# AIRLINE DISRUPTION MANAGEMENT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

UĞUR ARIKAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
INDUSTRIAL ENGINEERING

SEPTEMBER 2014

Approval of the thesis:

**AIRLINE DISRUPTION MANAGEMENT**

submitted by **UĞUR ARIKAN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Murat Köksalan
Head of Department, **Industrial Engineering** ________________

Assoc. Prof. Dr. Sinan Gürel
Supervisor, **Industrial Engineering Department, METU** ________________

Prof. Dr. M. Selim Aktürk
Co-supervisor, **Industrial Engineering Dept., Bilkent Uni.** ________________

**Examining Committee Members:**

Prof. Dr. Meral Azizoğlu
Industrial Engineering Department, METU ________________

Assoc. Prof. Dr. Sinan Gürel
Industrial Engineering Department, METU ________________

Assoc. Prof. Dr. Oya Ekin Karaşan
Industrial Engineering Department, Bilkent University ________________

Assoc. Prof. Dr. Canan Sepil
Industrial Engineering Department, METU ________________

Assist. Prof. Dr. Ayşegül Altın Kayhan
Industrial Engineering Department, TOBB ETÜ ________________

**Date:** ________________

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name, Last Name: Uğur Arıkan

Signature :

# ABSTRACT

AIRLINE DISRUPTION MANAGEMENT

Arıkan, Uğur
Ph.D., Department of Industrial Engineering
Supervisor : Assoc. Prof. Dr. Sinan Gürel
Co-Supervisor : Prof. Dr. M. Selim Aktürk

September 2014, 200 pages

In this thesis, we deal with recovering airline operations in cases of irregularities. In schedule design phase, airlines generally generate tight schedules in order to efficiently utilize resources and deal with the high competition in the industry. However, irregularities in operations, also called *disruptions*, occur due to various reasons such as unscheduled aircraft maintenance, late appearance of crew members, bad weather conditions, congestions in airports, etc., and prevent the airline operate its original schedules. Airline Operations Control Centers (AOCC) are responsible for recovering the schedules of entities such as aircraft, crew members and passengers. These controllers are generally equipped with a set of recovery actions such as departure holding, flight cancellation and aircraft swapping. Due to the large size of airline networks, interdependencies between different entity types and real time solution requirement, integrated airline recovery problem is challenging. A common practice in the literature and industry is sequential approach which firstly recovers aircraft schedules and schedule recovery of the remaining entities are carried out accordingly. However, sequential approach results in high disruption and recovery costs. On the other hand, literature lacks from practical methodologies for the integrated airline recovery problem. We focus on the integrated problem in this thesis and propose a new network representation, exact approaches and heuristic approaches. Due to the increasing competition in industry, passenger convenience is attaining more and more importance. We also place a special emphasis on passenger recovery. Finally, we

manage to integrate cruise speed control option in addition to the common recovery actions and our experiments have shown that speeding up flights is a very beneficial action to mitigate delays, create new swap opportunities and maintain passenger and crew connections.

Keywords: Airline operations, integrated recovery, disruption management, irregular operations, passenger recovery, cruise speed control, conic quadratic mixed integer programming, connection network

# ÖZ

HAVAYOLU AKSAKLIK YÖNETİMİ

Arıkan, Uğur
Doktora, Endüstri Mühendisliği Bölümü
Tez Yöneticisi : Doç. Dr. Sinan Gürel
Ortak Tez Yöneticisi : Prof. Dr. M. Selim Aktürk

Eylül 2014 , 200 sayfa

Bu tezde, aksaklıklar karşısında havayolu çizelgelerinin tamiri ile ilgileniyoruz. Çizelge tasarım aşamasında, havayolları kaynaklarını etkin bir şekilde kullanabilmek ve sektördeki rekabet gücünü artırabilmek için genellikle sıkışık çizelgeler oluştururlar. Ancak, planlanmamış uçak bakım gereksinimi, bir mürettebat üyesinin gecikmesi, olumsuz hava şartları ve havaalanlarındaki yoğunluk gibi birçok etmenden dolayı operasyonlarda aksaklıklar oluşur. Bu aksaklıklar havayolunun planlanmış çizelgeleri uygulamasını engelleyebilir. Uçak, mürettebat ve yolcu gibi farklı elemanların çizelgelerini onarmak havayolu operasyon kontrol merkezlerinin sorumluluğundadır. Kontrolörlerin uygulayabileceği, kalkış zamanı erteleme, uçuş iptali ve uçak rotası değiştirme gibi bir grup onarım eylemi vardır. Havayolu ağlarının büyüklüğü, farklı eleman türleri arasındaki ilişkiler ve gerçek zamanlı çözüm gereksinimi nedenleriyle bütünleşik havayolu onarım problemi zorludur. Literatürde ve pratikteki genel uygulama önce uçak rotalarını tamir edip, mürettebat ve yolcu çizelgelerini yenilenen uçak çizelgelerine göre onaran ardışık yaklaşımdır. Ancak ardışık yaklaışım yüksek aksaklık ve onarım maliyetlerine yol açar. Öte yandan, literatürde bütünleşik havayolu aksaklık problemi için pratik çözüm yöntemi eksikliği gözlenmektedir. Bu tezde, bütünleşik probleme yoğunlaşarak, problem için yeni bir ağ gösterimi ile kesin ve sezgisel çözüm yaklaşımları öneriyoruz. Endüstride artan rekabet nedeniyle, yolcu memnuniyeti giderek önem kazanmaktadır. Bu nedenle yolcu çizelgesi tamirine çok önem veriyoruz. Son olarak uçuş hız kontrolünü diğer tamir eylemlerine entegre ediyoruz.

Deneylerimiz de bazı uçuşlarının hızlandırılmasının gecikmeleri azalttığını, yeni uçak rota değiştirme imkanları yarattığını, ve yolcu ve mürettebat çizelgelerindeki bağlantıların korunmasına yardım ettiğini gösteriyor.

Anahtar Kelimeler: Havayolu operasyonları, aksaklık yönetimi, bütünleşik çizelge onarma, yolcu çizelgeleri, uçuş hız kontrolü, konik karesel karmaşık tamsayı

*To my family and friends.*

# ACKNOWLEDGMENTS

It would not have been possible to write this doctoral thesis without the help and support of the kind people around me, to only some of whom it is possible to give particular mention here.

First and foremost, I would like to express my special appreciation and thanks to my PhD advisors Sinan Gürel and Selim Aktürk. I thank you for guiding me since the day I began working on this research and being tremendous mentors for me. Your contributions of time and brilliant ideas for this study are invaluable. I will always remember your comments, suggestions and guidance while I grow as a research scientist. You have taught me, both consciously and unconsciously, how good research is done. I am also grateful for your help and advices on my career whenever I need to make important decisions. Last, but by no means least, I thank for your friendship and support during my tough times.

I would like to thank my committee members, Meral Azizoğlu and Oya Ekin Karaşan. Your invaluable advices played an important role in the quality of this research. I am also grateful for the enjoyable lectures you have taught. These lectures had great help on developing the main ideas of this research. I would also like to thank the other two members of my defense committee, Canan Sepil and Ayşegül Altun Kayhan for letting my defense be an enjoyable moment, and for your insightful comments.

I would like to thank my professors Sinan Kayalıgil, Sedef Meral, and Serhan Duran for supporting me on my decision to join my department as a research assistant. Being in my department, I have been able to fully concentrate on my research and academic career. I am thankful to Sinan Kayalıgil and Serhan Duran for sharing your experience with me during our enjoyable journeys to Bursa.

I am grateful to my Ms advisor Ömer Kırca and Zeynep Müge Avşar for your advices during all my METU years and motivating me to become a researcher. I thank Cem İyigün for supporting me as a teaching assistant since the days I have began working in my department. I would not be able to develop myself as an instructor without your trust and invaluable suggestions. I would also like to thank all my instructors in my department for sharing their extensive knowledge with me.

I appreciate the kindness and support of my research assistant friends. Thanks to you, it has always been a great pleasure working in my department. I also thank Onur Tavaslıoğlu, Utku Can Kunter and Gökhan Ocak for being great teammates during our teaching assistant tasks. Special thanks to my friend Hüseyin Bacanak for helping me overcome the difficulties whenever I struggle during my research.

I thank my mom and brother for your unconditional love. I could not have succeeded without feeling your support and encouragement. I thank my handsome nephew Kenan for cheering me up. I especially thank to my beloved wife for her support and faith in me.

# TABLE OF CONTENTS

# LIST OF TABLES

TABLES

# LIST OF FIGURES

FIGURES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| AA | At-an-Arc |
| AD | Actual Delay |
| AN | At-a-Node |
| AOOC | Airline Operations Control Center |
| APR | Aircraft - Passenger Recovery |
| APR2 | Aircraft - Passenger Recovery without Aircraft Swapping Action |
| ARP | Aircraft Recovery Problem |
| AT | Arrival Time |
| Avg | Average |
| BTS | Bureau of Transportation Statistics |
| CN | Connection Network |
| CNGA | Connection Network Generation Algorithm |
| CNRA | Connection Network Revision Algorithm |
| CPU | Central Processing Unit |
| CT | Connection Time |
| CQMIP | Conic Quadratic Mixed Integer Programming |
| DAYOPS | Day of Operations Scheduling |
| Des | Destination |
| D-O | Destination - Origin |
| DPM | Disrupted Passenger Metric |
| DT | Departure Time |
| FDA | Flight Delay Approximation |
| FT | Flight Time |
| GDP | Ground Delay Program |
| FAA | Federal Aviation Administration |
| FIFO | First In, First Out |
| FS | Flight String |
| GRASP | Greedy Randomized Adaptive Search Procedure |

| | |
|---|---|
| IH | Isolation Heuristic |
| LN | Linear |
| LP | Linear Programming |
| MINLP | Mixed Integer Nonlinear Programming |
| MIP | Mixed Integer Programming |
| MRC | Maximum Range Cruise |
| NAS | National Airspace System |
| O-D | Origin - Destination |
| Ori | Origin |
| PB1 | Push-Back Recovery Plan |
| PB2 | Push-Back Recovery Plan Maintaining Passenger Connections |
| PDM | Passenger Delay Metric |
| PNGA | Partial Network Generation Algorithm |
| PW | Piecewise |
| PRP | Passenger Recovery Problem |
| SAT | Scheduled Arrival Time |
| SDT | Scheduled Departure Time |
| SOCP | Second-Order Cone Programming |
| TSN | Time-Space Network |

# CHAPTER 1

# INTRODUCTION

Air transportation is being used for about one hundred years. Since then this growing industry has been a very important sector due to the economical and operational aspects. [9] states that airlines alone generated more than $300 billion in revenues in 2002, a lean year, and carried out about 1.6 billion passengers. Moreover, number of passengers carried is expected to grow at an annual rate of 4% – 5% in the next 20 years according to most forecasts. Due to the high operational costs and complex airline networks, airline industry has been a very interesting sector for researchers. Especially after 1950s, operations research has contributed a lot in this industry.

Airlines are dealing with high capital, high labor and operating costs with low profitability margins due to the high competition in the industry. In addition, their operations are restricted with security and safety concerns. Therefore, airlines need to analyze a very complex and a large-scale system for planning their operations. Most airlines make use of sophisticated optimization tools while they are making decisions. However, there are still decisions, such as recovery decisions against disruptions that have not been automated yet. A large number of methodologies to increase the profitability of airlines has been proposed by researchers. There are more than 1,000 operations research papers published for decision support of air transport industry. On the other hand, high operating costs and enhancements in computer technology encourage studies in this industry. Airline operations that researchers are mostly interested are schedule planning and revenue management. In schedule planning phase, flights to be operated by the airline are determined and schedules of airline resources such as aircraft and crew members are created. The objective of the schedule planning problem is generally capturing the maximum demand with minimum operating costs to maximize the profitability of airline operations. Revenue management in airline industry, on the other hand, deals with determining fare classes and ticket prices with the aim of: *selling the right product to the right customer at the right moment at the right price*. More recently, there is an increasing attention in airline recovery problem, or *disruption management problem*, which deals with the irregularities that occur while carrying out the scheduled operations. The aim of disruption management problem is minimizing the disturbances that arise due to these irregularities. In this thesis, we deal with the airline recovery problem and propose real-time solution methodologies. In Section 1.1, we explain the phases of schedule planning in order to introduce airline operations. We introduce the reasons of irregularities in airline operations and recovery actions in Section 1.2. Since one of the main focus of this thesis is integrating cruise speed control action in airline recovery, we introduce related concepts in Section 1.3. We discuss our motivation and contributions in Section 1.4 and present the outline of this thesis in Section 1.5.

## 1.1 Schedule Planning

There are numerous decisions that an airline needs to make before operating its first flight. Firstly, flight regions, i.e. origin-destination (O-D) pairs, need to be determined. Next decision stage is related with number of flights that will be assigned to each O-D pair. Moreover, departure and arrival time decisions are to be made. Once all flights that will be operated are scheduled, the airline needs to assign aircraft and crew members to flights. The assignment decisions are constrained by many operational limitations. Interdependencies among these decisions can easily be observed. However, it is challenging to develop an integrated optimization model for schedule planning problem. Moreover, due to the huge size of airline networks and complexity of schedule the planning problem, integrated approaches suffer from intractability. In the literature and in practice, schedule planning is generally broken into four core problems, each solved sequentially ([11]). In this *sequential approach*, each subproblem is constrained by the optimal schedules of its preceding problems. Even though the subproblems are much smaller and simpler than the integrated problem, they are still challenging. Subproblems of the schedule planning problem are listed below:

1. Schedule Design
2. Fleet Assignment
3. Aircraft Maintenance Routing
4. Crew Scheduling
    i. Crew Pairing
    ii. Crew Assignment

### 1.1.1 Schedule Design

The first stage of the sequential approach is the schedule design stage. In this stage, flight legs that will be operated by the airline are determined. A *flight leg* is a flight with a determined:

- origin airport,
- destination airport,
- departure time, and
- arrival time.

Each flight leg can be considered as a market for the airline, and hence, determining flight legs actually corresponds to deciding the markets to be served. Since it greatly determines the market share that the airline will capture, schedule design stage is regarded as the most important part of airline schedule planning. Schedule design is carried out about one year before the first flight of a leg and generated schedules are greatly driven by market considerations.

Schedule designs of most major airlines operate on hub-and-spoke networks in order to reduce operating costs. A *hub-and-spoke network* is a type of airline schedule in which a great majority of flight legs depart from or arrive at a small subset of airports which are called *hubs* ([38]). Remaining airports are called *spokes*, and have relatively limited flights. This structure is advantageous also in disruption management since a great number of recovery opportunities can be generated in hubs.

The solution of the schedule design problem includes an optimal set of flight legs to be operated. These decisions affect aircraft and crew schedules, however, [11] points out the intractability of a single model to optimize all these decisions.

### 1.1.2 Fleet Assignment

Once flight legs to be flown are determined, airlines next assign fleets to flight legs. A *fleet* is a set of aircraft of the same type. Deciding the aircraft types of flight legs has a great impact on the profitability of the airline. Aircraft of different fleets have different seat capacities. Assigning a small aircraft to a flight leg with a greater demand results in lost customers; while unnecessary operating costs would be incurred in the opposite case. Different aircraft types may also have different speed capabilities and fuel efficiencies.

The inputs for the fleet assignment problem are:

- set of flight legs to be flown (solution of schedule design stage)
- number of available aircraft in each fleet
- cost of operating each flight leg with aircraft of each fleet
- turnaround time restrictions
- maintenance requirements of aircraft

*Turnaround time*, or *turn time*, is the minimum time needed after the arrival of an aircraft to be ready for its next flight. The length of the turnaround time may depend on aircraft type and airport. Maintenance requirement of an aircraft is related with the age and total distance flown by the aircraft in addition to its type. General approach to solve fleet assignment problem in the literature is to model the problem as an integer multi-commodity network flow model. In this model, fleets are regarded as commodities and the decision is to assign a fleet to each leg.

Generally, a *timeline network* is used for this model. In these networks, a node corresponds to time and location of a flight departure or a flight arrival. There are two types of arcs connecting these nodes: flight arcs and ground arcs. A flight arc corresponds to a scheduled flight leg and the arrival time is modified taking the required ground time into consideration. This time is required for disembarking passengers of the completed flight and embarking the passengers for the new flight, unloading and loading baggage and refueling. A ground arc, on the other hand, represents a connection between two consecutive flights.

Objective of the problem is generally generating a feasible assignment while minimizing a cost function including the following two cost terms.

- Operating cost: Total cost of operating all flight legs with aircraft belonging to the assigned fleet.
- Opportunity cost: Total cost of lost sales. This term is calculated by considering the excess demands for flight legs that exceed the capacity of aircraft belonging to the assigned fleet.

Solution of fleet assignment problem consists of fleet types assigned to each flight leg. Most fleeting models assume that flight times and ground times are deterministic; and hence, resulting schedules are sensitive to disruptions because factors such as congestion, weather conditions or new security policies may result in large variations in these expectations.

### 1.1.3 Aircraft Maintenance Routing

Once the fleet types of all flight legs are determined, the next stage is aircraft maintenance routing. In this stage, routings or rotations of each aircraft are determined. Individual aircraft are assigned to flight legs. Aircraft maintenance routing is generally divided into subproblems for each fleet type. While making individual aircraft assignments, decision makers are subject to the following constraints:

- Flights need to be operated with the available number of aircraft in the fleet.

- Flight departure times need to satisfy the determined time windows.
- Required turnaround time between each consecutive flights needs to be provided.

Another important consideration is the maintenance restrictions of the Federal Aviation Administration (FAA). These restrictions require a periodic aircraft service called *scheduled maintenance*. The frequency of services is a function of both flight hours (air time) and number of assigned flights.

The objective is to find aircraft rotations that satisfy these restrictions (feasibility problem). An *aircraft rotation* is a sequence of flight legs which starts and ends at the same airport (maintenance station). A rotation may take more than one day to fly. A *daily route*, on the other hand, is a subsequence of a rotation including flight legs that will be flown in the same day ([64]). Generally, the problem is modeled as a network circulation problem with side constraints.

