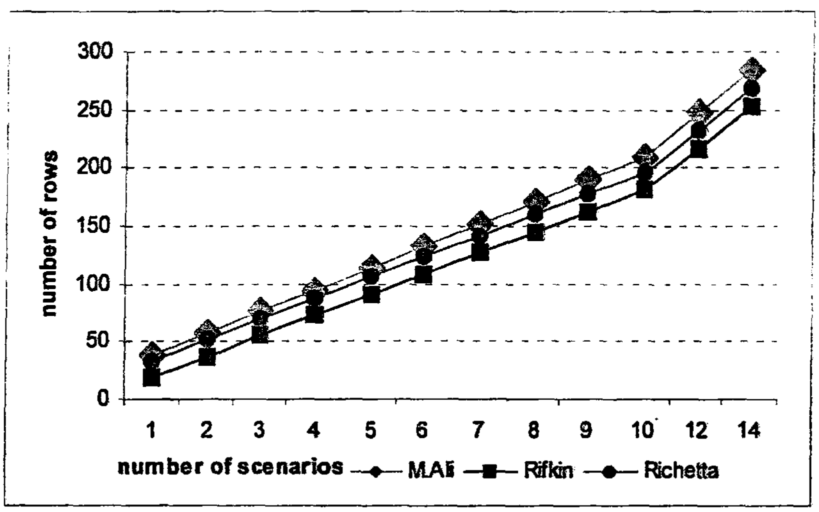

Figure 5.7. Benchmark on Number of Rows for Fixed T=16

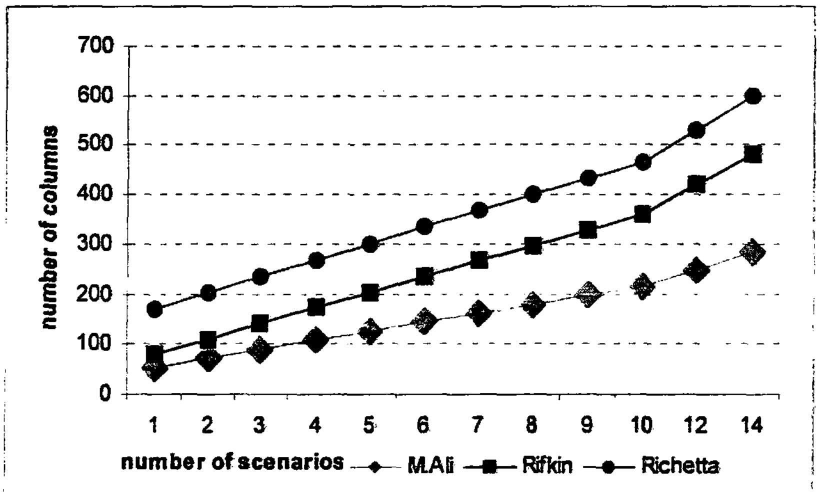

Figure 5.8. Benchmark on Number of Columns for Fixed T=16

Table 5.5. is devised to numerically analyze the complexity of the proposed M.Ali's model (SSGHP*) with respect to those of Rifkin's (Rifkin, 1998) and Richetta's (Richetta and Odoni, 1993). As seen from this table, S is fixed to 2 and for varying T values, the number of columns and the number of rows are computed for the models of M.Ali, Rifkin and Richetta. To better analyze the results, Figure 5.5. and Figure 5.6. are used. Figure 5.5. explores that for this specific results, the number of rows for all models show a piecewise linear trend. As thenumber of periods is increased for more than 40, the slope is plummeted. The data for M.Ali are specially bolded. The figure says that M.Ali's and Richetta's models are superior to that of Rifkin's by means of the number of rows for fixed number of scenarios. Figure 5.6. shows the number of columns for each model for fixed S=2. It is observed that Richetta's trend is curvilinear whereas the others are linear. For T=70, Richetta's model generates nearly 3000 columns whereas M.Ali's and Rifkin's models generate at most 500 hundred columns. Moreover, M.Ali's models are observed to be superior to Richetta's (of course!) and Rifkin's model with lesser columns, which proves faster computation.

Table 5.6., Figure 5.7. and Figure 5.8. are tooled for the same aim: to benchmark the models of M.Ali, Rifkin and Richetta on the basis of fixed number of periods. To exemplify, Table 5.6. says that, for fixed number of periods (T=16) and 10 number of different scenarios, M.Ali's model generates 209 rows and 216

columns whereas Rifkin's models does 181 rows and 359 columns and Richetta\s model does 196 rows and 466 columns. Figure 5.7. reveals out that all the models show piecewise linear trends on the basis of the number of rows generated with respect to the increase in the number of scenarios. It is observed that more less, all the models are competent in this category, yet Rifkin's model is superior to both of the others, with minor slack on the number of rows. Figure 5.8. now compares these models on the basis of the number of columns. Again, all the models show a piecewise linear trend. Yet M.Ali's model is superior to both of the models and Rifkin's model is superior to Richetta's one.

Of all the comments above, to sum up, it can be concluded that M.Ali' model makes substantial improvements on the recently developed ones and for all models, the complexity increases far more for the number of scenarios bigger than or equal to 10 and for the number of periods bigger than or equal to 40. This may enlighten the reason why S is chosen to be 10 in Table 5.4. Also it shows that the scheduler is more confident for $S<10$ and $T<40$, by computational means. Yet it should be noted that the data formed for these experiments are not denser than the actual data, i.e. every demand in each period will not be 1, as explained in section 5.2. In summary, the comparison on the same basis illustrates possible improvements in M.Ali's model.