Although airlines obey the maintenance restrictions of the FAA, experiencing mechanical problems during daily routes is not rare. Whenever such a problem occurs, the aircraft has to receive an *unscheduled maintenance*.

### 1.1.4 Crew Scheduling

Crews can be studied in two groups:

- *Cabin crews* are the employees which provide service to passengers.
- *Cockpit crews* are the pilots and hence, responsible for operating the aircraft.

Most studies on crew scheduling problem focus on scheduling cockpit crews for two reasons. Firstly, cockpit crews have greater salaries than the cabin crews. Note that second greatest cost term of the total operating costs of an airline corresponds to crew salaries while fuel cost is the leading term. Second reason is related with considerations about tractability. Concerning only cockpit crews reduce the complexity and size of the problem greatly ([11]). In this thesis, we will use crew members for mentioning about cockpit crews only.

In this stage crew members are assigned to the scheduled flight legs. Typically, pilots may only fly one type of aircraft. In this case, crew scheduling problem can be separated to subproblems corresponding to each fleet type. Moreover, it is a common practice, both in the literature and in industry, to break the problem into two sequential problems:

1. Crew pairing problem
2. Crew assignment problem

#### 1.1.4.1 Crew Pairing Problem

At this stage, pairings are determined. A *pairing* is a sequence of flight legs which starts and ends at the same city. Flights in a pairing may take multiple days. The objective of the problem is to generate minimum cost pairings while obeying restrictive work rules. Common work rules are listed below.

- Number of flights in a pairing is constrained by an upper bound.
- Air (flying) time of the flights in a pairing is constrained by an upper bound.
- Total length of the pairing, also called total *away-from-home time*, is constrained by an upper bound ([9]).
- Rest time between two consecutive pairings is constrained by a lower bound.
- Total air time in any 24-hour period is constrained by eight hours.

Modeling these restrictions is challenging. Moreover, the cost term of the objective function is generally defined by a nonlinear function of flying time, total working time and the time that the crew is away from home. A common approach is to solve a set partitioning problem after generating feasible pairings. Note that each pairing has a corresponding cost that may be calculated a priori, and hence, the objective function of this formulation is linear. However, the drawback of this approach is the huge number of feasible pairings. For a major U.S. airline, billions of feasible pairings may be generated. In order to handle this drawback, column generation techniques and metaheuristics are utilized in the literature.

#### 1.1.4.2 Crew Assignment Problem

The solution of crew pairing problem is a set of feasible pairings that covers all flight legs. Final stage of schedule planning is to assign pilots to these pairings. Generally, crew assignment is done using a bidline or preferential model. A *bidline* is a set of pairings that a crew flies in a month. Again, bidlines are subject to FAA and contractual rules. Due to the complexity of the problem, airlines generally seek feasibility instead of optimizing any objective function at this stage.

## 1.2 Irregular Operations

Airlines operate their flights with expensive resources in a highly competitive industry. Therefore, airlines generally create tight schedules in order to increase their profitability. These tight schedules rely on the assumption that the flight legs will be operated as planned. However, this optimistic scenario rarely occurs because of irregularities in operations. Irregularities may result in disturbances that are severe enough to prevent the airline continue with the original schedules. Such irregularities are called *disruptions*. In cases of disruptions, controllers need to make real-time recovery decisions. This thesis focuses on the problem of recovering aircraft, crew and passenger schedules, namely *airline recovery problem*, or *disruption management problem*.

Since disruptions damage aircraft and crew schedules, airlines incur operational costs. Passengers are also severely affected from disruptions and recovery actions. It is not easy to evaluate the cost of the effect of disruptions on passengers. However, passenger convenience is becoming more and more important in recent years. [12] state that in 2007, which is the last year of peak demand for air transportation before economical downturn, cost of arrival time delays to airlines is estimated to be $19 billion (U.S. Congress Joint Economic Committee, 2008). In the following year, profits of airlines were estimated to be only $5 billion (Air Transport Association, 2008). These statistics show the significance of disruption costs in airline industry. Joint Economic Committee report states that the economic cost of time lost by passengers due to the irregularities in airline operations is estimated to be $12 billion. On the other hand, Air Transport Association estimates the cost of passenger delays as $5 billion. The huge difference in these estimates results from different methodologies used and reveals the requirement for more accurate estimates to evaluate the cost of lost time by passengers in air transportation. However, significance of disruption costs in airline revenues can still be observed. In this thesis, we place a special emphasis on passenger recovery. We integrate passenger recovery decisions with the recovery of aircraft and crew schedules. Moreover, we try to contribute by proposing realistic passenger delay cost formulations.

### 1.2.1 Reasons of Disruptions

There are various reasons that may result in a disruption. Some of them are related with airline resources while the remaining may be caused by the outer system. Most common disruption types are listed below.

- *Unscheduled maintenance*. As explained in Section 1.1.3, each aircraft attends periodic maintenances that are scheduled with respect to the restrictions of the Federal Aviation Administration (FAA). However, unexpected mechanical problems may still occur. In these cases, aircraft experiencing such problems needs to have an unscheduled maintenance and during this time this aircraft will be unavailable. Severity of disruption is related with the duration of the maintenance. Some problems may be resolved in half an hour while more serious ones may take days. Since aircraft is scarcest resource of an airline, absence of a single aircraft for a period of time may result in great disruption costs.
- *Crew delays.* Crew members that are scheduled to a flight may arrive late, or even may not show up due to health problems. The prior one may result in flight departure delays, while the latter one may result in flight cancellations.
- *Problems in ground operations.* Problems in ground operations may be experienced due to the lack of ground resources. This may result in increased luggage loading time or fueling time. Therefore, ready time of the aircraft for its succeeding flight will be delayed. If there is not enough slack time between the flights, these problems may result in departure delays.

- *Poor weather conditions.* During takeoffs and landings, there has to be a separation between aircraft. Poor weather conditions may result in a decrease in sight and an increase in the *separation distance*. Less than scheduled number of aircraft may land or take off. In such cases, departure and arrival times of flights may be delayed. Air time of flight legs are generally assumed to be deterministic. However, poor weather conditions may result in longer flight times than scheduled, and again arrival times may be delayed. Finally, extremely bad weather conditions may result in airport closures for a period of time. If the closed airport is a hub, a very large number of flights need to be cancelled. This results in great disruption costs and recovery process requires a great effort.

- *Congestions at airports.* With increasing air traffic, congestions may be observed in airports, especially in hubs. Congestion in a hub may result in delays or cancellations of a great number of flights.

- *Delay propagation.* Delay propagation may be regarded as the consequence of some other disruption. Arrival delay of a flight may result in departure delays of *downstream* flights. Delay may propagate through the schedules of aircraft and crew members. Moreover, late arrival of connected passengers may also result in departure delays of their succeeding flights.

In order to understand the frequency of disruptions, we carry out a data analysis based on the data provided by Bureau of Transportation Statistics (BTS) (http://www.bts.gov). BTS publishes the on-time performance data of all scheduled flights. Public data contains information about scheduled departure times and arrival times together with the realized departure and arrival times. In our data analysis, we work with the data of flights of a major U.S. airline operated between January, 2011 and March, 2013. In Figure 1.1, we observe the behavior of arrival delay severities on a timeline. We observe that distribution of arrival delay severities in different time of year does not experience a significant change. In Figure 1.2, overall distribution of arrival delay severities over this 15-month period is illustrated. In airline industry, it is common to consider flights that do not experience an arrival delay greater than 15 minutes as on-time flights. In this sense, this analysis reveals that about 19.1% of all flights experience an arrival delay.

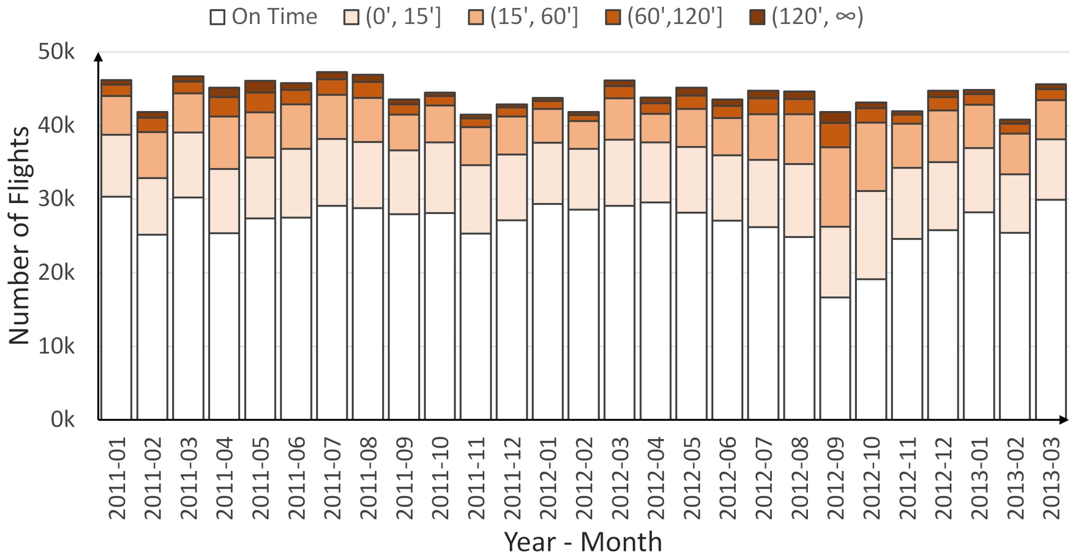


Figure 1.1: Behavior of arrival delays of a major U.S. airline on a timeline.

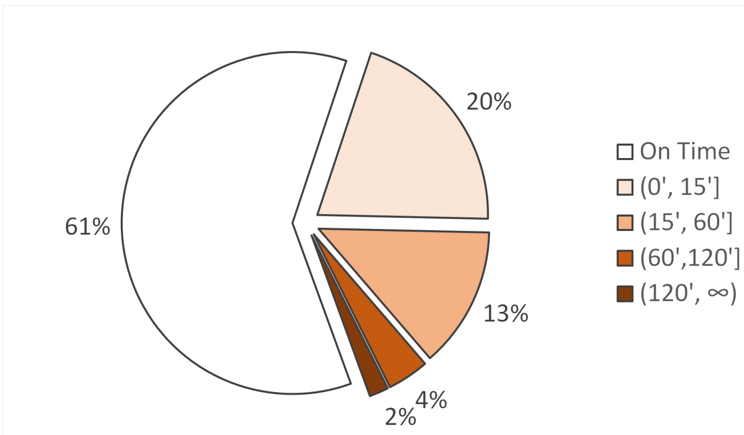


Figure 1.2: Distribution of flights of a major U.S. airline with respect to severities of arrival delays.

Reasons of arrival delays are classified into five categories by BTS:

- Carrier
- Weather
- National Airspace System (NAS)
- Security
- Late aircraft

We perform a data analysis to understand the effect of the delay sources proposed by BTS. Distribution of the number of delayed flights, total delay experienced and average delay per flight in January, 2011 are illustrated in Figure 1.3. We observe that carrier, NAS and aircraft related delays construct about 93.3% of all delayed flights. Security delays, on the other hand, result in most severe arrival delays. Overall average delay experienced is about 33 minutes per delayed flight.

Another important statistics provided by BTS is the number of cancelled flights. Cancellations are not as common as arrival delays. However, they result in greater disturbances and disruption costs. Within the investigated 15-month period, we observe that about 2.1% of all scheduled flights is cancelled.

In addition to our data analysis, [53] state that about 30% of the flights of a major U.S. airline is delayed in year 2000. Moreover, the authors report that the percentage of cancelled flights is about 3.5%. All these analyses reveal the importance of disruption management in air transportation.

### 1.2.2 Preventive Recovery Actions

Preventive actions against disruptions may be taken during schedule planning phase. Many researchers take the uncertainty in airline operations into account and propose methodologies to generate schedules that are less sensitive against disruptions, namely *robust schedules*. Main strategy in creating robust schedules is to add *slack times* between consecutive flights. Total slack time is limited since flight legs to be

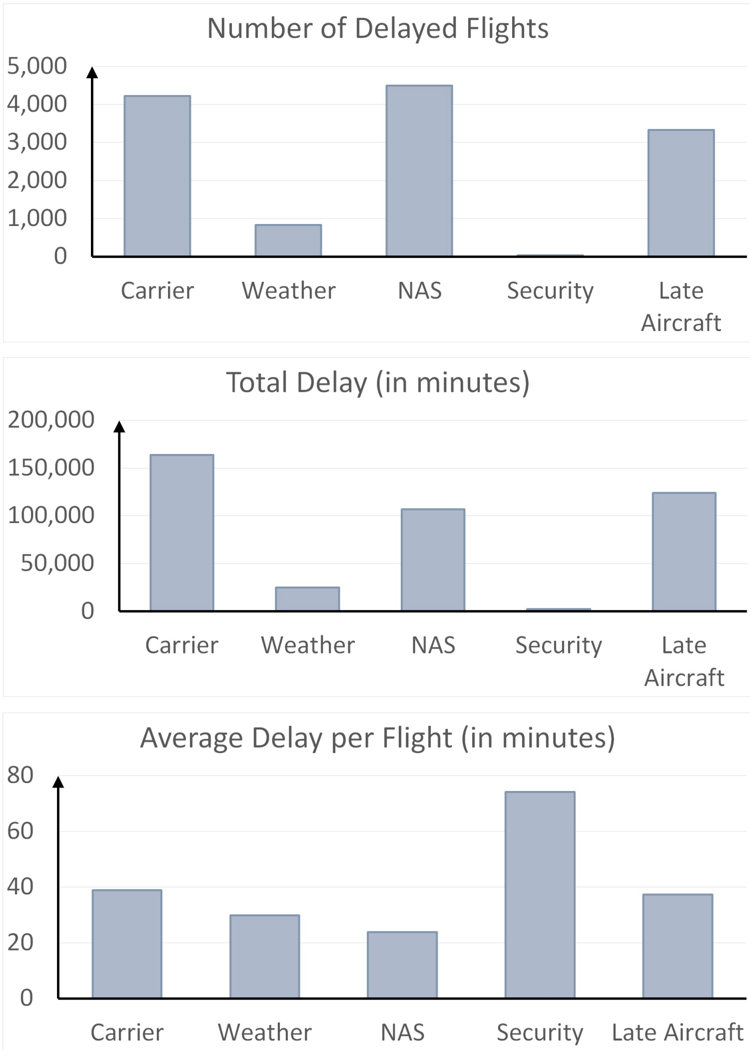


Figure 1.3: Severities of arrival delays of a major U.S. airline with respect to reasons proposed by BTS.

operated, number of available aircraft and crew members are determined. Proposed approaches try to distribute total slack time in the optimal way so that delays due to forthcoming disruptions may be absorbed and delay propagation is minimized.

Another approach is related with structure of the schedule design. In hub-and-spoke networks, airlines mostly prefer to generate aircraft routings having several *out and back* flight legs. In particular, if a flight departs from a hub and arrives at a spoke, the next flight assigned to this aircraft arrives at the same hub. Crew schedules are designed in the same manner. With this schedule design, any such flight pairs can be cancelled without affecting the downstream flights in aircraft and crew schedules.

### 1.2.3 Real-Time Recovery Actions

In this section, we explain real-time recovery actions which correspond to the alternative courses of actions of the proposed solution approaches in this thesis. In cases of disruptions, airlines need to take action in order to reduce the costs resulting from disruptions. Generally, controllers, or *dispatchers*, in Airline Operations Control Center (AOCC) are responsible for recovery actions. Dispatchers in AOCC continuously track all operations and make the final decision in the case of a disruption ([25]). The actions to be taken against disruptions are generally decided manually by these controllers based on their experience and intuition. This intuitive decision making process may be understood by the following questions to which dispatchers try to find answers ([77]). Moreover, we make use of this sequential process in the *Isolation Heuristic* (explained in Chapter 5) that we propose to limit the problem size while dealing with huge instances.

- Can all scheduled flights be made with the available aircraft?
- If not, which flights should be cancelled?
- What are the new departure times of the flights which will minimize passenger delay throughout the network?
- What is the new aircraft rotation plan (which aircraft of different routings will be swapped)?

- Is it possible to make the planned flights with the available crews?
- What is the crew rotation plan?
- Does the new schedule satisfy maintenance restrictions?

Dispatchers are equipped with a set of recovery actions to deal with disruptions. Most common recovery actions are listed below.

- *Holding departure times.* Departure time of flights may be delayed to some extent. Maximum allowed delay is determined with the available time slot assigned to the airline in the corresponding airport. Dispatchers may delay the departure time of a flight due to various reasons. First of all, an unscheduled maintenance of an aircraft may delay its first flight. In this case, dispatchers may assign this flight to another available aircraft or may need to cancel several flights. However, if the problem is not that severe, dispatchers may decide to hold the departure time of the flight until the end of this unscheduled maintenance, so that original schedule may be executed as planned except that several flights experience arrival delays. Similarly, in cases of late arrivals of crew members or connected passengers, dispatchers may delay departure time of the corresponding flights. It is important to note that, holding the departure time of a flight affects downstream flights, as well. In particular, delay propagates through the downstream flights in the schedules of the aircraft and crew members that are planned to operate the delayed flight. As explained in Section 1.2.1, this effect is called as delay propagation. Moreover, delayed flights may result in insufficient connection time for connected passengers, as well. In this case, dispatchers may decide to cancel the flights of these passengers or try to reallocate them. Otherwise, experienced departure delay also propagates through passenger itineraries.

- *Flight cancellation.* Flight cancellation is explained as a disruption type in Section 1.2.1. However, dispatchers may use flight cancellation as a recovery action, as well. Flight cancellation is a very costly and an undesired recovery action. Aircraft, crew and passenger schedules are severely affected from cancellations. On the other hand, delay propagation may also result in extreme disturbances, and hence, in some cases, cancelling a few flights may be preferred.

- *Swapping aircraft and crew members.* One of the most common and beneficial recovery actions used in practice is swapping the routings of two aircraft. In cases of delays in ready time of aircraft or late arrival of aircraft due to delay propagation, dispatchers may search for swap opportunities. In general, routings of two aircraft that arrive at a common airport within the recovery horizon may be swapped. Whenever their routings are swapped each aircraft continues with the routing of the other one. This recovery action may help to prevent delay propagation. Moreover, if the final destination of both of the swapped aircraft is the same airport, this recovery action results in minimal operating costs. On the other hand, if they end up in different airports at the end of the day of operations, dispatchers may need to relocate them. Structure of hub-and-spoke networks helps to generate numerous swap opportunities at hubs since at any given time dispatchers may find many available aircraft on ground. Dispatchers may also utilize this recovery action for crew members.

- *Rerouting aircraft and crew members.* Swapping is only a subset of possible rerouting actions. Actually, schedules of aircraft and crew members may completely be altered in cases of disruptions. However, due to the complexity of schedules, rerouting action is not common in practice and in the literature. Recall that all flights have already assigned aircraft and crew members. If the dispatchers decide to reroute an aircraft through a completely different set of flights, number of aircraft that are affected from this decision may be as many as the number of flight legs in the new schedule. Swapping is a preferred action since the number of affected aircraft is bounded by two. In Chapter 4 and Chapter 5, we propose methods to allow evaluating all rerouting opportunities and generate the complete solution space.

- *Spilling passengers.* Whenever the airline cancels the tickets of passengers, it is said that these passengers are spilled. Flight cancellation may most probably result in passenger spills. On the other hand, late arrivals of flights may result in insufficiency in the required connection time for passengers that change their aircraft. Note that passengers having at least two flights in their itineraries may be spilled due to insufficient connection times. Major airlines generally operate with one and two flight itineraries, while some long trips may also include three flights. Passenger spilling is an undesired recovery action. Generally two cost terms are incurred: cost of lost sales and cost of passenger inconvenience. The latter one is probably more important than the prior one due to the high competition in the industry.

- *Reallocating passengers*. Passengers with cancelled flights or experiencing insufficient connection times may be reallocated to later flights. Passengers may experience arrival delays, however, passenger inconvenience is less compared to spilling action. Therefore, dispatchers prefer to reallocate passengers whenever available. Main consideration for this recovery action is the number of empty seats in available flights.

- *Ferrying aircraft*. Due to the experienced disruptions or as a consequence of the recovery actions, some aircraft may end up the day at different airports than their expected locations. In such cases, dispatchers may need to relocate these aircraft to their expected airports so that original schedules may be resumed in the next day. The action of flying the aircraft to its expected location without passengers is called ferrying. This recovery action is not a desired recovery action due to its high operating costs.

- *Deadheading crew members*. Similar to ferrying action, crew members may also be relocated in order to recover the schedules. Deadheaded crew members are transported as passengers.

- *Using standby aircraft*. A standby (also called surplus or spare) aircraft, especially located at hubs, may provide valuable recovery actions to prevent delay propagation and cancellations. However, since keeping an idle aircraft is very costly, this recovery action is not preferred by most airline companies.

- *Calling up reserve crew members*. This action resembles using standby aircraft action and it is relatively less costly. Airlines generally locate reserve crew members at major airports for quickly responding crew-related disruptions.

- *Cruise speed control*. Airlines typically operate aircraft at their most economical speeds, usually called *maximum range cruise (MRC)* speeds. MRC speed is always lower than the maximum speed that the aircraft may reach. The difference between the maximum speed and MRC speed depends on the O-D pair of the flight and technological properties of the operating aircraft. In cases of disruptions, speeds of several flights may be increased to reduce arrival delays. It is important to note that, reducing the arrival delay of a flight affects the arrival delays of downstream flights as well. In other words, delay may be mitigated in the same manner it propagates through the flight schedules of related aircraft, crew members and passengers. Moreover, being able to use variable operation times (flight times) increases the number of swapping, rerouting and reallocating opportunities since the solution space is enlarged. However, if the speed of a flight exceeds its MRC speed, fuel consumption increases, and an additional fuel cost is incurred. Relation between the deviation from the MRC speed and fuel consumption is expressed with a nonlinear function. Therefore, this recovery action is excluded in most airline recovery methodologies proposed in the literature. On the other hand, despite the complexity it adds to the formulation, in this thesis we try to integrate cruise speed control action with common recovery actions and analyze its performance.

Dispatchers need to take recovery actions in a very short period of time. According to many authors in the literature, solution times of airline recovery problem needs to be limited by about five minutes since recovery decisions made later than this upper bound may not be implementable due to the dynamic nature of airline operations. There are also some studies that limit decision making process with about 30 minutes. Explanations of the recovery actions reveal the dependencies of decisions related to different entity types. In this thesis, we use *entity* to refer to individual aircraft, crew members and passengers. For instance, while swapping the routings of two aircraft connected passengers must be considered. Similarly, since the flight sequences of aircraft and crew members do not overlap, decisions related to one of these entity types directly affects the decisions related to the other one. Therefore, decisions related to all entity types need to be integrated in a methodology that seeks for the global optimum. However, major U.S. airlines operate more than a thousand flights in a single day of operations. Due to huge size of airline networks, complexity of the problem and short solution time requirements, integrated recovery problem is challenging. Therefore, it is common to follow a *sequential approach* while recovering the schedules both in practice and in the literature. The sequential recovery approach resembles the sequential approach used in schedule planning. Considering the aircraft as the scarcest resource, aircraft routings are repaired firstly. Resulting aircraft routings are used as inputs for crew and passenger recovery problems.

## 1.3 Cruise Speed Control

Duration of a flight is separated into six phases:

- take-off and initial climb,
- climb,
- cruise,
- descent,
- holding, and
- approach.

Cruise phase corresponds to the greatest portion of the flight time, especially for long trips. Even it is the most fuel economical phase, majority of fuel is consumed at this stage. Fuel cost optimization of cruise phase involves determination of the optimal altitude and optimal cruise speed. With a given a flight level which may be determined by the airline or imposed by air traffic control, speed is the only remaining parameter that requires selection ([3]). Being the longest phase of a flight, cruise speed has a significant effect on flight time. In [2], total cost of a flight is reported to be the summation of the following three terms:

- fixed cost independent of flight time,
- cost of consumed fuel,
- cost related with the flight time.

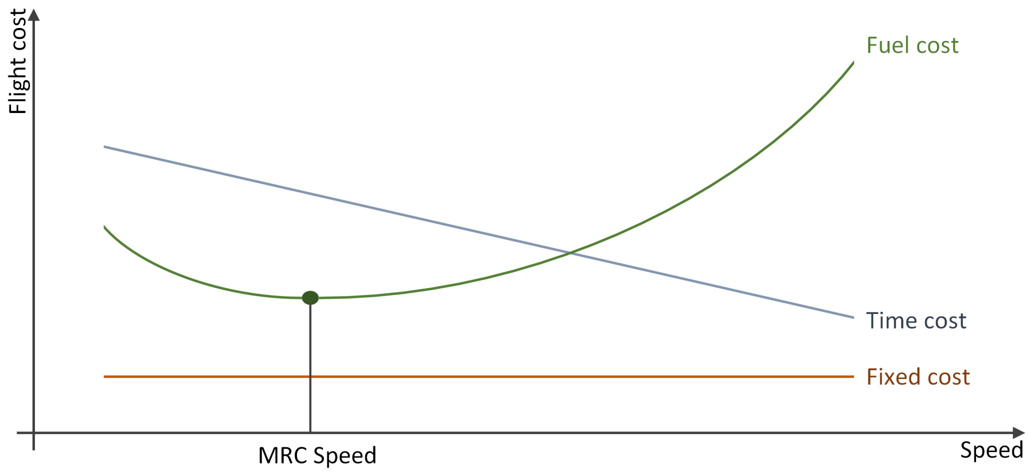


Figure 1.4: Behavior of cost terms with respect to cruise speed according to [2].

[2] illustrates the behavior of these cost terms as displayed in Figure 1.4. Time related cost is represented by a linear function which does not involve the effect of airline networks. Moreover, passenger inconvenience is ignored. Placing a special emphasis on passenger recovery, we propose formulations that try to evaluate time-related costs as accurate as possible. On the other hand, we observe a nonlinear and convex relationship between fuel consumption, or fuel cost, with the cruise speed. Minimizer of this function (speed that minimizes fuel consumption) is called maximum-range cruise (MRC) speed. Most airlines operate their flights at MRC speeds. This figure displays a simplified tradeoff between the increased fuel cost and decreased time-related cost whenever the cruise speed of a flight is increased. Reducing the arrival delay of a flight by speeding up results in delay mitigation in the downstream flights, as well. Furthermore, new swap and rerouting opportunities can be created by increasing the speeds of some flights, even if they are not disrupted. Therefore, it does not seem possible to evaluate this tradeoff without considering the entire airline network. We integrate cruise speed control action in our formulations in Chapters 3 and 4, and optimize cruise speeds with other recovery actions simultaneously, while coping with airline recovery problems.

## 1.4 Motivation and Contributions

Air transportation industry has been a very interesting research area, especially for operations research field, after 1950s. One of the main reasons of this attention is the high capital together with high labor and operating costs. Even minor improvements in airline operations may result in great impacts on profitability. Moreover, the complexity and size of airline systems make airline planning problems difficult to solve. Despite the huge number of studies, there are still decisions that have not been studied sufficiently, and hence, not automated yet. Intuition and experience of controllers in airlines plays an important role in making these decisions. Therefore, there is still an increasing demand for new optimization methods for airline processes.

This dissertation focuses on the *airline recovery problem*, or *disruption management problem*. Airline recovery problem is relatively less studied than schedule planning problems. Majority of the studies in the literature follows the sequential approach used in schedule planning problems, and hence, breaks the recovery problem to subproblems corresponding to different entity types, such as aircraft, crew members and passengers. The aim of this approach is to deal with the complexity of the recovery problem and limited solution times. In the literature, different time limitations are proposed for recovery actions varying between three to 30 minutes. Integrated formulations generally suffer from reaching the optimal, or even feasible, solutions for practical-size instances within this time limit.

On the other hand, sequential recovery decisions result in suboptimal solutions that incur high operating and passenger inconvenience costs. Integration of aircraft and crew recovery has been studied due to the effect of aircraft recovery decisions on crew schedules in the sequential approach. More recently, with the increased competition in industry, there is an increasing number of studies that integrate passenger recovery with aircraft recovery. Finally, there are several studies that try to integrate all entity types. In Chapter 3, we initially integrate passenger recovery with aircraft related decisions. In Chapter 4 and Chapter 5, we propose solution approaches to integrate all entity types.

Integrated recovery formulations in the literature are based on two traditional problem representations: time-space networks and flight strings. However, these studies also suffer from intractability in real time for practical sized instances. Generally, sequential approach or cost approximations are proposed to maintain the tractability of large instances. Main drawback of these representations is the huge problem size required to represent the solution space. In this thesis, we propose an alternative problem representation that utilizes connection, also called activity-on-node, networks. Connection networks have a significant advantage in size of the problem representation. Main reason of the reduction in problem representation is being able to represent time-related decisions with continuous variables instead of binary variables. Unlike traditional approaches, we model schedules and recovery actions of all entity types with connection networks. Interdependencies among different entity types are well expressed on the common set of flight nodes. Therefore, we manage to easily integrate aircraft, crew and passenger recovery processes. Since all entity types are represented with connection networks having the same structure, other entity types such as luggage can also be integrated with the proposed problem representation.

As explained in Section 1.2.3, cruise speed control action adds complexity to the recovery problem due to the nonlinear relationship between cruise speed of flights and fuel consumption. Since the integrated recovery problem is already complex, there are only a few studies that integrate cruise speed control action with common recovery actions. Moreover, there does not exist any study that evaluates all possible cruise speed options without discretization. Our alternative problem representation enables to represent cruise speed decisions with continuous decision variables. Our formulations include a nonlinear cost term in the objective function for evaluating the tradeoff between the additional fuel cost and reduction in disruption costs. We linearize the objective functions by introducing additional constraints and finally show that these constraints are second order cone programming (SOCP) representable. Our experimentations have shown that proposed reformulation is efficient and real time solutions can be proposed to large-sized problems. Moreover, we report in Section 4.5 that the reduction in total disruption and recovery costs provided by integrating cruise speed control action is significant.

This study also places a special emphasis on passenger recovery. As mentioned above, passenger convenience is becoming more and more important in recent years due to the high competition in airline industry. Passengers are severely affected from recovery decisions made by the sequential recovery process and this results high passenger inconvenience and cost of goodwill loss. On the other hand, proposed integrated approaches generally do not utilize all passenger reallocation opportunities. In our proposed problem representation, we model all entity types in the same manner and evaluate all possible recovery actions. Therefore, passenger recovery is completely integrated.

Another challenge in passenger recovery is related with the difficulty in estimating arrival delay costs of passengers. Modeling total passenger delay cost is complex due to passenger reallocation and spilling decisions. Most integrated approaches propose an approximation method which ignores these decisions and assumes that passengers are transported as planned. It is common to express the relationship between the experienced arrival delay and passenger delay cost with a linear function. Some authors believe that passenger delay cost can be represented by a nonlinear convex function of the experienced arrival delay. We propose a piecewise cost function to estimate this nonlinear relationship in addition to the linear cost function. Moreover, we manage to solve large instances by calculating the actual delays of each individual passenger instead of using the approximation method. In order to calculate actual arrival delay experienced by each passenger while utilizing passenger reallocation opportunities, each passenger needs to be modeled explicitly. To the best of our knowledge, this study is the first to model each passenger explicitly without aggregating the passengers in the same fare class of an itinerary. In addition to enabling actual delay calculation, explicit modeling has the advantage of assigning different delay and spill cost parameters to each individual.

Finally, we propose an alternative heuristic approach for making recovery decisions for huge instances in real time in Chapter 5. There are very few heuristic approaches in the literature for airline recovery problem, and to the best of our knowledge there is no proposed heuristic approach for the integrated recovery problem. Note that good recovery decisions that can be made in five minutes is much more valuable than the optimal solution that can be reached within a couple of hours in airline industry. As explained above, our proposed problem representation has an advantage of expressing the interdependencies of different entity types through the set of flight nodes.

Utilizing this beneficial structure of the connection networks, we propose *Isolation Heuristic* that limits the solution space. We try to mimic the intuitive decision making process of dispatchers in the sense that they try to recover schedules with minimal disturbations on the original schedules. Therefore, a great majority of the entities follow their original schedules in the proposed recovery. Proposed approach cleverly tries to isolate these entities and find good recovery solutions over a smaller solution space. The heuristic is equipped with mechanisms to control the tractability, stability and quality of the solutions. The heuristic is independent of the optimization methodology. The algorithm only reduces the solution space in a very short time, and the resulting solution space can be optimized with any methodology.

### 1.5 Organization of Dissertation

This dissertation consists of six chapters.

Following this introduction chapter, a literature review on airline operations is presented in Chapter 2. Literature review is investigated in two sections. In the first section, important studies on scheduling problems are listed, while in the second one we focus on airline recovery approaches.

In Chapter 3, an approach that integrates passenger recovery with aircraft recovery, and integrates cruise speed control action with common recovery actions is presented. Proposed approach is illustrated with a small-sized numerical example. An optimization model is proposed to solve aircraft and passenger recovery problem. Since cruise speed control action is utilized, proposed formulation is a mixed integer nonlinear programming model. An efficient reformulation scheme is proposed to represent the problem with a second order cone programming model. The approach is experimented with the flight data of a major U.S. airline. At the end of the chapter, we conclude that the approach is promising in the sense that it can provide real time solutions to practical size instances and cruise speed control action provides significant savings in disruption and recovery costs.

In Chapter 4, an alternative problem representation is proposed which allows easily integrating aircraft, crew members, passengers, and any other entity type. Based on this representation, a network based formulation is proposed to optimally solve the integrated recovery problem. Mathematical formulation utilizes all possible recovery actions to guarantee global optimality. Due to the nonlinear tradeoff between increased fuel consumption and reduction in disruption costs, a reformulation similar to the one introduced in Chapter 3 is proposed. Moreover, two important preprocessing approaches that reduce problem size without sacrificing optimality are proposed to enhance solution times. Finally, four alternative methodologies to evaluate arrival delay cost of passengers are developed. Experimentations of the proposed formulation, preprocessing approaches and passenger delay cost functions are presented at the end of the chapter.

In Chapter 5, a heuristic approach named as *Isolation Heuristic* is proposed. Aim of the heuristic approach is controlling the tradeoff between the tractability of instances and quality of the solutions that can be reached in real time. Moreover, a solution procedure is proposed that quickly updates connection networks with respect to the locations of the entities at the moment a disruption occurs, quickly isolates the solution space and finds fast and good recovery decisions. The heuristic isolates the solution space regardless of the optimization model. In experimentations, four optimization models proposed in Chapter 4 are used to test the performance of the solution procedure.

In Chapter 6, a summary and conclusion of this thesis is presented. Future research directions are also explained in this chapter.

# CHAPTER 2

# LITERATURE REVIEW

In this chapter, we first summarize studies in schedule planning problems and recovery problems in Section 2.1 and Section 2.2, respectively. Note that, these problems are related since recovery process is applied on the scheduled operations. In Section 2.3, we classify the underlying problem representations of the solution approaches in the literature. We investigate the studies that deal with cruise speed optimization in Section 2.4. Finally, we summarize expressions for calculating passenger-related disruption costs proposed in the literature in Section 2.5.

## 2.1 Schedule Planning Problems

We investigate literature on schedule planning problems in four categories:

1. Schedule Design
2. Fleet Assignment
3. Aircraft Maintenance Routing
4. Crew Scheduling

These categories correspond to the stages of schedule planning in the sequential planning approach. In many studies reviewed in this section, at least two of these problems are integrated.

### 2.1.1 Schedule Design

We present few studies in this section since most studies that deal with schedule design problem are integrated with at least one of the later-stage problems.