5.4. A Numerical Example

Here a numerical example is explored. In this fashion, the model implementation and the results will be divulged. The data set is constructed for 16 time periods and 5 scenarios. If one gets the time slot to be 15 minutes, then the model runs for 4 hrs time horizons, which is sufficient to be practical in real life applications. Note that the period length should be at least 15 minutes because of the dependability statistics. The demand and the scenarios are depicted as follows:

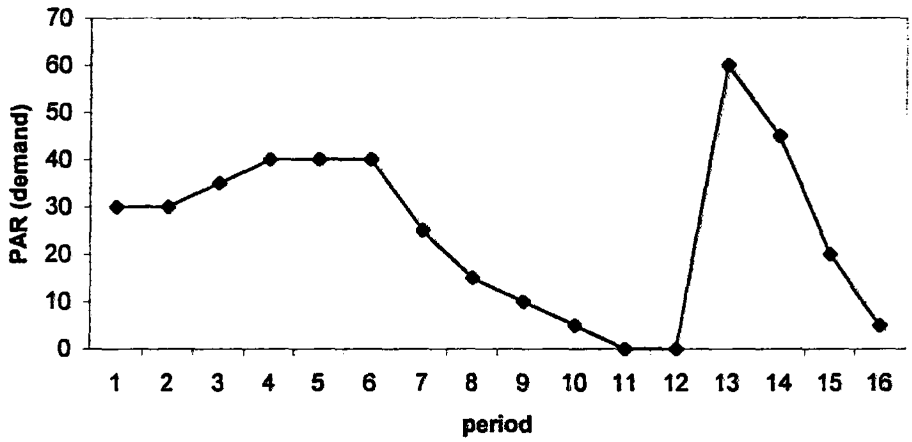

Figure 5.9. Demand Profile for Numerical Example

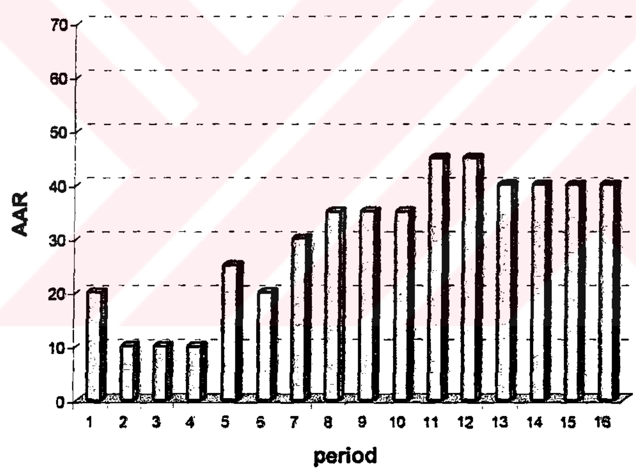

Figure 5.10. AAR Scenario 1 ($p_s = 0.1$)

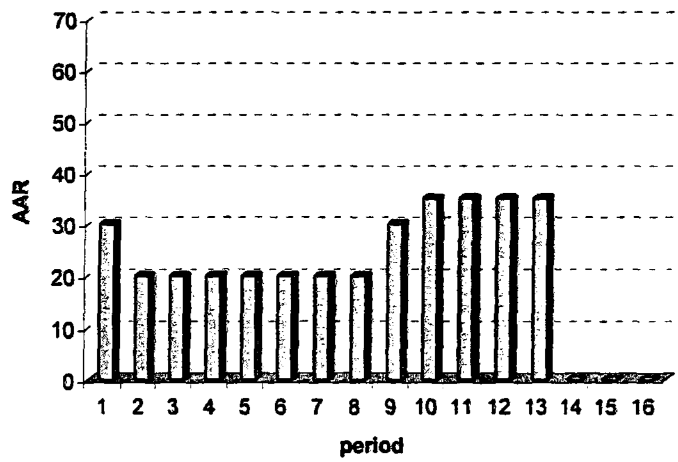

Figure 5.11. AAR Scenario 2 ($p_s = 0.1$)

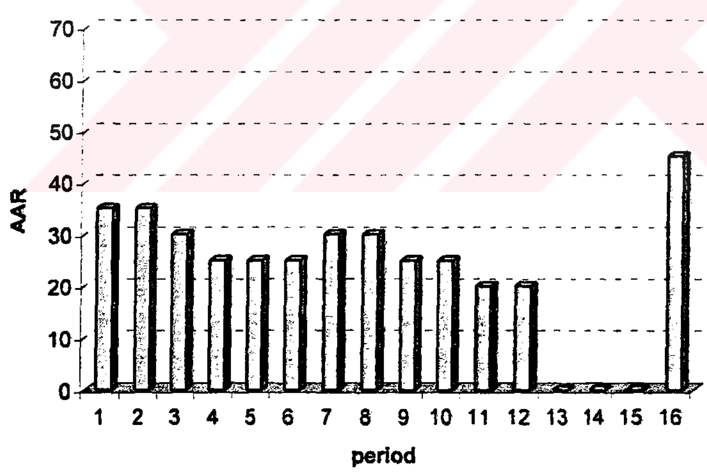

Figure 5.12. AAR Scenario 3 ($p_s = 0.2$)