[32] focus on airlines operating on hub-and-spoke networks. The authors point out the reduction in complexity of system analysis with this network structure. Proposed approach aids in flight schedule selection and route price determination. An important contribution of the study is the proposition of an expression which calculates demand for each route as a function of the service quality and prices of all derived routes. A heuristic approach is proposed to find the flight schedule and route prices with the objective of maximizing the profit of the airline. Decision making process takes the prices of other airlines into consideration and optimizes against the competitors' decisions.

[49] study schedule design problem in charter airlines, in particular. Characteristics of charter networks are exploited in developing exact and heuristic service network design models and algorithms for the problem. The authors manage to express many of the constraints and complexities of schedule design problem with a specially design network so that the problem is formulated as a classical network design problem. Proposed model can achieve quality solutions in long solution times. They develop a fast solution approach for the special case having a single fleet type. Then, the single-fleet model is adapted to develop a heuristic approach that can handle multiple-fleet problems.

### 2.1.2 Fleet Assignment

Fleet assignment problem has gained a great attention of researchers due to its significance in the profitability of airline companies.

[30] study fleet assignment problem in hub-and-spoke networks. The problem is formulated as an integer programming model unlike the traditional multi-commodity network flow problems. For solving the proposed model, a Lagrangian relaxation is utilized together with heuristics for converting the Lagrangian solutions into primal

feasible solutions, and for improving the obtained solutions.

[18] define Demand Driven Dispatch, $D^3$, as an operating concept addressing the problem of capacity assignment to flight schedules to meet fluctuating market needs. The authors suggest that dynamic aircraft capacity assignment may provide improvement in operating profits. Proposed approach is based on the fact that a demand forecast improves as departure approaches, and hence, dynamic assignment may provide better matches between demand and assigned capacity. A time-space network representation is proposed, and the problem is formulated as a multi-commodity network flow problem with side constraints.

[42] deal with a basic daily, domestic fleet assignment problem. Steps taken to find efficient solutions to the problem are chronologically presented. Problem is represented with a time-expanded network. The authors formulate the problem as a large-scale multi-commodity network flow problem with side constraints. Due to the following reasons listed by the authors, solution times of these problems can be very long.

- These problems are severely degenerate, and hence, applying standard linear programming techniques leads to poor performance.
- The large number of integer variables can make finding optimal integer solutions difficult.

The authors attack to this challenging problem using several methods such as interior-point algorithm, dual steepest edge simplex, cost perturbation, model aggregation, branching on set-partitioning constraints and prioritizing the order of branching. Experimentations have shown that the proposed algorithm finds the solutions more than two times faster than a standard LP-based branch-and-bound code.

[85] propose a solution approach for integrated schedule design and fleet assignment problem. The study aims to present a framework that helps airlines adjust their flight schedules and fleet assignments in response to expected changes in market demand conditions in the near future. Integrated problem is represented on a multi-fleet time-space network. Several strategic models are formulated as multi-commodity network flow problems. For solving these mathematical formulations, the authors suggest the Lagrangian relaxation accompanied by the network simplex method, a Lagrangian

heuristic and a modified subgradient method. A computational study based on a case study of a major Taiwan airline is also presented.

[67] deal with the solution of large scale fleet assignment problems. Similar to [18], the problem is formulated as a multi-commodity network flow problem with side constraints, except that, the underlying representation utilizes connection networks. Side constraints are related with marketing, operational and crew restrictions.

[63] study the effect of schedule flexibility on fleet assignment problems. Proposed approach is based on the fact that allowing variability in scheduled departure times of flights improves flight connection opportunities, and hence, a more cost efficient fleet assignment can be generated. The authors present a generalized fleet assignment model that also schedules flight departures simultaneously. Due to the increased complexity with the integration, two algorithmic approaches are proposed to solve the model. Solution approach is tested on a real and large-scale flight data of a major U.S. airline. An important insight provided by the experimentation is the significance of the improvement in fleet assignments with flexible flight schedules.

[13] discuss the shortcomings of earlier fleet assignment models such as static passenger demand assumption. The authors propose a new assignment model, an *Itinerary-Based Fleet Assignment Model*, which is capable of capturing network effects. Moreover, spill and recaputre of passengers are more accurately estimated. Performance of the model is experimented with full-scale networks of a large U.S. airline.

[55] propose an integrated approach for schedule design and fleet assignment problems. The schedule design deals with determining where and where to offer flights so that the profits are maximized. On the other hand, in fleet assignment stage, aircraft types are assigned to flights so that revenues are maximized and operational costs are minimized. The authors propose an integrated approach that simultaneously optimizes these decisions. This study is important in the sense that it presents a framework for expressing demand and supply interactions. Two integrated models and solution procedures are proposed. The first model captures interactions between demand and supply through proposed demand correction methods. However, the approach suffers from intractability issues in large sized problems. Therefore, an approximate schedule design and fleet assignment model is proposed to deal with

large-sized instances. A limited experimentation based on the flight data of a major U.S. airline is presented in the paper.

Similar to [63], [16] solve fleet assignment problems while determining the departure time of the scheduled flights within defined time windows. Main difference of this study from other fleet assignment approaches is that it deals with a periodic scheduling horizon instead of daily operations. Short spacings between consecutive flights having the same origin-destination pair are penalized. The authors propose a non-linear integer multi-commodity network flow model. In order to solve the model, a branch-and-price strategy is applied. [15], on the other hand, deals with weekly fleet assignment problem in particular. An exact mixed-integer linear programming model and a heuristic solution approach based on mathematical programming are presented. The approach is experimented on large-sized instances provided by Air Canada.

### 2.1.3 Aircraft Maintenance Routing

Similar to schedule design approaches, studies on aircraft routing problems are generally integrated with either fleet assignment problem or crew scheduling problem.

[35] develop minimum-cost multi-commodity network flow model with integral constraints for aircraft maintenance routing problem. Aim of the approach is minimizing the number of facilities for a given flight schedule. A two-phase heuristic approach is proposed to solve the problem. In the first phase, aircraft assignments matching flight requirements are generated efficiently by exploiting the Eulerian property of the underlying graph. This procedure is followed by a probabilistically perturbed set covering heuristic.

[24] propose an approach to determine the routes flown by each aircraft in a given fleet for a commercial passenger airline. The authors relax the common practice of fixing the connections during the day and only using overnight connections as options for maintenance routing. Proposed approach enables to consider all connections as options in maintenance routing. Proposed mathematical formulation is an asymmetric traveling salesman problem with side constraints. As the solution procedure, Lagrangian relaxation and subgradient optimization methods are utilized.

[31] deal with the daily aircraft routing and scheduling problem (DARSP). The aim of the problem is to determine daily schedules with the objective of maximizing the anticipated profits derived from the aircraft of a heterogeneous fleet. Two mathematical models are proposed:

- a set partitioning type formulation, and
- a time constrained multi-commodity network flow formulation.

A column generation technique is used to solve the first model, and a Dantzig-Wolfe decomposition is used to solve the linear relaxation of the second one.

[39] cope with aircraft maintenance problem in USAir. Maintenance considerations used in their model include three day maintenance and balance check visit requirements. Simple and polynomial-time algorithms are proposed to determine aircraft routings.

[10] present a single model and solution approach to make fleet assignment and aircraft routing decisions simultaneously. A string-based model and a branch-and-price solution approach using column generation is proposed to solve the integrated problem. A *string* is defined to be a maintenance feasible sequence of connected flights departing and arriving at maintenance stations. An extension of the model is also proposed that can handle complicated constraints such as equal aircraft utilization requirement.

[45] study fleet assignment and aircraft routing problem in the long range planning process, and introduce a new type of constraints related with schedule synchronization. When flights with same origin, destination and time windows are flown on different weekdays, the departure has to be scheduled at the same time every day, for marketing purpose. The constraints forcing such flights to depart at the same time ensure the required schedule synchronization.

[48] present a basis for the development of an on-line decision support system for fleet operations management within airlines. The problem addressed in the study is the integrated fleet assignment and aircraft routing problem. Unlike the majority of studies in the literature, proposed approach aims to face on-line operation conditions.

A Dynamic Programming approach is proposed to cope with the fleet assignment problem, while a heuristic technique is used to solve the embedded aircraft routing problem.

[68] remark the impracticality of existing aircraft maintenance routing approaches. Majority of the approaches generate long-term plans and consider only one or two of the primary maintenance checks that must be performed on a regular, long-term basis. The authors state that these plans are often ignored by the controllers who are required to make quick decisions on maintenance requirements and other irregular events, such as severe weather changes or equipment failures. In order to provide practical solutions, the authors introduce maintenance resource availability constraints. Proposed model is solved using a branch-and-price algorithm.

[43] investigate models and solution approaches for integrated fleet assignment and aircraft routing problems. The aim of the study is to investigate fast optimization-based approximation algorithms for solving the problem. The authors state that a branch-and-price approach would be useful for optimizing the problem, however, would require high CPU times in addition to the significant effort required for implementation. Alternatively, they propose fast network-flow based heuristic approaches based on well-known network flow techniques. Ease of implementation of the proposed heuristic approaches is remarked. Experimentation of the approach with real-data provided by TunusAir shows that proposed heuristics provide fast and near-optimal solutions.

### 2.1.4 Crew Scheduling

This complex stage of schedule planning has also gained great attention of the researchers. The problem is studied continually, and with an increasing interest due to the high costs of flying personnel ([44]). Some of the important studies in the literature are reviewed in this Section.

[44] propose a branch-and-cut approach to optimally solve large set partitioning problems. Proposed branch-and-cut solver generates cutting planes based on the underlying structure of the polytope defined by the convex hull of the feasible integer points.

These cuts are incorporated with a tree-search algorithm. Contractual labor requirements are represented with side constraints. Solution approach is experimented with 68 large-scale real-world crew scheduling problems.

[23] model crew pairing problem as a set partitioning zero-one integer program. Each column in this representation corresponds to a pairing while each row represents a flight. In a feasible solution each flight is covered by exactly one pairing. Costs corresponding to the excess crew cost of the pairing is assigned to each column. The authors point out the impossibility of fully representing the constraint matrix for large schedules, and propose a graph based branching heuristic applied to a restricted set partitioning problem representing a collection of "best" pairings.

[80] develop a new model alternative to the traditional set partitioning representations. Proposed model is based on breaking the decision making process into two stages. In the first stage of the approach, a set of duty periods that cover the scheduled flights is determined. In the succeeding stage, pairings are built based on these duty periods. In order to solve the proposed model, a decomposition approach is proposed.

[14] contribute by introducing a new lower bound for the crew scheduling problem based on a dynamic programming approach. Proposed lower bound is used in a tree search procedure.

[29] integrate aircraft routing and crew scheduling problems. Integration is facilitated by linking constraints imposing minimum connection times for crews that depend on aircraft connections. The authors propose a Benders' decomposition algorithm to handle the linking constraints. The iterative approach solves a master problem for aircraft routing decisions, and a subproblem for crew scheduling decisions.

[50] introduce a new solution approach for crew scheduling problem which enumerates hundreds of millions random pairings. Linear relaxation of the problem is solved first and millions of columns with best reduced cost are selected. These columns are further reduced by a linear programming based heuristic. Integer problem with these columns are solved by a commercial integer programming solver. The authors state that the proposed approach outperforms the approaches in the current practice.

[51] propose a partially integrated approach for schedule design, aircraft routing and

crew scheduling problems. Actually, the approach focuses on crew scheduling, but provides more flexibility while maintaining the feasibility of aircraft routings by introducing *plane count constraints*. Moreover, the approach allows to modify departure times of scheduled flights within given time windows. As a result of their computational study, the authors conclude that provided flexibility while performing crew scheduling results in significantly lower costs.

[83] study the crew scheduling problem of a Taiwan airline. The authors report that work rules in the considered airline are relatively simple compared to the airlines in other countries. This enables the authors to use pure network models in addition to traditional set covering models. Pure network formulations can be solved both efficiently and effectively using real constraints.

[82] cope with cockpit crew scheduling problem. A set partitioning model is used and a column generation algorithm is proposed for efficiently solving the problem. Similar to [83], solution procedure is tested using real data from a Taiwan airline.

[52] report their valuable industrial experience on crew rostering problem. The authors describe real-world constraints and objectives, and reveal the natural complexity of the practical problem. Methodologies used in the Carmen Crew Rostering system, a commercial crew rostering system used in several major European airlines, are also presented.

[41] consider crew scheduling problem where crew members are stationed unevenly among home bases. The authors introduce the basic idea of a partially integrated approach for solving the problem and remark the advantages of integration over the traditional sequential approach. Proposed approach can handle dynamic changes in the availability of the crew members during the planning period. These changes may occur due to pre-scheduled activities, such as vacancy, or off-duty days.

## 2.2 Recovery Problems

Majority of studies in airline recovery literature focuses on aircraft recovery problems since aircraft are considered to be the scarcest resource of airlines. The next most studied recovery stage in the literature is crew recovery, while there are fewer approaches that focus on passenger recovery. We group the studies on airline recovery problem in four subsections:

1. Aircraft recovery
2. Crew recovery
3. Passenger recovery
4. Partially and fully integrated recovery

Studies explained in aircraft recovery section, Section 2.2.1, are generally dedicated approaches. In some of these studies, crew and passenger related costs are involved, however, crew and passenger recovery actions are not integrated. Some of the studies summarized in crew recovery section, Section 2.2.2, are dedicated while some of them are integrated with aircraft recovery. Dedicated crew recovery approaches try to repair crew schedules assuming that aircraft recovery decisions are already made. Unlike crew recovery, we have not found any dedicated passenger recovery approaches. Therefore, studies summarized in passenger recovery section, Section 2.2.3, are integrated with either aircraft recovery or with aircraft and crew recovery. Finally, we list partially and fully integrated recovery approaches in Section 2.2.4. For a recent review on airline recovery problems, we refer to [26].

### 2.2.1 Aircraft Recovery

Aircraft routing recovery is the recovery process that focuses on the aircraft resource. Considering the number of aircraft in the fleets of the airline and maintenance constraints, aircraft routing recovery aims to return to the original schedule with respect to a preferred objective. The objective is often minimizing the operating costs, maximizing the profit or minimizing the time required to return to the original schedule. Common recovery actions used for recovering aircraft schedules are aircraft swapping, flight cancellation, standby aircraft, and departure time holding. In addition to optimization models, interactive solution approaches that benefit from the experience and intuition of airline controllers provide an important contribution.

The first study on aircraft recovery problem is proposed by [75] in 1984. The authors deal with the disruption scenario in which some aircraft in the fleet become unavailable for a period of time. Recovery actions utilized in the proposed approach are rerouting aircraft and departure time holding. The aim of the approach is to operate the scheduled flights with the reduced number of available aircraft. The problem is represented by a type of a connection network where scheduled flight legs are represented by nodes. Moreover, available aircraft are represented by a second type of nodes, as well. Based on this representation, the authors develop a nonlinear integer model with the objective function of total passenger delay. In order to determine aircraft routings and departure time decisions resulting in the minimum total passenger delay, a Branch and Bound procedure is proposed.

[74] focuses on airport closures in some regions during winter months due to meteorological conditions. Airport closure is one of the most major disruption types resulting in a great number of flight cancellations, and entire airline network is affected from cancellations. The author studies the reliability of aircraft schedules related with the meteorological conditions. One of the main contributions of this study is the proposed indicator which quantifies the adaptability of aircraft routings to meteorological conditions. A heuristic approach which tries to minimize the number of aircraft required to operate the scheduled flight legs with a given traffic volume is proposed. When alternative solutions are found, the heuristic selects the recovery decisions that result in the minimum number of passengers whose flights are cancelled.

Due to many reasons such as mechanical problems, one or more aircraft may be taken out of operation during the execution of flight schedules. [76] deal with such disruptions where the airline has to operate the flights with an aircraft shortage. Setting the recovery horizon as the day of operations, the aim of the proposed approach is to create a new daily aircraft schedule which minimizes total number of cancelled flights in these situations. A secondary objective is also used to select the solution resulting in minimum total passenger delay among those with the same number of cancelled flights. The problem is broken into subproblems for each fleet type. For the solution methodology, the authors propose a greedy heuristic algorithm for solving a lexicographic optimization problem. Efficiency of the proposed algorithm is tested on a small-sized scenarios. In the proposed approach, crew related restrictions are not taken into account.

[77] enhance the solution methodology proposed in [76] by considering crew-related restrictions. The problem is again solved for each aircraft type separately. Similarly the primary of the proposed approach is maximizing the number of flown flight legs in cases of aircraft shortages, while the secondary objective tries to minimize total arrival delay experienced by the passengers. The authors propose a heuristic approach that uses first in, first out (FIFO) principle and a sequential approach based on dynamic programming which tries to facilitate the tasks carried out by the dispatchers. A software package based on the proposed heuristic model is developed. An interesting and important contribution of the proposed approach is the interaction with the dispatchers. Dispatchers are assigned an active role using the software on confirming the final solution. Moreover, they may change the fleet type of flights and resolve the problem. In addition to retiming decisions of scheduled flights and new aircraft routings, the output also includes crew rotations.

[47] point out the importance of aircraft recovery problem and present an overview of a decision support system with the attempt of conceptualizing the problem and form a basis. In particular, the authors deal with aircraft shortages. In order to recover the schedules of the aircraft, departure time holding, flight cancellation, aircraft swapping and using standby aircraft actions are utilized. Proposed solution methodology is based on network flow theory. The problem is represented by a timeline network. The authors develop two minimum cost network flow models. Solution of the first model, called the delay model, determines the set of flights to be delayed that can absorb the shortages. The model is a pure minimum cost network with arcs bounded by a flow of unity, and the proposed approach delays flights until the shortage is fixed. The second model, called the cancellation model, by determining the optimal set of flights to be cancelled with respect to the same goal. The model can handle more than one cancellations, and again utilizes aircraft swapping and using standby aircraft actions. Both models are solved using Busacker-Gowen's dual algorithm. One significant drawback of the proposed approach is the separation of departure time holding and flight cancellation decisions into different models. The tradeoff between cancelling and delaying a flight can be evaluated to some extent. However, since the models are pure delay and pure cancellation models, solutions with both cancelled and delayed

flights are not evaluated. On the other hand, the main aim of the proposed models is to provide decision support for the dispatchers in airline operations control centers while finding good solutions in real-time. The authors present a computational study based on a network with three airports and a considerable air traffic. Some test scenarios are based on the real data of United Airlines.

[60] presents a novel approach on the integration of computer science and operations research techniques in airline industry. The study focuses on development of a decision support system for the dispatchers in airline operations control centers. Described system is designed and developed on distributed desktop UNIX workstations, networked through ethernet TCP/IP communications, with an X Windows Motif graphical user interface. Features of the application can be summarized as:

- real time flight following,
- aircraft routing,
- maintenance planning,
- crew management,
- gate assignment,
- flight planning and aircraft recovery against disruptions.

The interactive, graphical user interface provides several representations to aid dispatchers follow and edit the airline operations. Some of these representations are routing chart, station-to-station activity chart, and gate activity chart. Based on the integration of airline operations, the author proposes a rule system which tracks operational constraints. Whenever, dispatchers decide to deviate from the original schedules, the rule system quickly checks whether these constraints are violated or not. The rules are of the *If-Then* form and are classified under: maintenance, crew, operational, etc. categories. Parameters of these rules can be updated by the dispatchers, such as number of aircraft overnighting at a particular station. As a second dimension, the rules are classified according to their criticalities. Violations of level I rules issue a visual alarm to the dispatchers, while level III violations only generate a violation report. Another feature of the proposed application is its ability to carry out *What-If* analysis by simulating disruptions, such as closure of an airport. Finally, the application is equipped with aircraft recovery mechanisms. In cases of cancellations or significant flight delays, aircraft rescheduling alternatives minimizing the effects of the disruption are generated. The problem is represented by a network flow model, and the solution procedure is based on an Out-Of-Kilter network flow algorithm.

[84] propose a decision support framework to handle schedule perturbations in airline industry. The authors assume a single fleet type and focus on disruptions occurred due to an aircraft breakdown. The problem is represented on a time-space network. Based on this representation, the authors propose pure network flow models and network flow models with side constraints. The prior ones are solved using the network simplex method, while for the latter one the authors apply Lagrangian relaxation with subgradient method. Computational study presented in the study includes real life problems of a major Taiwan air carrier.

[73] considers the problem where the schedule is given and one of the fleet assignments need to be changed. In other words, it is aimed to swap the fleet type of a specific flight satisfying all restrictions such as flow balance, flight coverage and aircraft count, while trying to minimize the resulting cost. The procedure is restricted to find swaps between two aircraft types at a time. The reason why swaps across three or more aircraft types are not allowed is the great burden on computation times. The author aims to find a solution within a minute or two. The favorable swaps are the ones that involve few or none overnight equipment type changes due to maintenance. Two algorithms are proposed in the study to find swap opportunities. The first algorithm runs in linear-time and finds same-day swaps (with no overnight equipment type changes) if such a swap exists. For the cases where no such opportunity exists, the author proposes a general algorithm that restricts the search for k-overnight equipment changes which is polynomial for fixed $k$.

[6] present a greedy randomized adaptive search procedure (GRASP) to carry out aircraft recovery. Flight cancellation, departure time holding and aircraft swapping are the set of recovery actions integrated in the solution procedure. The GRASP described in the study is adapted for use as a randomized neighborhood search technique. Neighbor generation operations are performed on pairs of aircraft routes.

These routes are generated from an incumbent aircraft routing. Feasibility considerations in neighbor generation process include flight coverage and aircraft balance at airports. Alternative routes are evaluated by the incurred cancellation and delay costs. Proposed approach keeps a limited set of candidate solutions corresponding to the best local routings. The authors present a computational study based on the flight data of Continental Airlines and report solutions within at most 5% gap with the optimal solution is obtained in about 70% of all test instances.

[21] propose a real-time decision support tool for adapting flight schedule and fleet assignment in cases of unforeseen perturbations in the planned schedule. The authors aim to simultaneously evaluate delay and cancellation options. They present a quadratic 0-1 programming model for the integrated problem which tries to maximize the profit while taking into account delay and cancellation cost penalties. In addition to their base model, the authors also consider the issues of ferrying aircrafts and multiple aircraft type swapping capabilities as an extension. In the second part of the paper, [22] present an effective algorithm for solving the problem in real time.

Ground Delay Program (GDP) is one of the several programs that the FAA is administering for efficient and equitable use of scarce airspace and airport capacity. In poor weather conditions, the FAA may decide that the number of planned arrivals at an airport will exceed the airport's capacity. In such cases, GDP is initiated, and in a GDP usually the arrival times of these flights are delayed. [58] address such disruptions. The performance measure is the percentage of flights that are delayed more than 15 minutes. The problem is modeled as an integer problem. Valid inequalities and variable reduction methods are used to solve the problem.

[78] deal with disruptions due to aircraft shortages. Different from earlier studies, main objective of the solution approach is to minimize the deviations from the original aircraft schedules. In other words, the emphasis is placed on the *stability* of the scheduled operations. Proposed solution methodology is a network model with side constraints that utilizes departure time holding and flight cancellation actions. The authors present a computational study based on the flight data of Continental Airlines and report that generally optimal or near-optimal solutions are obtained. The solutions are obtained from the LP relaxation of the network model, and a rounding

heuristic is proposed to deal with non-integral solutions. An important feature of the proposed approach is the interaction with the decision makers, i.e. the dispatchers. The model is flexible in the sense that the dispatchers can reflect their preferences in the objective function.

[8] present the time-band optimization model for reconstructing aircraft routings in the cases of groundings and delays experienced over the course of the day. The objective of the problem is to minimize the costs measured by flight delays and cancellations. While rerouting, the following constraints are considered:

- every flight in each routing must depart from the airport where the immediately preceding flight arrived,
- a minimum turn time must be ensured between each flight arrival and subsequent departure,
- the recovery period extends to the end of the current day,
- the schedule should be resumed the next day,
- airport departure curfew restrictions should be observed,
- no scheduled maintenance should be violated.

The authors construct time-band model transforming the problem into a time-based network whose time horizon is discretized. The resulting formulation is an integral minimum cost network flow problem with side constraints.

[65] propose *delay threshold policy*. Upon the realization of a disruption, propagated delay is estimated. The authors state that if the total delay is not over an acceptable value, *push back policy* may be implied. Otherwise, aircrafts need to be rerouted. In the procedure presented, aircraft recovery problem is solved for each fleet separately. Authors point out the difficulty of determining delay and cancellation costs. It is stated that anecdotal evidence suggests minimizing cancellations, delay minutes, and total delays sequentially. They develop a heuristic that selects only a subset of aircrafts to be rerouted in order to reduce the size of the problem.

[79] consider a specific case of disruptions, called *hub closure*. In the time interval that the hub is being closed, no transient activity is permitted. The inputs of the problem are the positions of aircrafts at the time of closure, the original flight schedule, the time of hub closure and reopening, and a time set for recovery. The objective is to find the best assignment of aircrafts to flights such that at the end of the recovery period, all flights can be flown as originally scheduled. In order to find real-time solutions, estimates are made. The authors suggest a rolling horizon approach in which the models can be rerun when more accurate information becomes available. The problem is modeled as a multicommodity network model. Then, a bundle algorithm is presented to provide feasible near-optimal solutions much more quickly than a mixed-integer program. The performance of the bundle algorithm is compared with the MIP solved with CPLEX and the authors report the superiority of solutions of bundle algorithm with respect to finding much quicker solutions and providing multiple high-quality solutions instead of a single best solution.

[5] consider four actions for recovery, delaying, swapping, cancelling and ferrying. The problem is modeled as a mixed integer multi-commodity flow model with side constraints, where aircrafts are commodities. Using Dantzig-Wolfe decomposition, the model is reformulated as a set packing model.

[56] deal with the dedicated aircraft recovery problem. The authors propose several heuristic approaches based on a network representation of the problem with the aim of handling problems of a realistic size (about 100 aircraft and 500 flights) in real time (no more than three minutes). The approach tries to balance the tradeoff between delays, cancellations and swaps. They test their approach with disruptions from British Airways, and revised flight schedules with good quality are generated in less than 10 seconds on the average.

[34] define the new concept called recovery network as a set of nodes and arcs, such that each possible recovery scheme of the unit (aircraft, crew member or passenger) corresponds to a path. In addition, each unit-specific constraint is modeled as a resource and an associated resource limit. For example, an aircraft has limits on the consecutive flown hours, crew members have limits on the duration of duty and passengers have limits on the delay of their itinerary. Resources are either consumed (by

flights) or renewed (maintenance). Aircraft recovery program (ARP) aims to assign a recovery scheme to each aircraft such that original schedule is maintained at a determined point in time. While determining the recovery plan, maximal flight hours, maximal number of take-offs and landings and the maximal absolute time elapsed between two maintenances are obeyed. Output of ARP, which is a feasible recovery plan, is used as input for passenger recovery program (PRP). The authors assume that either the passenger should be brought to their final destination within a maximum delay limit which depends on the original itinerary length or the itinerary should be cancelled. This time, each passenger itinerary (or groups of passengers with the same itinerary) is considered as a resource. Capacities of the aircrafts are taken into account while creating the recovery plan. Crew recovery problem is also solved similarly, using the output of ARP as input.

In this thesis study, we try to recover from disruptions at the end of the operating day in order to guarantee that each aircraft reaches its scheduled maintenance point. In Chapter 3, we integrate cruise speed control action with departure time holding and aircraft swapping actions in addition to passenger recovery actions. In Chapter 4 and Chapter 5, we propose a new network representation that generate all possible rerouting opportunities for each of the entity types.

### 2.2.2 Crew Recovery

Aircraft recovery takes the first place in sequential recovery approaches in the literature and in practice due to the scarcity of aircraft. Crew recovery plans are generally generated based on the recovered aircraft routings. Majority of studies dealing with crew recovery problem is not integrated. On the other hand, a minor part of these studies try to integrate aircraft recovery with crew recovery.

Because of the complexity of crew schedules, restrictive crew legalities and the size and scope of hub-and-spoke networks, [81] state that, crew management during irregular operations is usually the bottleneck of the recovery process. The approach assumes that the flight schedule has been fixed and thus is given. The authors focus on crew scheduling problem only. The objective of the problem is defined as to return to the original schedule as soon as possible, preferably in a cost-effective way.

Crew pairing repair problem is modeled with a space-time network within a given time window. The start of the time window is considered as the current time while the end of the time window is the proposed time at which the original schedule will be restored. Since pilots are qualified for only one type of aircraft and the crew pairings are built by fleet type, the problem is separated to aircraft types and modeled as an integer multi-commodity network flow problem. For obtaining fast solutions, a heuristic search algorithm is presented.

[71] provide the first attempt to solve operational airline crew scheduling problem. The problem is to modify personalized scheduled monthly assignments of crew members during day-to-day operations. Proposed procedure tries to cover all flights (no cancellation) with available crew members while trying to minimize the crew cost and disturbances of crew members. In order to find an integrated solution, the authors model and solve crew pairing problem and crew assignment problem simultaneously. Crew recovery problem is formulated as an integer nonlinear multi-commodity flow problem. Dantzig-Wolfe decomposition principle is embedded in a branch-and-bound search tree in order to solve the integer nonlinear formulation. The decomposition is equivalent to the column generation scheme with a master problem defined as a Set Partitioning type model and a specific pairing generator (subproblem) for each crew candidate. In order to define the pairing costs, three terms are used:

- real cost of the pairing: credited time cost, transportation cost between the airport and the hotel, deadhead cost, per diem and hotel cost, etc.,
- penalties: these costs are used to penalize the violation of global constraints,
- bonuses: these costs are used when a certain activity is suggested to an employee.

[54] point out the complexity of airline recovery problems due to size of the entire schedule and real time nature of the problem, and describe the necessity to reduce the complexity and size of the problem instead of applying a full-scale optimization method. Based on the fact that the original schedules are optimal prior to the experienced disruption, the authors propose preprocessing techniques to extract a subset of crew schedules to be rescheduled. In other words, the problem size is reduced by

reducing the number of schedules that may be altered and an optimization model is solved on the reduced feasible region. Proposed solution approach is based on an integer programming formulation. Mathematical formulation together with the pre-processing methods provide recovery plans for crew members in almost real time. Moreover, the schedule selection procedure in preprocessing stage guarantees that the original schedule will be disturbed as little as possible. In other words, the disturbance on the stability of airline operations is limited.

[69] contribute to the literature by introducing the problem of crew scheduling which performs well in operations with disruptions taking uncertainties into consideration. An easily implemented procedure for finding approximate solutions for the problem of minimizing expected crew costs. Moreover, a lower bound on the expected cost of any crew schedule is provided. In an uncertain environment, a measure for evaluating the performance of crew schedules is developed and it is shown that this measure performs better than a deterministic model. SimAir, a Monte Carlo simulation of airline operations with disruptions, is used to evaluate crew schedule's performance. Finally, it is aimed to provide insight into what type of pairings perform better in cases of disruptions. The authors focus on push-back recovery policy because of its simple structure. Two methods for finding crew schedules that may perform well in operations are proposed. These methods try to reflect the pairing costs in operations with disruptions more accurately than deterministic models. Once these costs are obtained, a set partitioning model is solved using an algorithm developed by [50].

[61] focus on crew recovery by integrating crew pairing and crew assignment problems during the recovery process. The authors deal with critical crew recovery problems arising on the day of operations. Given a disruption, infeasibilities may occur in crew schedules, and the authors list the following examples for such cases:

- arrival delay of a flight may result in an insufficient connection time among two consecutive flights of a crew member;
- due to a flight cancellation, the flight schedule of an operating crew member can be operationally infeasible;
- a crew calling sick disturbs the scheduled flights assigned to this crew member;

- decisions related to aircraft recovery may cause infeasibilities in crew schedules (e.g. illegal aircraft type - crew member match).

In these cases, the proposed approach tries to repair all illegal individual roster while trying to cover all scheduled flights. The authors propose to carry out pairing construction and pairing assignment in a single step. Provided solution methodologies are based on simple tree search and more sophisticated column generation and shortest-path algorithms.

[1] develop an integrated Decision Support Tool for Airlines schedule Recovery during irregular operations (DSTAR). The tool is designed for the operators in AOCCs and is capable of detecting current and future flight delays and aims to generate proactive integrated recovery plan to avoid these delays. Proposed framework integrates a schedule simulation model and a resource assignment optimization model. The schedule simulation model predicts the list of disrupted flights, while the optimization model tries to find the optimal plan of crew and aircraft swapping, reserve utilization and flight delays to recover the predicted disruptions. Besides its integrated solution approach, the study contributes to the literature by creating proactive actions against anticipated resource problems. It is aimed to obtain real-time solutions. Schedule simulation model and optimization solver are integrated in a rolling horizon framework. First schedule simulation model is activated to simulate all flights and their resources. The model projects all potential downline resource violations within a given horizon due to the introduced delays and cancellations. Resource violations may include misconnect, rest and duty limit violations. If a violation is projected, all the flights that are in the routing of that flight are considered as disrupted flights. Once the set of disrupted flights is projected, the first recovery stage in the horizon consisting of the earliest set of resource-independent flights is determined. To be solved at each stage, a mathematical formulation of the recovery problem as a mixed integer program (MIP) is developed. The objective of the MIP is to minimize the total cost associated with recovering all flights in the stage under consideration. Total cost consists of resources assignment cost, total delay cost and cancellation cost. In order to illustrate the application of the tool, a detailed example to recover the schedule of a major U.S. airline is provided.

[70] propose an approach that integrates certain aspects of:

- schedule design,
- fleet assignment,
- aircraft routing, and
- crew scheduling problems.

Proposed solution methodology is an alternative mixed-integer programming model. In order to deal with the complexity of the integrated problem, the authors propose a reformulation-linearization technique. Moreover, a Benders' decomposition-based solution approach is proposed to deal with large-sized problems. A computational study using the real flight data of United Airlines is presented and potential profitability that can be achieved by applying the proposed approach is reported. One of the important contributions of the study is the proposed alternative problem representation. The authors utilize a flight network representation alternative to the traditional time-space networks.

In Chapter 4 and Chapter 5, present methods to generate all possible rerouting actions for each crew member while satisfying their restrictions. Furthermore, speeding up some flights enables new crew swapping and rerouting opportunities, and hence, enlarges the solution space.

### 2.2.3 Passenger Recovery

There are fewer studies on passenger recovery in the literature than aircraft and crew recovery. In practice, passenger recovery is observed as the last stage of the sequential recovery approach. In some methodologies passenger related costs are considered in the prior stages occurring according to the recovery decisions. These methods try to evaluate the effect of aircraft and crew recovery decisions on passenger convenience which has a great impact on the profitability of airlines. However, passenger recovery decisions such as reallocating or spilling are not integrated. Therefore, without detailed analysis of passenger itineraries and recovery actions, it is not possible to make optimal recovery decisions. In the literature, studies that utilize passenger recovery actions are not dedicated approaches. Due to the high dependency of passenger schedules on aircraft and crew schedules, passenger recovery is either integrated with aircraft recovery or with both aircraft and crew recovery.

[64] present a stochastic model, which is a discrete event semi-Markov process, described in terms of states and transitions that can either be random or deterministic. The input of the model is an original schedule which includes:

- a set of crews, their pairings and their bidlines,
- a set of aircrafts and their routings,
- a set of itineraries and their passengers, and
- a set of reserve crews that are not assigned to pairings but can be used in operations.

The state of the stochastic model includes:

- the deviation of the current schedule from the original one,
- historical information required to calculate performance measures and to determine whether the current schedule will violate planning rules, and
- conditions that are not in control of the airline, such as weather.