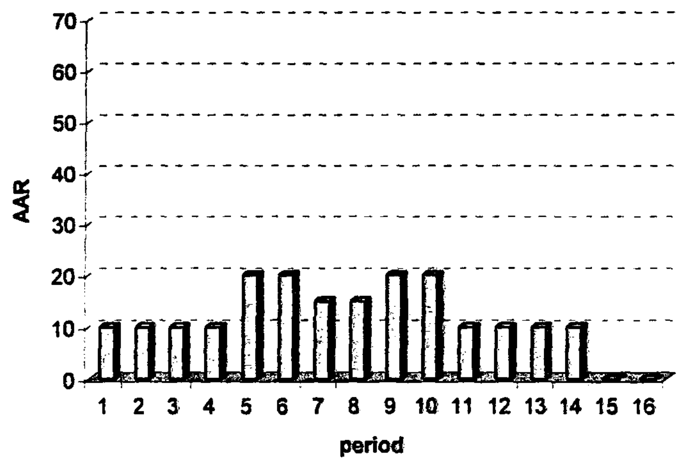

Figure 5.13. AAR Scenario 4 ($p_s = 0.3$)

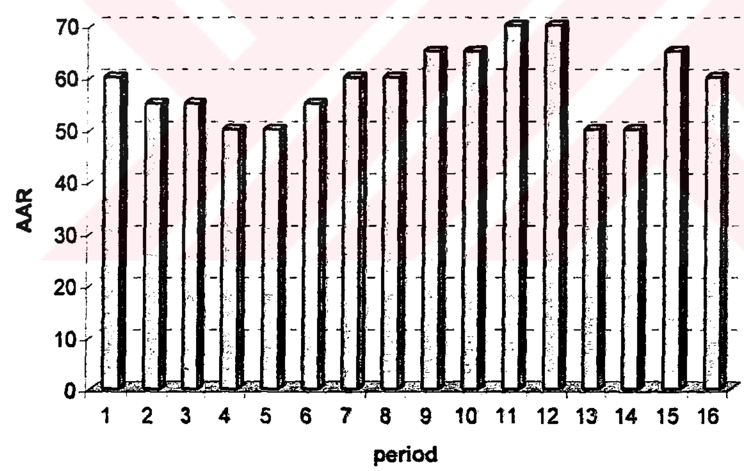

Figure 5.14. AAR Scenario 5 ($p_s = 0.3$)

Note that the 4th scenario is a pessimist one whereas the 5th one is optimist. The rest are deviating between these limits. The data suffice to represent a real case compared to the example in (Ball et al., 1999). The model (ssghp.mod and ssghp.dat) is coded in OPL Studio 3.0.2 and is listed in Appendix B.1. This specific example consists of 126 variables and 233 constraints. The solution time is 1.21 seconds, which is very appealing via CPU time considerations. An exemplary visualization of

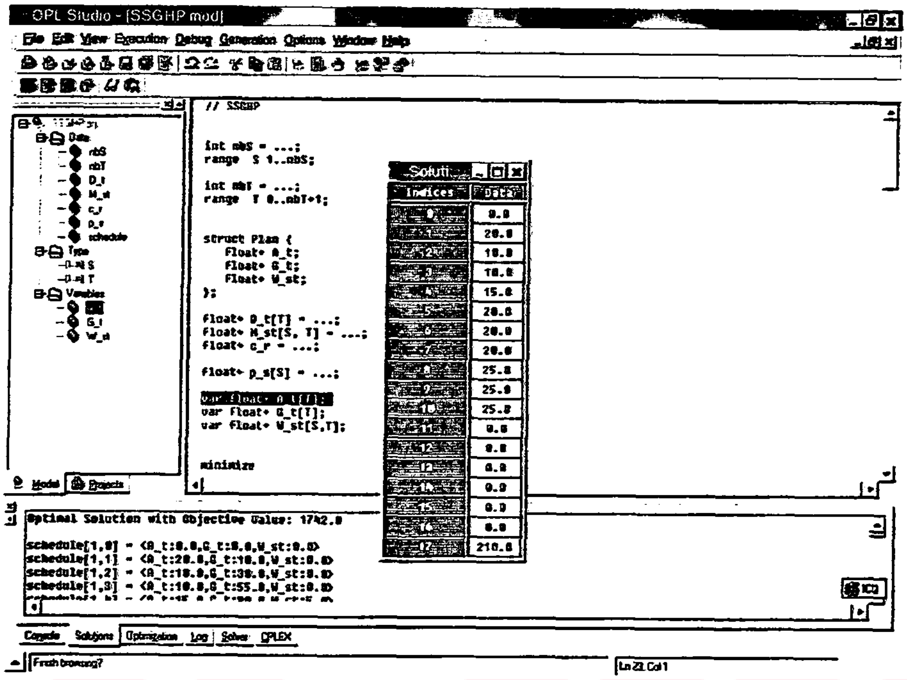

Figure 5.15. Visual Exhibit of the Model Output (I)