Transitions are observed as a result of an event (e.g. departure events, arrival events, weather events, congestion events) or a set of decisions. Proposed model uses a semi-Markov process for the state of the weather. The model uses aggregate distributions for the *ground time*, the time duration from the moment the plane and crew are ready until the departure of a leg. As recovery decisions, the authors use delaying legs, cancelling legs, deadheading crews, ferrying aircrafts, swapping aircrafts, rerouting crews on new reconstructed pairings and rerouting passengers. Crew costs, cancellations and on-time percentage, passenger misconnection are used as performance measures. For the implementation of the stochastic model, the authors present SimAir and give computational results.

[72] deal with Day of Operations Scheduling (DAYOPS) problem that involves determining real-time changes to planned airline schedules in cases of perturbations with the objective of minimizing passenger inconvenience and airline operating costs. The authors contribute by modeling and solving DAYOPS optimally in real-time provided that the disruptions are minor. The actions that are used against these minor disruptions is only modifications in arrival and departure times. In addition to activity start times, activity durations (flight times) are also considered as variables. The problem is modeled with the objective which consisting of cost terms defined as linear functions of arrival times and flight times. The authors show that dual reformulation of the problem is a network problem that can be solved in time linear in problem size; and hence, can create real-time solutions. The idea that the authors suggest is that, in case of disruptions, the dispatchers can try to restore the schedule simply by delaying some flights using the approach presented in their study, and expensive changes, such as rerouting aircrafts, swapping, can be considered if a satisfying solution is not provided.

[20] point that passenger disruptions rarely drive operational decision-making while AOCCs create recovery plans. Studies show, however, that arriving on time is the service characteristic most valued by passengers. The authors propose airline recovery models that decides flight departure times and cancellations with the objective of minimizing operating costs, like conventional models, and are extended to include passenger delay and disruption costs. *Disrupted Passenger Metric* model is proposed. The model includes a binary variable which determines whether a planned itinerary is disrupted or not, and passenger disruption cost is approximated by the product of number of disrupted passengers and an estimated cost per disrupted passenger defined for each itinerary. The authors make the assumption that passenger itineraries include at most two legs and state that this assumption could be relaxed. This assumption is true for a major fraction of flights; however, there also exists three leg flights: spoke-hub leg, hub-hub leg and finally hub-spoke leg. Using an Airline Operations Control simulator developed by the authors, experiments are provided for several days of operations of a major U.S. airline.

[53] propose two new approaches to minimize passenger disruptions and achieve robust airline schedule plans. The first approach involves aircraft routing and the second

one involves retiming flight departure times. In most optimization models for the aircraft maintenance routing problem, the objective is to maximize *through revenue*, the potential revenue obtained by offering passengers the opportunity to stay on the same aircraft when making a connection at an airport. The authors state that this additional revenue is very difficult to estimate in practice and moreover, its financial impact is relatively small. Therefore, in their study, aircraft maintenance routing problem is considered as a feasibility problem with the aim of achieving robustness with minimal cost implications. *Propagated delay* is defined to be the delay that occurs when the aircraft to be used for a flight is delayed on its prior flight. As a result, this delay is a function of the aircraft's routing. In the proposed robust aircraft maintenance routing (RAMR) model, the authors try to minimize the expected total propagated delay. RAMR is a stochastic discrete optimization problem without random variables in the constraints. Since the expectations in the objective function can be computed offline, RAMR is actually a deterministic mixed-integer linear program with a large number of binary variables. For solving realistic problems, a branch-and-price approach is used.

In the second part of their study, [53] consider passengers who miss their flights due to insufficient connection time. Aim of this approach is to minimize the number of passenger misconnections by retiming the departure times of flight legs within a small time window. In case of a disruption, slacks may absorb the delay and prevent passenger misconnections; however, adding too much slack would reduce the productivity of the fleet. Therefore, it is important to make the right decision on where to add the slack so as to maximize the benefit to passengers without requiring additional aircraft to fly the schedule (no rerouting). The method used to add slacks is moving departure times of flights while maintaining aircraft productivity. In practice, departure times of flights can be altered in a small time window that starts several weeks before the flight and ends in the day of departure. For retiming departure times, the authors propose the connection-based flight schedule retiming (CFSR) model. The objective of the model is to minimize the expected total number of disrupted passengers. Similar to RAMR, CFSR is also a deterministic mixed-integer program with a large number of binary variables. Again a branch-and-price algorithm is developed to solve practical-size problems. The method is experimented using data from a major U.S. airline and the authors report that number of passenger misconnections can substantially be reduced using the proposed approach.

Like [20], [46] also point out the lack of studies on passenger recovery in aircraft recovery models in the literature. The authors present an assignment model for airline schedule recovery which recovers both aircraft and disrupted passengers simultaneously. Schedule recovery actions that may be used by the model are determined as:

- calling up reserve aircraft,
- swapping aircrafts,
- over-flying (to fly to another scheduled destination),
- ferrying,
- delaying, and
- cancellation.

Proposed mathematical formulation of the problem has the objective of minimizing the sum of aircraft assignment cost, total delay cost, cancellation cost, and disrupted passenger cost. Disrupted passenger costs include the cost of rerouting them to the earliest available itinerary or transporting them to the destination by another way. In the proposed approach, the recovery horizon is broken into recovery stages in order to reduce the number of disrupted flights and aircrafts that will be included in the model.

[59] integrate recovery decisions for aircraft and passengers with cruise speed decisions. Proposed approach is based on a time-space network representation similar to the one presented in [20]. Similarly, flight copy generation method is applied to represent different departure time decisions for each scheduled flight. The formulations of [20] is enhanced by the integration of cruise speed control action. Different cruise speed options are represented by introducing a second set of flight copies for each flight copy having different timing decisions. Flight copies in this second dimension have different slopes, each corresponding to a different arrival time while these arcs originate from the same departure time. The authors propose two mathematical models to find the optimal aircraft and passenger recovery decisions. In the

first mathematical model, passenger rerouting actions are evaluated integrated with aircraft recovery actions. The authors report the intractability of this formulation due to the large size of airline networks. Therefore, an alternative formulation is proposed which uses an approximation for evaluating passenger delays.

[62] integrate aircraft, crew and passenger recovery problems. Formulations are based on a single-day recovery horizon. The authors propose separate mixed-integer programming models for each of the:

- schedule recovery,
- aircraft recovery,
- crew recovery, and
- passenger recovery problems.

These subproblems are based on a flight string representation. The coordination between subproblems are constructed by a Benders' decomposition scheme together with the column generation approach. Time limit for the integrated recovery is set to 30 minutes in the experimentations. The authors also propose a sequential approach, similar to the sequential recovery approach in practice, to handle disruption scenarios on large airline networks.

In Chapter 3, we integrate beneficial passenger recovery actions with aircraft recovery decisions to reduce passenger delay and spill costs. With problem representations presented in Chapter 4 and Chapter 4, we propose to generate all possible passenger reallocation actions in order to optimize passenger recovery decisions. Furthermore, we observe a significant reduction in total passenger delay and in the number of passenger misconnections with cruise speed control action.

### 2.2.4 Partially and Fully Integrated Recovery

There is an increasing effort in the literature for integrating several or all aspects of aircraft, crew and passenger recovery. We have discussed integrated as well as dedicated approaches in the previous subsections which are placed according to the order of the subproblems in the sequential recovery approach and the main focus of studies. In this subsection, we list some important integrated recovery approaches. [1] and [70] integrate aircraft and crew recovery problems. [72] and [46] deal with aircraft recovery with considering passenger related costs, while [53] integrates aircraft and passenger recovery processes. [20], [59] and [62] consider recovery actions related with aircraft, crew and passengers simultaneously.

These studies integrate recovery decisions to different extents in terms of the considered entity types and utilized recovery actions. In Chapter 3 we propose an approach that integrates aircraft and passenger recovery. In the network-based formulation that we propose in Chapter 4, we try to achieve a full integration by:

- considering disruption and recovery costs related with aircraft, crew and passenger in the objective function;
- allowing to model entity type-dependent operational restrictions; and
- including all possible recovery actions related with each of these entity types in the solution space.

### 2.3 Problem Representations

Majority of problem representations used in airline operations studies in the literature can be classified into three groups:

1. time-space network (TSN) representation,
2. flight string (FS) representation.
3. connection network (CN) representation, and

Classification is made on the main characteristics of the representations. However, representations in the same group may differ slightly from each other. In this section, we discuss some important studies in the literature that utilize these representations.

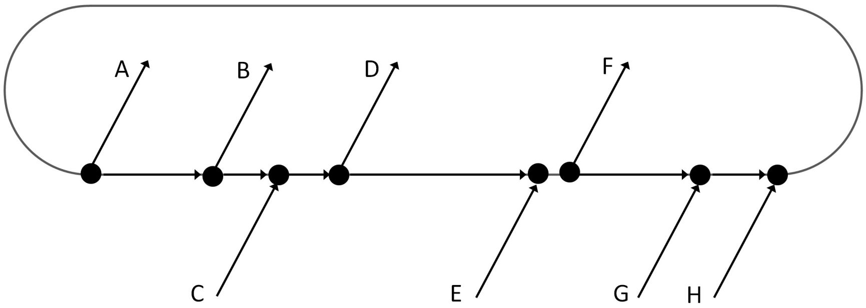


Figure 2.1: A city-fleet time line presented in [42].

### 2.3.1 Time-Space Network Representation

An early example of TSN representation is proposed by [42] for dealing with the fleet assignment problem. The authors classify the proposed representation as *time-expanded multi-commodity network*. Aircraft balance is satisfied by modeling activities at each airport with a time line for each fleet. An example time line corresponding to a city and a fleet is illustrated in Figure 2.1. Nodes on the time line correspond to arrivals and departures of the corresponding fleet at the corresponding airport. Time of arrival nodes are modified by adding refueling and baggage handling time to the arrival time. Therefore, time of an arrival node corresponds to the time when the aircraft is ready to takeoff. Departure and arrival nodes of a flight are connected by a decision variable which represents the assignment of the corresponding fleet to this flight.

[79] use TSN representations to deal with hub closures. Problem is represented by a collection of TSNs, each TSN corresponding to an equipment type (fleet). A single TSN is illustrated in Figure 2.2. Time is represented by the vertical axis, while horizontal axis represents airports. Authors deal with aircraft recovery, and hence, flows on the network are aircraft. Flow is from top to down. Sloping arcs represent flights from one airport to another. Vertical, or ground, arcs (arrows are omitted to avoid clutter) represent the aircraft on the ground that are waiting for a flight. Authors propose to use binary variables for flight arcs (as only a single fleet type will be assigned to each flight), while integer variables are used to represent ground arcs (since there may be many aircraft at an airport at a particular time).

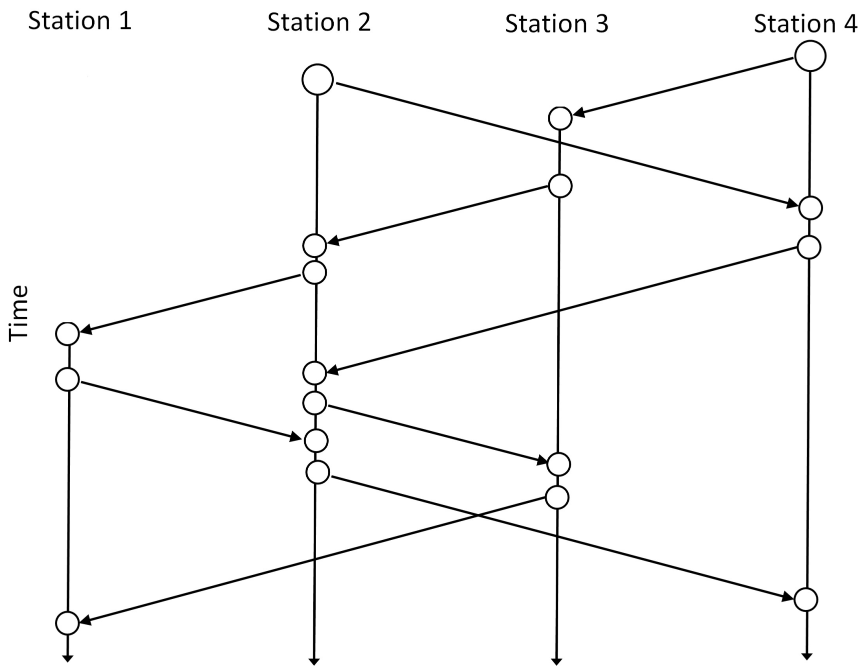


Figure 2.2: TSN representation presented in [79].

[20] integrates aircraft, crew and passenger recovery by utilizing departure time holding and flight cancellation actions. Underlying representation of the proposed formulation is a TSN representation similar to [79]. In addition to flight arcs corresponding to scheduled flights, the authors propose to generate copy arcs for utilizing departure time holding action. For each possible departure time of a flight, a copy arc is included to represent the corresponding departure time decision. In this manner, time is discretized. Flight copy generation process is illustrated in Figure 2.3. In this figure, three copies of a DCA-ORD flight with a scheduled departure time of 7:50 are presented. Note that space is represented by the vertical axis and time is represented by the horizontal axis. [84], [85], [78] and [5] are other important studies that generate flight copies for every m minutes for each flight. [20] show that many of the flight copies are dominated, and hence, can be eliminated from the solution space.

[59] enhance the approach proposed by [20] by integrating cruise speed control with common recovery actions. Flight copies are again generated for each departure time decision of each flight. Additionally, the authors propose a second dimension of flight copy generation process to represent cruise speed decisions. For each flight copy corresponding to a departure time of a flight, flight copies corresponding to

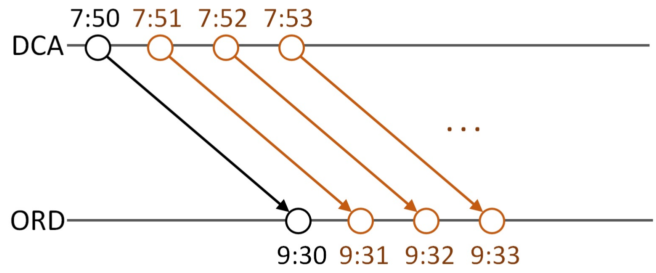


Figure 2.3: Flight copy generation process for departure time holding action presented in [20].

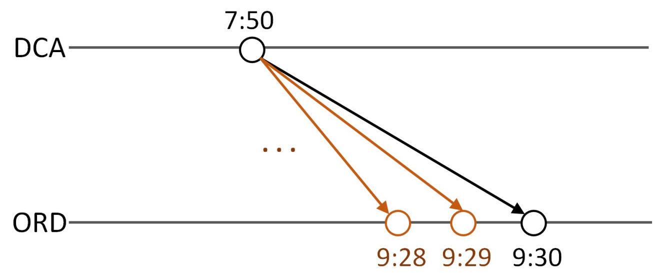


Figure 2.4: Flight copy generation process for cruise speed control action presented in [59].

different cruise speeds are generated. In addition to departure time options, cruise speed options are discretized in the same manner. This second type of flight copies have different slopes than those of the original flights. Two flight copies of a DCA-ORD flight with a scheduled arrival time of 9:30 are represented in Figure 2.4. Note that these copies start from the same departure node, however, they arrive earlier due to the increased cruise speed.

As addressed by [20], the main drawback of TSN representation in airline recovery problems is the huge size of the generated networks. Problem size has a linear relationship with the number of flights multiplied by the number of departure time options. If time is discretized for each minute, a precise representation would be achieved. In this case, if two hours of departure time delays are allowed by the airline policy and time slot restrictions, problem size will be linear with 120 times the number of flights. With cruise speed integration, this coefficient will be multiplied with the number of cruise speed options considered for each flight. Therefore, some authors limit the departure delays by 15 minutes, discretize time for each five minutes and evaluate only several cruise speed options to be able to provide real time solutions. However, this practice restricts the solution space and results in suboptimal solutions.

In our alternative problem representation explained in Chapter 4, flights are represented by nodes (activity-on-node). Therefore, we represent departure and arrival time of flights with continuous variables. This has two advantages. Firstly, problem size is linear with the number of scheduled flights. Secondly, we do not need to discretize time and restrict the solution space. Arcs connecting flight nodes represent feasible flight connections for each of the entity types: aircraft, crew and passenger. Number of arcs, on the other hand, is directly related with the number of rerouting opportunities. Therefore, problem is represented with its natural limits. Moreover, variable operation times, i.e. cruise speed control, is easily integrated with the proposed networks. Cruise time decisions are also represented by continuous variables, and hence, cruise speed options are not discretized.

### 2.3.2 Flight String Representation

[10] propose flight string models to solve the integrated fleet assignment and aircraft routing problem. The authors define a *flight string* as a sequence of connected flights with the following characteristics:

- origin of the first flight and the destination of the last flight in the string correspond to a common maintenance station,
- origin of a flight in the string is the same airport with the destination of its preceding flight, and
- the sequence of flights are maintenance feasible, i.e. it satisfies all FAA and carrier-specified maintenance requirements.

Then, an *augmented flight string* is defined to be a FS with the minimum time necessary to perform maintenance attached to the end of the last flight in the string. The objective of the proposed formulations is to select the set of augmented FSs so that:

- each flight is assigned to exactly one fleet;

- for each flight, assigned flights are partitioned into a set of rotations;
- each aircraft in a fleet is assigned to at most one rotation;
- total costs are minimized.

A branch-and-price algorithm is proposed to solve the string-based models.

[27] use FS representation to improve crew scheduling by incorporating key maintenance routing decisions. The main contribution of the study is the integration of maintenance routing and crew pairing problems with FSs. [66] propose a string-based fleet assignment model similar to the one proposed in [10]. The authors use FSs to reduce hub connectivity and increase robustness of fleet assignments.

In a recent study, [62] uses FS representation for fully integrated airline recovery problem. A flight string is defined to be a sequence of flights, with timing decisions, to be operated by the same aircraft. Note that the same sequence of flights may be present in more than one FS, each corresponding to different departure time decisions. A single model for each of the schedule recovery, aircraft recovery, crew recovery and passenger recovery is proposed. Four of these models are string-based formulations. A Benders' decomposition scheme is proposed to decompose the integrated problem where schedule recovery problem is naturally selected to be the master problem.