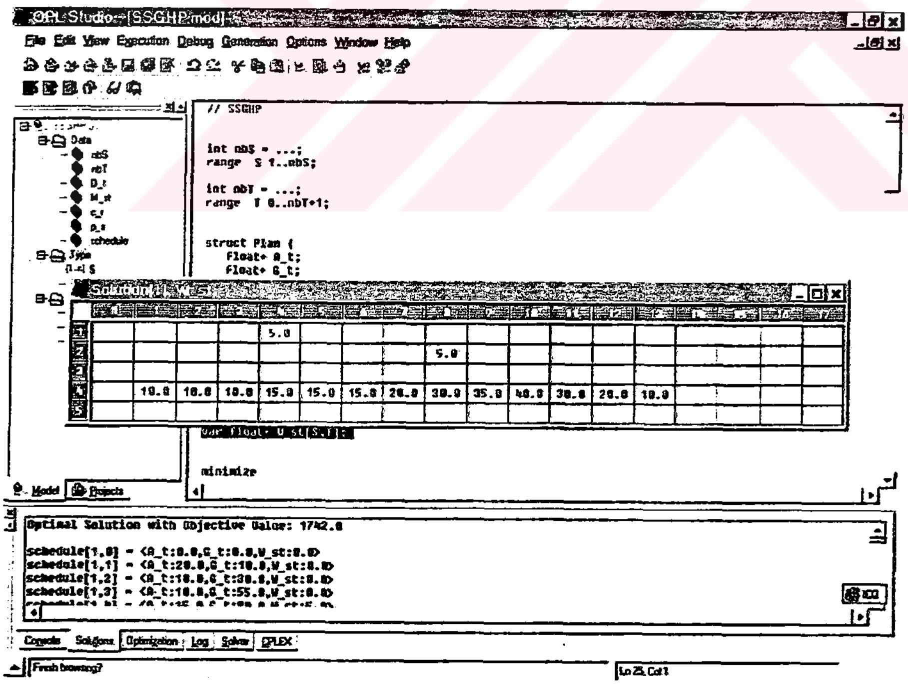

Figure 5.16. Visual Exhibit of the Model Output (II)

the model output is shown in Figure 5.15. In this figure, it is shown that for this specific example, the A_t decision variables can explicitly be seen whereas the $W_{s,t}$ variables are shown in Figure 5.16. As to exemplify, the full solution tableau for this specific example with cost ratio 3.0 is given in Appendix B.2.

It should be noted that the decision variables are naturally found to be integral which coincides with the theoretical result of the network structure of SSGHP. It should also be observed that the A_t and G_t variables are fixed for all scenarios (since they are static!) and that $W_{s,t}$ may differ for each scenario(since it is stochastic!).

5.5. Some Experimental Highlights on Cost Analysis of SSGHP

As one of the main assumptions in modeling SSGHP, it is noted that it is highly difficult to assess the actual values of ground holding and airborne holding costs. Yet as a logical evaluation, the ratio of airborne cost and ground holding cost is super linear[1]. There is well done study on cost frontier analysis of holding costs in GHPP in (Hoffman, 1997). For varying values of cost ratio (c_r), the following results are obtained for the numerical example. Figure 5.17. depicts that as the cost ratio increases, the objective value increases curvilinear and stabilizes at some point. As seen, for c_r greater than or equal to 15, the objective value is found to be constant. This means that the worst case works for the model so as to impose as much ground holds as much possible rather than air holds. A second way to investigate the effect of cost ratio, Figure 5.18. is devised. Here *d* (norm) represents the normalized increase in objective value as the cost ratio increases. It is observed that there is a sharp decrease between the cost ratio domain of 1 and 4, this means that the analysis should be done with more care in this region because the model is most sensitive to cost in this region.

[1] It is estimated that for a typical flight, every minute of ground delay costs $20.35 while every minute of airborne delay costs $45.85. (ATAA, 1995)