[62] state that the main advantage of FS representation is in the ease of obtaining integer solutions for the routing problem. Moreover, it enables flexible models which involve nonlinear, complex costs and constraints. Note that such objective functions and constraints can easily be linearized since the coefficients of flight strings can be calculated prior to solving the mathematical models. On the other, the main drawback is the huge number of FSs to be generated. When timing decisions are included in FSs, this number grows exponentially. Authors generally propose to select *eligible* FSs to reduce the problem size. Most formal size reduction procedure is explained by [62]. Proposed preprocessing approach is a straightforward method that identifies *disrupted* and *disruptable* resources based on some definitions. Set of eligible FSs are generated with respect to these resources. However, resources that are not disrupted, or disruptable, however can be used to recover disrupted operations are not utilized. For instance, consider an aircraft experiencing an unscheduled maintenance

and having a delayed ready time. Then, flights assigned to this aircraft are disrupted; and crew members and passengers related with these flights are disrupted, as well. On the other hand, an aircraft with a remarkable amount of idle time that is not disrupted or disruptable is left out of the solution space. Note that such a resource would be beneficial since it can carry out the disruptable flights and prevent delay propagation.

In our proposed problem representation, we use a connection-based network. Therefore, we do not enumerate FSs, but instead optimal FSs are generated by the network-based mathematical models. Note that there is a great difference in size between generating all flight connections and generating all FSs (or paths in the CN). We agree with problem size reduction considerations and place a special emphasis on preprocessing methods. In Chapter 4, we propose two important preprocessing methods that provide great reductions in problem size without compromising optimality. Moreover, in Chapter 5, we propose *Isolation Heuristic* that is independent of the optimization models but dedicated to problem size reduction. To the best of our knowledge, this heuristic is the first approach to heuristically solve fully integrated airline recovery problems. Moreover, instead of a straightforward method, the heuristic has parameters to control the tradeoff between solvability of the instances and quality of the solutions.

### 2.3.3 Connection Network Representation

Connection networks are often named as *flight networks*, and *activity-on-node* networks in the literature.

[73] proposes a flight network, or a connection network, representation for swapping applications in fleet assignment problem. Iin the proposed CN representation, flights are represented with arcs, and hence, it differs from the CN definition in this thesis (not an activity-on-node network). In addition to flights, proposed network includes ground arcs and overnight arcs. The nodes, on the other hand, are called *turn nodes*, and are used to represent connecting possibilities between arriving and departing flights. An example presented by the authors is given in Figure 2.5. The example network represents the daily operations at station l.

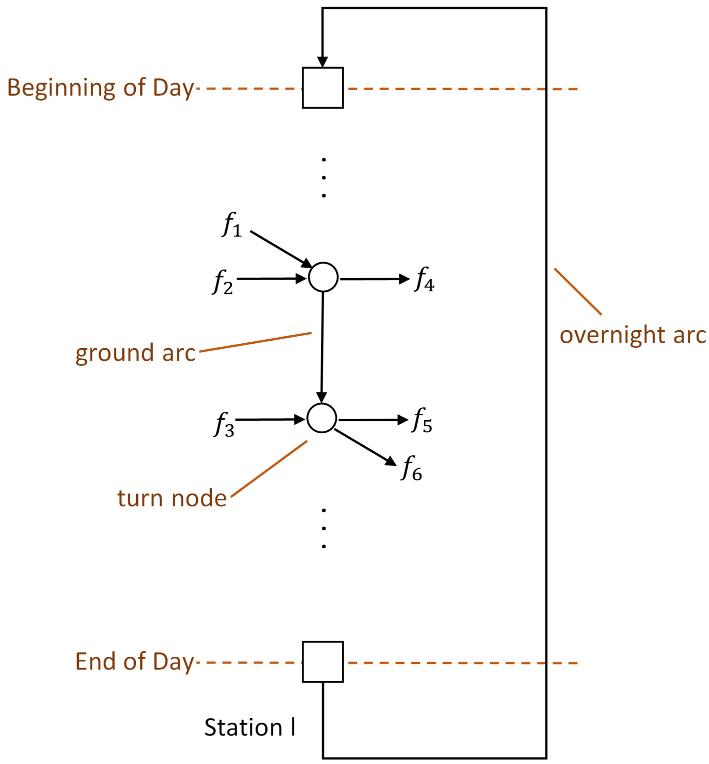


Figure 2.5: CN representation presented in [73].

[67] use CN representation to deal with large scale fleet assignment problems. Proposed representation is called as *event-activity network* representation. A node in the event-activity network designated by $(f, e)$ represents the start event of flight $f$ by fleet type $e$. Existence of an arc designated by $(f, g, e)$ in the network represents the availability of a connection between the starting events of flights $f$ and $g$ for type $e$. These arcs are named as *flight activity arcs*. With this representation, fleet assignment problem can be considered as a fixed-schedule time-constrained routing problem.

[5] propose a solution approach based on a CN representation for aircraft recovery problem. Underlying networks include three kinds of node: source nodes, flight nodes and flight sink nodes. Each node belongs to an airport. Source nodes are used to represent the position of aircraft at the start time. Each flight sink node is associated with a single aircraft, the one that is originally assigned. This forces to resume the original schedules after the end time.

[70] use a CN representation in the approach that tries to integrate certain aspects of schedule design, fleet assignment, aircraft routing, and crew scheduling problems.

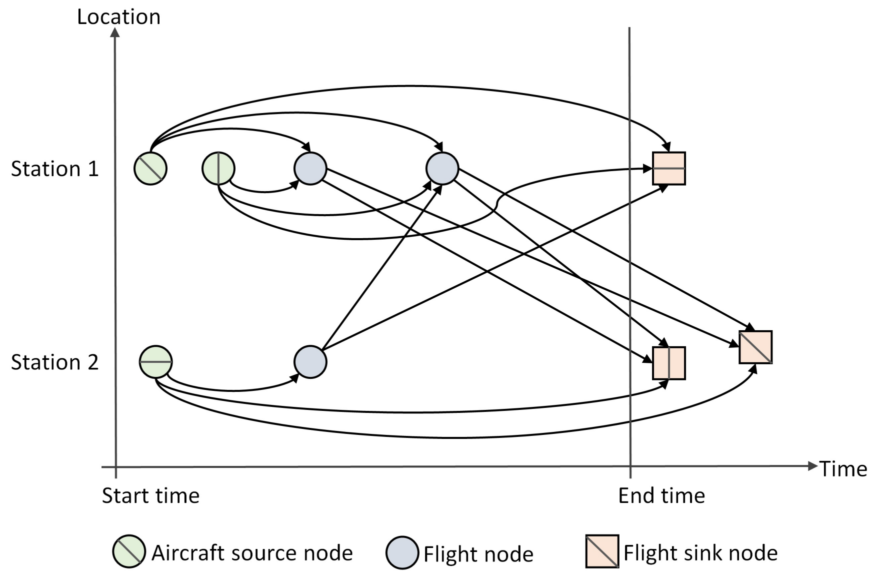


Figure 2.6: CN representation presented in [5].

Each flight is represented by a node $j$ ($j = 1, ..., n$, where $n$ is the number of flight legs). Moreover, nodes $n + s$ are included in the network to represent different stations. Three types of arcs are used in the network:

- $(n + s, j) \Leftrightarrow$ station $s$ is the departure station of flight $j$;
- $(j, n + s) \Leftrightarrow$ station $s$ is the arrival station of flight $j$;
- $(j, k) \Leftrightarrow$ an aircraft can cover flight $k$ immediately after $l$.

An example CN is proposed by the authors is displayed in Figure 2.7. Feasible periodic routings of two aircraft are designated by bold arcs. Note that each of these routings start at a station node, visits some flight nodes and arrives at a station node.

CN representations discussed in this section have significant differences depending on the problem type that the authors are dealing with. The alternative representation that we present in Chapter 4 can be classified as a CN representation. Structure of the proposed CNs can be summarized with the following:

- activities (flights, maintenances, etc.) are represented in nodes,

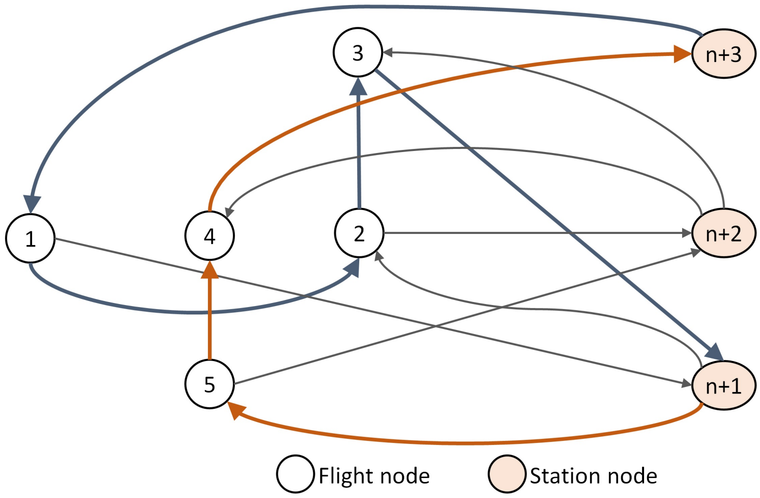


Figure 2.7: CN representation presented in [70].

- existence of an arc represents the possibility of sequentially operating the corresponding pair of activities by a single entity (aircraft, crew, and passenger).

These simple rules construct a general representation. Note that only aircraft routings are represented by CNs in the studies that we discuss in this section. We model crew and passenger operations with CNs in addition to aircraft operations. This allows to easily model the dependencies between different entity types in a mathematical model or in an algorithm. To the best of our knowledge, this thesis is the first study in which integration of different entities is represented by a CN representation.

## 2.4 Cruise Speed Control Action

We suggest readers [2], [3] and [19] to understand flight stages, fuel economy, impact of cruise stage on total trip cost, and considerations in cruise speed optimization. In the study of [28], delay recovery opportunities by speeding up some flights to some extent are discussed. Furthermore, [57] present a novel approach on fuel cost minimization by optimizing cruise speeds and altitude profiles.

The approach proposed by [59] integrates cruise speed control as a recovery action in the integrated aircraft and passenger recovery problem. A flight planning engine, JetPlan, is used to generate the flight plans used in the solution approach. The authors show that disruption and recovery costs can be significantly improved by considering the network effects on time-related costs when compared with the state-of-the-practice at airlines. Our alternative problem representation and fuel cost expression as a function of cruise time (similar to the one proposed by [57]) enable to represent cruise speed decisions with continuous variables. In this thesis, we try to contribute by proposing formulations that allow continuous cruise speed optimization while all possible recovery actions are optimized simultaneously in fully integrated airline recovery problems.

[4] is the first study to integrate cruise speed controllability in an aircraft recovery problem with environmental constraints and cost coefficients. In order to optimally solve the tradeoff between increased fuel cost and disruption costs, the authors propose a fuel cost function based on the fuel flow model of Base of Aircraft Data (BADA) project of EUROCONTROL (the air traffic management organization of Europe). When the cruise speed of a flight is increased, $CO_2$ emission is increased as well, and an additional cost is incurred. The authors also contribute by integrating $CO_2$ emission cost in the airline recovery problem. The study presents nonlinear mixed integer formulations for single and multi fleet aircraft recovery problems, which are then strengthened by conic quadratic reformulation. Furthermore, the authors propose extensions to handle flight delay costs represented by nonlinear and step functions in addition to the linear relation used in the preceding formulations.

The approach proposed by [33] falls into robust airline scheduling category. In robust airline scheduling, the uncertainty in operations is taken into consideration while cruise speed is usually considered as a fixed parameter. [33] consider cruise times as controllable and non-cruise times as uncertain. By integrating cruise speed controllability, the authors manage to add flexibility to the problem. The experimentations show that 60% of idle time costs can be reduced by only a 2% increase in fuel costs. This result points out the effectiveness of cruise speed control action in schedule planning phase.

## 2.5 Passenger-Related Disruption Costs

In cases of disruptions or as a consequence of recovery decisions, passengers may be spilled or they may experience arrival delays. Generally, a cost coefficient is assigned to each passenger in each passenger itinerary to represent the cost of spilling (also called passenger disruption cost in some studies). On the other hand, it is not an easy task to evaluate the cost of arrival delays experienced by passengers. Furthermore, passenger delay cost formulation in integrated airline recovery problems is challenging due to the complexity of the problem.

[20] present two optimization models for the problem. In the first one, called Disrupted Passenger Metric (DPM) model, passenger disruption costs, i.e. spill costs, are evaluated. In Passenger Delay Metric (PDM) model, on the other hand, a more accurate passenger-related cost calculation method including passenger delay costs is presented. The formulation benefits from the underlying TSN and flight copy representation. For flight copies each associated with different arrival times, a passenger delay cost coefficient can be assigned. This allows to handle any relation between the amount of arrival delay and corresponding passenger delay cost, while a linear relationship is used in the experimentations. In a similar approach [59] use a simulator to compute the passenger delay costs of flight copies. However, TSN representation requires complete enumeration of passenger rerouting options. Furthermore, it is difficult to obtain real time solutions due to the increased problem size with TSN representation. [20] report that DPM is fast enough to provide real time solutions, while PDM cannot be solved in real time for practical-sized problems. Similarly [59] propose an approximate aircraft and passenger recovery model to provide real time solutions. The approximate model calculates total passenger delay cost of a solution assuming that all passengers are transported as scheduled. In other words, the method ignores rerouting and spilling decisions of passengers, and hence, actual delay cost can be underestimated or overestimated by the model.

Even though the authors use a FS representation, the approach of [62] resembles the studies of [20] and [59] in the sense that all passenger rerouting opportunities together with retiming decisions of flights are enumerated. Therefore, any passenger delay cost function can be incorporated. The drawback is again with the number of alternatives to be enumerated. The authors, hence, propose to generate only eligible alternative itineraries in the passenger recovery process. In the computational study, the authors use a linear relationship between the realized delay and passenger delay cost.

Passenger delay cost can be regarded as the loss of goodwill cost for the carrier. On the other hand, with the passengers' point of view, it may be regarded as the amount of money a passenger is willing to pay for an arrival with one minute less delay. The value of this parameter may be different for each passenger and it may even vary for an individual in different days, in different times of the day and in different origin-destination pairs. [36] presents a novel study that describes the use of choice theory in air transportation for modelling passenger behavior. [37] state that from a passenger's perspective, representing the passenger behavior using 'door-to-door' time instead of 'airport-to-airport' time would be more accurate and proposes a formulation to calculate the values of times. There are studies in the literature that model the behavior of business class passengers, such as [40]. The authors study ticketing, refund and exchange behavior of business class passengers depending on factors such as frequency of travel, carrier, time from ticket purchase and time before flight departure. In terms of delay costs for business class passengers, we can intentionally state that value of time is greater than that for economy class passengers.

While most studies in the literature work with a linear passenger delay cost function, some authors believe that the relation between the amount of delay and the delay cost is nonlinear and convex. In the formulations that we present in Chapter 4, we try to evaluate passenger delay cost as accurate as possible. In addition to linear delay cost function, we also formulate a piecewise linear function to approximately model the nonlinear relationship. Similar to the approaches of [20] and [59], we propose approximation models. Furthermore, we propose models that calculate passenger delay cost of the solutions based on the actual (realized) delay of passengers in order to solve the tradeoff more accurately. In order to achieve actual delay calculation, we model each passenger explicitly rather than aggregating the passengers in the same fare class of an itinerary. This approach has an additional advantage that different cost functions can be applied to each individual passenger. To the best of our knowledge, this study is the first to model passengers explicitly and provide real time solutions.

# CHAPTER 3

# INTEGRATED AIRCRAFT AND PASSENGER RECOVERY WITH CRUISE TIME CONTROLLABILITY

In this chapter, we present a mathematical formulation for the integrated aircraft and passenger recovery problem that considers aircraft and passenger related costs simultaneously. The problem is explained on an example in Section 3.1. In Section 3.2, we propose a mathematical model for aircraft-passenger recovery problem which handles departure and arrival delays of any severity. Superimposition of aircraft and passenger networks is considered to create an integrated recovery plan. The objective is to minimize the total recovery cost which includes routing-related costs together with passenger-related costs. Passengers are discretized both by their itineraries and classes. We aim to contribute by proposing an integrated solution approach that reschedules departure and arrival times, swaps aircraft, determines passenger itineraries that will be disrupted and finds optimal cruise speeds of flights. In addition to more commonly used recovery actions, we aim to contribute by including cruise speed control in the alternative courses of recovery actions. As discussed in Section 1.3, fuel consumption increases with cruise speed. In order to find the optimal trade-off between fuel consumption and delay propagation, fuel cost is expressed as a nonlinear function of cruise time and included in the objective function. The problem is formulated as a mixed integer nonlinear programming (MINLP) model. In Section 3.3, we show that the model can be represented as a conic quadratic mixed integer programming (CQMIP) problem. Finally, we generate conic quadratic constraints to solve the problem with commercial CQMIP solvers, such as CPLEX. We test our approach using publicly available schedules of a major U.S. airline. We present the computational study of the proposed approach In Section 3.4. Our experiments have shown that we could solve the integrated aircraft and passenger recovery problem to optimality on a four-hub network of a major U.S. airline in less than one minute on the average. Finally, we present our conclusions on the proposed approach and experimentations in Section 3.5.