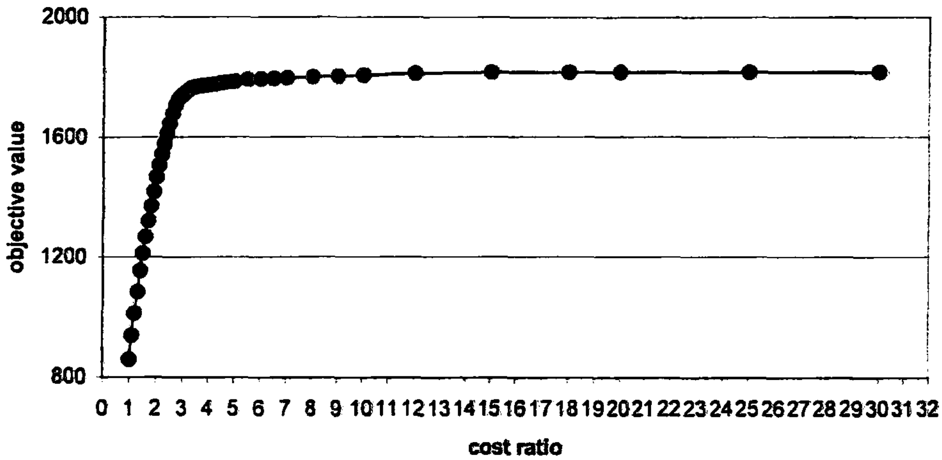

Figure 5.17. Sensitivity on Cost ratio vs. Objective Value

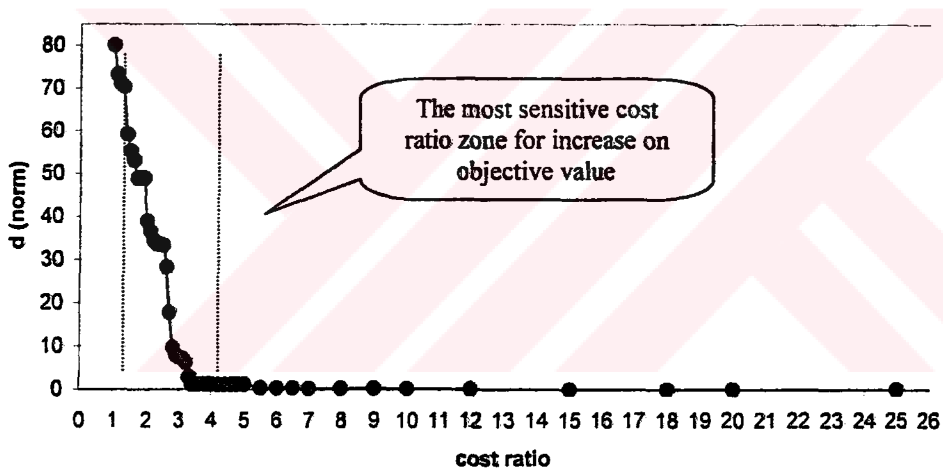

Figure 5.18. Sensitivity on Cost ratio vs. Normalized Increase in Objective Value

Both figures on cost sensitivity suggest that the cost ratio in SSGHP (actually in most of the stochastic models) should be examined with care since it directly affects the execution of the model.

6. CONCLUSIONS AND FUTURE RESEARCH

6.1. Research Summary

Research in this thesis concentrates on the introduction of ATFMP in broad sense and specifically GHPP thereafter. The GHPP simply tries to balance the air traffic flow system, centralizing on imposing ground holds for the planes rather than letting them enforce airborne delays, which are more costly and worse in safety considerations. The problem is introduced with a general scope and then narrowed to SSGHP gradually. A new model is developed taking the Rifkin's model as a seminal work. This model is coded in OPL 3.0.2 optimization language for varying cost structures and for a specific real-like data. As of modeling neatness and of computational time better results are attained. Also our research focuses in developing a new robust model to penalize unused capacities in destination airports. The research findings yield that the proposed model is better than the recent models with respect to computational efficiency on the basis of the number of rows and columns generated in the model matrices for specific problem instances.

6.2. Concluding Remarks

Here are some concluding remarks on this thesis and are mattered as follows:

- Models like GHPP are characteristically treated as side-constrained network flow problems in which nodes represent locations in space and time, leading to extremely large problem instances. Because such a problem requires integer solutions and is not a true network flow problem, specialized heuristic algorithms have to be developed to take advantage of the underlying network structure.
- Such large-scale problems like SSGHP require specific algorithmic approaches for its sub problems and the current and seemingly future research will focus on the computational efficiencies of these algorithms.
- SSGHP needs to be delved for precise cost structures.

- It is shown in this thesis that models (especially for integer programs) have to be modeled strong as well as artsy foreseeing the computational problems that may be faced.
- Network propagation effects in SSGHP, especially for the multiple airports case are open areas to investigate.
- CDM is truly an important concept in ATFM and is to be given a weight of research it deserves.
- The SSGHP is cost sensitive and this may intervene the solution procedure of the model.
- The SSGHP is proven to be totally unimodular, thus it yields integral solutions when linear programming relaxation techniques are used.
- Pre and post processing can reduce most of the large-scale problems to smaller (actually tractable) sizes.
- OPL Studio 3.0.2. attains very satisfying computational results for this special kind of problem. Its highly enhanced optimization language enables to easily solve the constraint programming as well as integer programming with various solution options.
- The research on air traffic elimination seems to get more attraction in OR and is very challenging, not by only bringing out specialized solution techniques but also by being typical prototypes for large scale optimization problems.
- The new proposed SSGHP model brings out substantial improvement on the recent models by means of robust approach on waste penalties and computational efficiencies on the number of rows and columns generated.
- It is the system that is crowded, but not the sky!

6.3. For Future Research

SSGHP is an example of large-scale combinatorial optimization problems and such problems require special solution techniques. Regarding the integer property of the problem, things get more complicated. Also the model can be transformed to assignment problem, multi-period production planning and scheduling, just to enumerate few. The combinatorial aspect of SSGHP will be more challenging regarding the multi-airport case. Also incorporating CDM and the 'hub and spokes' operations, there seems to be a lot to be researched in this field, which may be an interesting doctoral level research. Also, the execution of the previously developed heuristics is an open area to concentrate on. There also stands a work to be investigated on the robust approach implementation of the newly proposed model.

REFERENCES :

ATAA, Air Transport Association of America,1992. Anatomy of a Delay, NY, 63 p.

ATAA, Air Transport Association of America,1995. Aircraft Operating Costs by Stage of Flight and Associated Air Traffic Control Delay Costs Based on a Typical Flight, Washington, USA, pp 5-11.

ANDREATTA, G., ROMANIN-JACUR, G., 1987. Aircraft Flow Management Under Congestion Transportation Science, 21, 249-253

BALL, M., HOFFMAN, R., ODONI, A., and RIFKIN, R., 1999. The Static Stochastic Ground Holding Problem with Aggregate Demands. NEXTOR T.R. 99-1: 4-11.

BERTSIMAS, D., STOCK, S., 1998. The Air Traffic Flow Management Problem with En Route Capacities. Operations Research, 46(3): 406-412

BIXBY, R. E., CUNNINGHAM, W.H., 1980. Converting Linear Programs to Network Problems. Mathematics of Operations Research, 5(3): 321-357.

BURLINGAME, R., BOYD, E. A., and LINDSAY, K. S., 1995. Solving Large Integer Programs Arising from Air Traffic Flow Problems, in Intelligent Scheduling System, (D.E. BROWN and W.T. SCHERER eds.), Kluwer Academic Publishers, Boston,USA.

EPSTEIN, L., FUTER, A., MEDVEDOVSKI, L., 1992. Multiple airport Scheduling, ATMS Interim Report, Volpe National Transportation System Center, Cambridge, MA, USA.

GLOVER, F., KLINGMAN, D., and PHILIPS, N., 1992. Network Models in Optimization and Their Applications in Practice. . John Wiley & Sons, Inc., New York, 284 p.

GRANDEAU, C. G., CLARKE, M. D., and MATHAISEL, D. F.X., 1998. The Process of Airline System Operations Control. : Operations Research in the Airline Industry, (G. YU ed):313-369

HALL, S., 1985. Air Traffic Simulation: One and Two Runway Airports, Transportation Engineering, 106 : 756-784

HELME, M. R., LINDSAY, K. S., MASSIMINI, S. V., 1992. Optimization Models for Air Traffic Flow Control Management : New Formulations. MTR 92W0000152, The Mitre Corporation, McLean, VA, USA

HENTENRYCK, P. V., 1999.The OPL Optimization Programming Language. The MIT Press, MA, USA, 257 p.

HOFFMAN, R. L., 1997. Integer Programming Models for Ground-Holding in Air Traffic Flow Management. Dissertation, UMD-College Park, USA, 153 p.

LAWLER, E. L., 1976. Combinatorial Optimization: Networks and Matroids. Holt, Rinehart and Winston., New York, 374 p.

LINDSAY, K. S., BOYD, E.A., BOOTH, G., HARVEY, C., 1998. The Use of Optimization to Perform Air Traffic Flow Management, in Operations Research in Airline Industry (G. YU ed.), Kluwer Academic Publishers, Boston, USA, p 287-311

LINDSAY, K.S., 1993. Traffic Flow Management Modeling with the Time Assignment Model, MTR 93W0000119, the MITRE Corporation, McLean, VA, USA.

LUO, S., YU, G., 1998. Airline Schedule Perturbation Problem: Landing and Takeoff with Nonsplittable Resource for the Ground Delay Program, in Operations Research in Airline Industry (G. YU ed.), Kluwer Academic Publishers, Boston, USA, p 405-431

MT Commission Report (T.C. Ulaştırma Bakanlığı 9. Ulaştırma Şurası, Havayolu Ulaştırması Komisyon Raporu) June 1998, Ankara, Turkey, pp 7-24.

NAVAZIO, L., ROMANIN-JACUR, G., 1998. The Multiple Connections Multi-Airport Ground Holding Problem: Models and Algorithms, Transportation Science, 32(3): 268-276

NEMHAUSER, G.L., WOLSEY, L.A., 1988. Integer and Combinatorial Optimization. John Wiley & Sons, Inc., New York, 763 p.

ODONI, A.R., 1987. The Flow Management Problem in Air Traffic Conrol, in Flow Control of Congested Networks (A.R. Odoni, L. Bianco, G. Szego, eds.), Springer-Verlag, Berlin, :269-288

RICHETTA, O., ODONI, A R., 1993. Solving Optimally the Static Ground-Holding Policy Problem in Air Traffic Control. Transportation Science, 27. 228-238.

RIFKIN, R. M., 1998. The Static Stochastic Ground Holding Problem. Master's Thesis, Massachusetts Institute of Technology, MA, USA, 46 p.

TERRAB, M., 1990. Ground Holding Strategies for Air Traffic Control. PhD Dissertation, MIT, MA, 292 p.

TERRAB, M., ODONI, A., R., 1993. Strategic Flow Management for Air Traffic Control. Operations Research, 41, 138-152

YILDIRIM, I., 1995. An Object-Oriented Simulation Model for Airport Traffic Control. MSc Thesis, Bilkent University, Ankara, 114 p.

CURRICULUM VITAE

Muhammed Ali ÜLKÜ was born in Adana, Turkey, in 1975. He attended Adana Anatolia High School and Adana Science High School hereinafter. He enrolled to Bilkent University, Industrial Engineering Department, awarded with full university scholarship and graduated in 1998. Afterwards, he worked in Efes Pilsen, Guney Brewery and Malt Co. Inc, as Productivity Expert until the end of 1999 and was hired in Cukurova University, Department of Industrial Engineering, as Research Assistant. His major interests comprise of Operations Research, Combinatorial Optimization, Network Flows, Productivity Analysis, Poetry and Philosophy of Science and Art.

APPENDIX A

Proof of Corollary 1: (Ball et al., 1999)

For the case $Q = 2$ and $T = 2$, the tree R will be constructed, thereby a showing clear exhibit of how to extend the construction to an arbitrary number of scenarios and time periods.[°] The primal problem is as follows:

$$
\begin{array}{llllllllllllll}
\text{Min} & \multicolumn{13}{l}{c_g * G_1 + c_g * G_2 + p_1 * c_a * W_{1,1} + p_1 * c_a * W_{1,2} + p_2 * c_a * W_{2,1} + p_2 * c_a * W_{2,2}} \\
& A_1 & & & +G_1 & & & & & & & & & = D_1 \\
& & A_2 & & -G_1 & +G_2 & & & & & & & & = D_2 \\
& & & A_3 & & -G_2 & & & & & & & & = D_3 \\
& -A_1 & & & & & +W_{1,1} & & -S_{1,1} & & & & & = -M_{1,1} \\
& & -A_2 & & & & -W_{1,1} & +W_{1,2} & & -S_{1,2} & & & & = -M_{1,2} \\
& & & -A_3 & & & & -W_{1,2} & & & & & & = -M \\
& -A_1 & & & & & & & +W_{2,1} & & & -S_{2,1} & & = -M_{2,1} \\
& & -A_2 & & & & & & -W_{2,1} & & +W_{2,2} & & -S_{2,2} & = -M_{2,2} \\
& & & -A_3 & & & & & & & -W_{2,2} & & & = -M
\end{array}
$$

In this problem, M is defined to be any sufficiently large constant (e.g., $M = \sum_{t=1}^{T} D_t$) and A is defined to be the constraint matrix associated with this problem.

To show that A^T is a network matrix, one must find a directed tree R on 16 nodes such that every column of A^T (equivalently, every *row* of A) corresponds to the characteristic vector of a path in R. Figure A.1. shows the required tree. Each arc of R is labeled with the corresponding primal variable. It can be verified that every row of A corresponds to the characteristic vector of a path in R. For example, the fifth row of A corresponds to the path formed by traversing arc $S_{1,2}$ backward, then arc $W_{1,2}$ forward, then arcs A_2 and $W_{1,1}$ backward. Figure A.2. shows the graph G spanned by the tree R. There, each arc is labeled with the associated dual cost coefficient (in the primal, the associated RHS element). This construction extends to an arbitrary number of scenarios and time periods.

[°] The general construction is explicitly formulated as a matrix transformation in (Hoffman, 1997), thus providing a more formal proof.

The dual problem can be recast in a more familiar form as a MCNFP by negating the objective function and multiplying the dual LP in equality form by N, where N is the node-arc incidence matrix associated with R with a single row deleted. Figure A.3. exhibits the resulting flow problem for $Q = 2$ and $T = 2$. The labeled arcs can accommodate either positive or negative flow, as they are associated with equality constraints in the original primal problem; the flow must be nonnegative on the unlabeled arcs, which are associated with dual slack variables. The unlabeled arc are zero-cost arcs and the unlabeled nodes are neither sources nor sinks.□

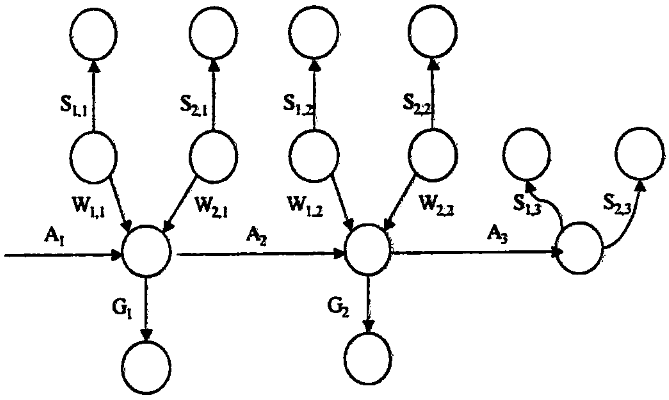

Figure A.1. The dual tree *R* of SSGHP

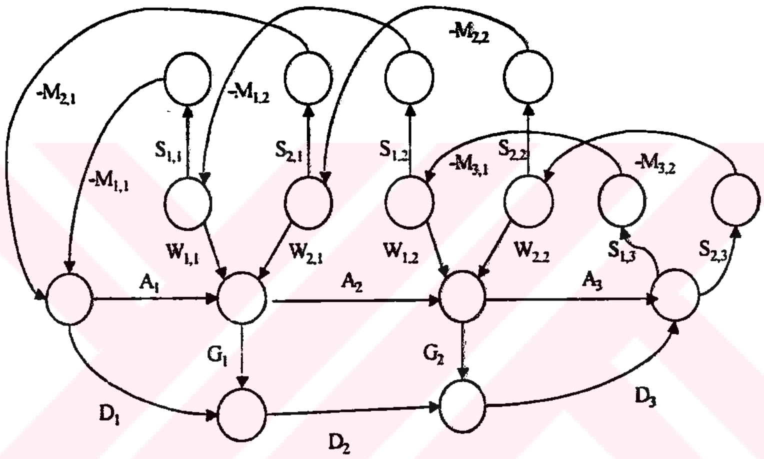

Figure A.2. The dual tree *R*, with columns of A^T

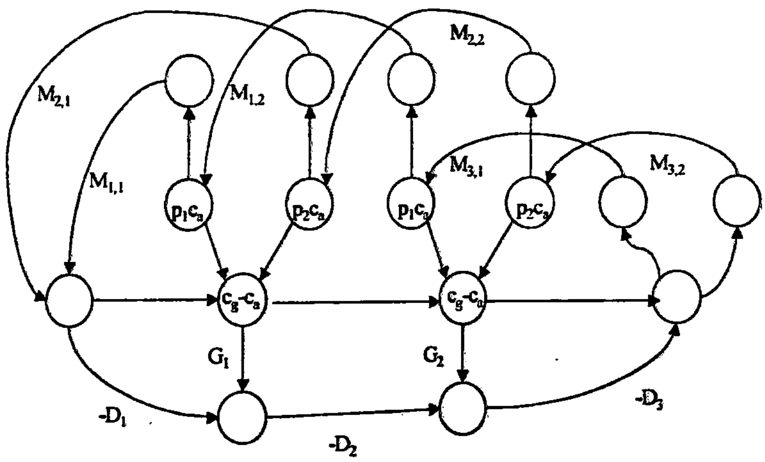

Figure A.3. The transformed MCNFP

APPENDIX B.1.

(ssghp.mod)

```
int nbS = ...;
range  S 1..nbS;

int nbT = ...;
range  T 0..nbT+1;


struct Plan {
   float+ A_t;
   float+ G_t;
   float+ W_st;
};

float+ D_t[T] = ...;
float+ M_st[S, T] = ...;
float+ c_r = ...;
float+ p_s[S] = ...;

var float+ A_t[T];
var float+ G_t[T];
var float+ W_st[S,T];


minimize

   sum ( t in 1..nbT)
        G_t[t-1] +

sum (s in S, t in 1..nbT)
        c_r*p_s[s] *W_st[s,t]


subject to {

   forall (t in 1..nbT+1)
  A_t[t] - G_t[t-1] + G_t[t] = D_t[t];

   forall(s in S, t in 1..nbT+1)
    -W_st[s,t-1]+W_st[s,t]-A_t[t] >= - M_st[s,t];

   forall(s in S, t in T)
   W_st[s,0] = 0;

   forall(s in S)
   W_st[s,nbT+1] = 0;
```

```
forall(t in T)
   G_t[0] = 0;

   forall(t in T)
   G_t[nbT+1] = 0;      };

Plan schedule[s in S, t in T] = <A_t[t],G_t[t],W_st[s,t]>;
display schedule;
```

(ssghp.dat)

```
nbS = 5;
nbT = 16;


D_t = [0,30,30,35,40,40,40,25,15,10,5,0,0,60,45,20,5,0 ];

M_st = [
           [0 20 10 10 10 25 20 30 35 35 35 45 45 40 40 40 40 1000 ]
           [0 30 20 20 20 20 20 20 20 30 35 35 35 35 0  0  0  1000 ]
           [0 35 35 30 25 25 25 30 30 25 25 20 20 0  0  0  45 1000 ]
           [0 10 10 10 10 20 20 15 15 20 20 10 10 10 10 0  0  1000 ]
           [0 60 55 55 50 50 55 60 60 65 65 70 70 50 50 65 60 1000 ]

           ];
c_r = 3 ;

p_s = [ 0.1 0.1 0.2 0.3 0.3];
```

APPENDIX B.2.

Optimal Solution with Objective Value: **1742.0** (c_r = 3)

Scenarios	period	A_t	G_t	W_st
1	0			
1	1	20	10	
1	2	10	30	
1	3	10	55	
1	4	15	80	5
1	5	20	100	
1	6	20	120	
1	7	20	125	
1	8	25	115	
1	9	25	100	
1	10	25	80	
1	11		80	
1	12		80	
1	13		140	
1	14		185	
1	15		205	
1	16		210	
1	17	210		
2	0			
2	1	20	10	
2	2	10	30	
2	3	10	55	
2	4	15	80	
2	5	20	100	
2	6	20	120	
2	7	20	125	
2	8	25	115	5
2	9	25	100	
2	10	25	80	
2	11		80	
2	12		80	
2	13		140	
2	14		185	
2	15		205	
2	16		210	
2	17	210		
3				
3	1	20	10	
3	2	10	30	
3	3	10	55	
3	4	15	80	
3	5	20	100	
3	6	20	120	
3	7	20	125	
3	8	25	115	
3	9	25	100	
3	10	25	80	
3	11		80	
3	12		80	
3	13		140	
3	14		185	
3	15		205	
3	16		210	
3	17	210		
4	0			
4	1	20	10	10
4	2	10	30	10
4	3	10	55	10
4	4	15	80	15
4	5	20	100	15
4	6	20	120	15
4	7	20	125	20
4	8	25	115	30
4	9	25	100	35
4	10	25	80	40
4	11		80	30
4	12		80	20
4	13		140	10
4	14		185	
4	15		205	
4	16		210	
4	17	210		
5	0			
5	1	20	10	
5	2	10	30	
5	3	10	55	
5	4	15	80	
5	5	20	100	
5	6	20	120	
5	7	20	125	
5	8	25	115	
5	9	25	100	
5	10	25	80	
5	11		80	
5	12		80	
5	13		140	
5	14		185	
5	15		205	
5	16		210	
5	17	210		