## 3.1 Numerical Example

Before presenting the mathematical formulation, we illustrate the main idea on a small-sized numerical example. The input of aircraft and passenger recovery problem consists of the flight schedule, characteristics of the assigned aircraft (such as seat capacities and fuel efficiencies), planned passenger itineraries, numbers of passengers of each class assigned to these itineraries, and lastly a set of disrupted flights with the severities of these disruptions. Flight schedule of the example is illustrated on a time-space network in Figure 3.1. Vertical axis corresponds to space (airports) while the horizontal axis is the time line. Each line style (color) represents a different aircraft routing. The horizontal line segments of a routing represents the time that the aircraft spends at the airport. Length of these line segments need to be at least as long as the turn time of the aircraft. Turn time is the minimum time required after the arrival of an aircraft to be ready for its next departure. In the example problem, turn times for all flights are set to 30 minutes. Inclined line segments, on the other hand, correspond to planned flights. For instance, N5FCAA performs two flights: a 440-minute flight from HNL to DFW, and a 440-minute flight from DFW back to HNL. The aircraft spends 70 minutes at DFW between these flights.

Flight schedule is tabulated in Table 3.1, and operational details of these flights are presented in Table 3.2. The rightmost column of Table 3.2 corresponds to the allowable compression percentage which is the aircraft-and-flight-dependent parameter indicating the allowable compression of flight time expressed in terms of the percentage of cruise time. For instance, cruise time, and hence the flight time, of flight 1 can be decreased by 4.8 minutes at most, if the aircraft of the flight is not swapped.

Passenger itineraries expressed in terms of sequential flight numbers are listed in Table 3.3 (the second and third columns represent the numbers of passengers in econ-

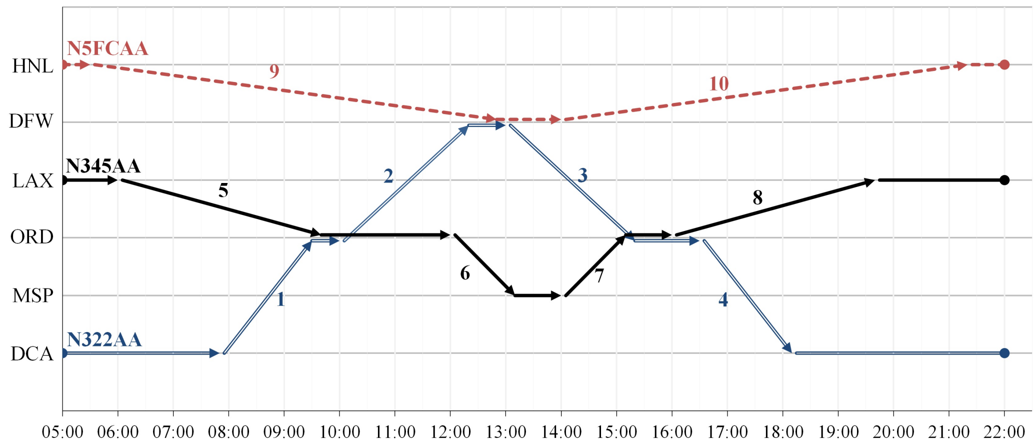


Figure 3.1: Time-space network displaying the flight schedule of three aircraft routings

omy and business classes, respectively); and space network of flights and passenger itineraries are represented in Figure 3.3. It is important to note that we deal with the integrated network, which is the superimposition of aircraft routing and passenger itinerary networks. For instance, passengers in itinerary 1-2-10 are three-flight passengers who travel from DCA to ORD, ORD to DFW, and finally DFW to HNL (Figure 3.1). At DFW, passengers in this itinerary switch from N322AA to N5FCAA. Aircraft schedules must provide sufficient time for connected flights, which is called the minimum connection time. In this example, minimum connection times are all set to 30 minutes. When the minimum connection time is not satisfied, the itinerary is said to be disrupted. Spill costs of all itineraries in this example are $50 and $200 for economy and business class passengers, respectively. Delay costs are separated into two terms: delay cost for aircraft and delay cost for passengers. Aircraft delay cost contains several components such as maintenance cost, flight crew and cabin crew cost, and marginal depreciation or lease cost per flight minute. On the other hand, passenger delay cost is the estimated cost of time lost by passengers. Aircraft delay costs are all set to $20 per minute; and passenger delay costs per passenger minute are set to $0.05 and $2 for economy and business class passengers, respectively.

In the given scenario, N322AA is facing maintenance problems before its first flight, and hence, it cannot depart before 09:50. In other words, departure time of flight 1 is delayed by two hours. A common recovery strategy for such disruptions is to

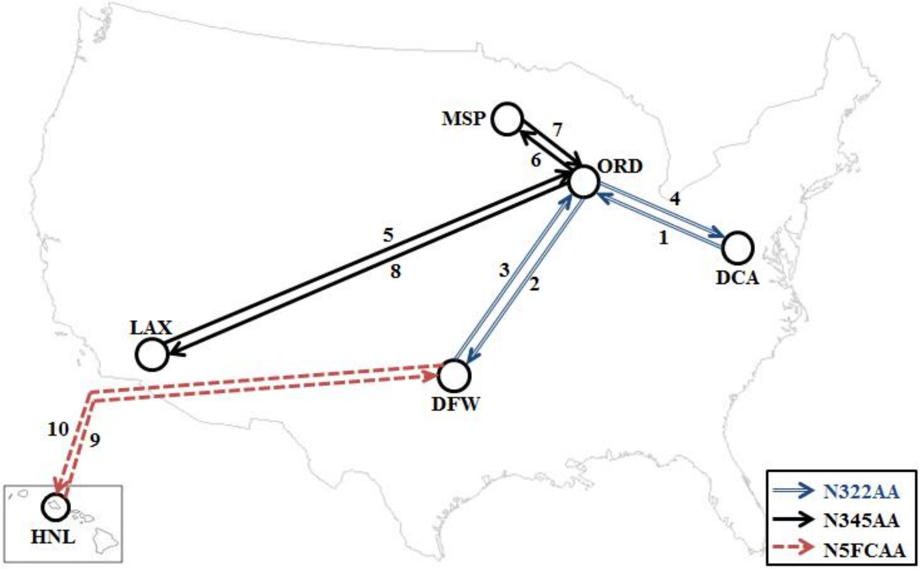


Figure 3.2: Aircraft Routings

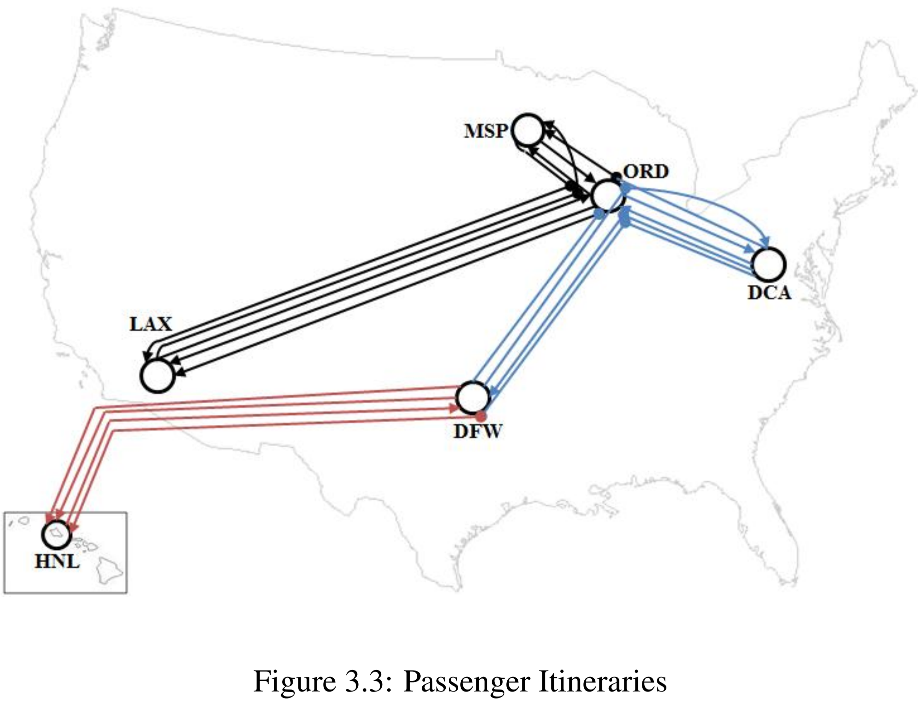


Figure 3.3: Passenger Itineraries

Table 3.1: Planned flight schedule

| Tail # | Flight # | From | To | Departure time | Arrival time |
|---|---|---|---|---|---|
| N322AA | 1 | DCA | ORD | 7:50 | 9:30 |
| | 2 | ORD | DFW | 10:00 | 12:20 |
| | 3 | DFW | ORD | 13:00 | 15:20 |
| | 4 | ORD | DCA | 16:30 | 18:10 |
| N345AA | 5 | LAX | ORD | 6:00 | 9:40 |
| | 6 | ORD | MSP | 12:00 | 13:10 |
| | 7 | MSP | ORD | 14:00 | 15:10 |
| | 8 | ORD | LAX | 16:00 | 19:40 |
| N5FCAA | 9 | HNL | DFW | 5:30 | 12:50 |
| | 10 | DFW | HNL | 14:00 | 21:20 |

postpone all flights until their assigned aircraft are ready. This strategy is known as *push-back recovery plan* (**PB1**). A variation of PB1 is *push-back recovery plan maintaining passenger connections* (**PB2**) which may be a good idea when spill costs overweigh delay costs. In PB2, none of the itineraries are disrupted since the departures of all flights are postponed until aircraft and passengers are ready. On the other hand, more flights are delayed and greater delay propagation is experienced.

In this study, we propose a new approach with the objective of finding minimum cost recovery plans by deciding on which flights to postpone and how much to postpone, which itineraries to disrupt, cruise times of which flights to compress and how much to compress, and the aircraft of which flights to swap. This integrated approach will be named as aircraft-passenger recovery (APR). Swapping aircraft may provide opportunities to mitigate delays; however, the characteristics of the aircraft should be considered while swapping aircraft. In particular, different types of aircraft may have different fuel efficiencies; and hence, flying the same flight leg with different aircraft may result in different fuel costs. For simplicity, all aircraft in the example have similar fuel efficiencies, and fuel cost of a flight is computed by $0.25 \times \texttt{Distance}^{2.5}/crt^{1.5}$, where $crt$ is the cruise time of the flight. Compression limitations may also vary as displayed in Table 3.2. Finally, aircraft may have different seat capacities. Therefore, a swap decision may result in spilled passengers due to insufficient seat capacity. In the example, capacities of aircraft N322AA, N345AA,

Table 3.2: Details of scheduled flights

| Tail # | Flight # | Cruise time (min.) | Time to next flight (min.) | Dist. (miles) | Allowable compr. |
|---|---|---|---|---|---|
| N322AA | 1 | 80 | 30 | 610 | 6% |
| | 2 | 120 | 40 | 800 | 6% |
| | 3 | 120 | 70 | 800 | 6% |
| | 4 | 80 | - | 610 | 6% |
| N345AA | 5 | 200 | 140 | 1745 | 10% |
| | 6 | 50 | 50 | 335 | 10% |
| | 7 | 50 | 50 | 335 | 10% |
| | 8 | 200 | - | 1745 | 10% |
| N5FCAA | 9 | 70 | 3780 | 9% | |
| | 10 | 420 | - | 3780 | 9% |

and N5FCAA are 270-30, 243-27 and 234-26 for economy and business class passengers, respectively. When the aircraft of two flights are swapped, a swap cost is incurred for returning to the original schedule at the end of the day.

Three recovery schemes are displayed in Figures 3.4, 3.5, and 3.6 . Lines in grey color represent the flights in the original schedule. In Figure 3.4, it can be seen that only flights in the routing of N322AA are delayed with PB1 recovery policy. Total delay is 420 minutes, resulting in a delay cost of $36,374. On the other hand, the connection between flights 2 and 10, and the connection between 3 and 8 are disrupted. Therefore, the passengers in itineraries 1-2-10, 2-10 and 3-8 are spilled. A total of 87 economy and 8 business class passengers are spilled, and the resulting total spill cost is $5,950. As given in Figure 3.5, in PB2 recovery policy, all connections are maintained; however, in addition to flights in the disrupted routing, flights 8 and 10 are also delayed. Total delay is 570 minutes and the corresponding delay cost is $45,676.

Figure 3.6 presents the solution obtained by APR approach. APR utilizes the swap opportunity between flights 2 and 6. In the recovery scheme, N322AA operates flights 1, 6, 7 and 8, while N345AA operates flights 5, 2, 3 and 4. Resulting swap cost is $ 5,000. In order not to delay flight 2, cruise time of flight 5 is compressed by 10 minutes so that it arrives 30 minutes (turn time) before the scheduled departure of flight

Table 3.3: Planned passenger itineraries

| Itinerary | # of pass. Economy | # of pass. Business | Itinerary | # of pass. Economy | # of pass. Business |
|---|---|---|---|---|---|
| 1 | 183 | 19 | 4 | 113 | 15 |
| 1-2 | 33 | 4 | 5 | 65 | 13 |
| 1-2-10 | 12 | 2 | 5-6 | 90 | 8 |
| 1-6 | 42 | 5 | 6 | 57 | 6 |
| 2 | 153 | 18 | 7 | 130 | 16 |
| 2-10 | 72 | 6 | 7-8 | 113 | 11 |
| 3 | 67 | 7 | 8 | 51 | 5 |
| 3-4 | 137 | 13 | 9 | 234 | 26 |
| 3-8 | 3 | 0 | 10 | 103 | 10 |

2. It is important to note that flight 5 is not originally affected from this disruption, but its cruise time is compressed (in albeit of additional fuel cost) due to the swapping decision. Such an interaction will not be foreseen without a global optimization model such as the one proposed in this study. The current industry solution of cost index ratio is useful but the flight crew may not be able to see the consequences of their local decisions. In this solution, passengers in itineraries 1-2 and 1-2-10 are disrupted; and hence, 45 economy and 6 business class passengers are spilled. APR captures the change in aircraft capacities due to the swap; 7 economy and 1 business class passengers are spilled from flight 4. Resulting spill cost is $4,000. Delay is not propagated and only flight 1 is delayed. Moreover, cruise time of flight 1 is compressed by 4.8 minutes leading a total delay of only 115.2 minutes. Corresponding total delay cost is $10,771. Speeding up two flights costs $5,757 for increased fuel consumption. Some performance measures and cost terms of all recovery schemes are summarized in Table 3.4. It can be observed from the cost values that APR finds the optimal tradeoff between the cost terms by optimizing a unified objective function. In this example, a 40% reduction in total recovery costs with respect to those of PB1 and PB2 schemes is provided.

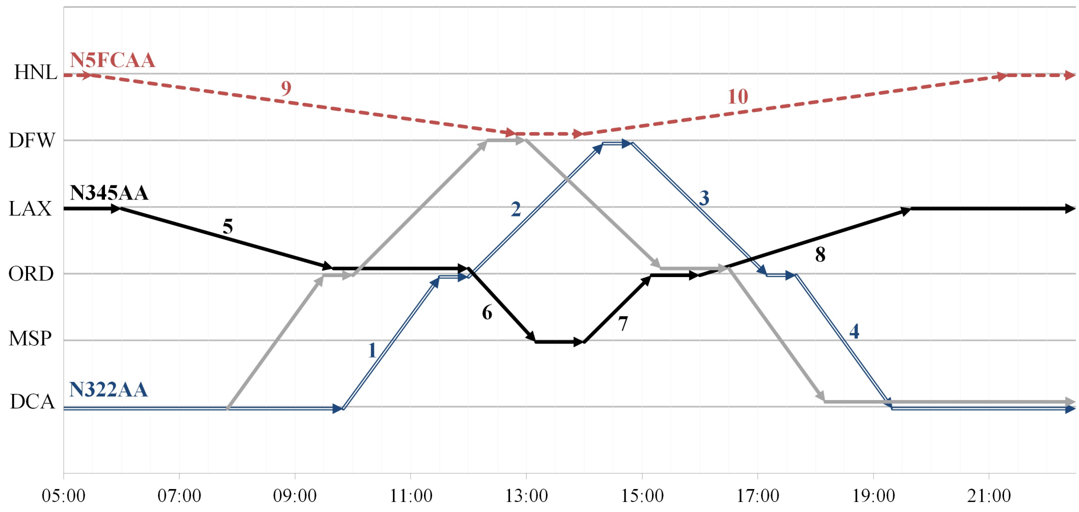


Figure 3.4: PB1 solution

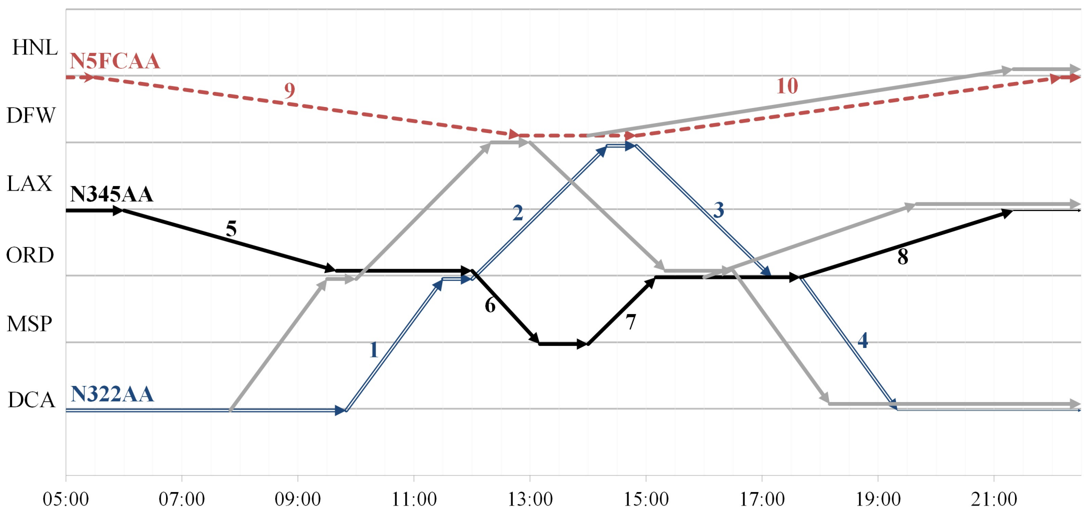


Figure 3.5: PB2 solution

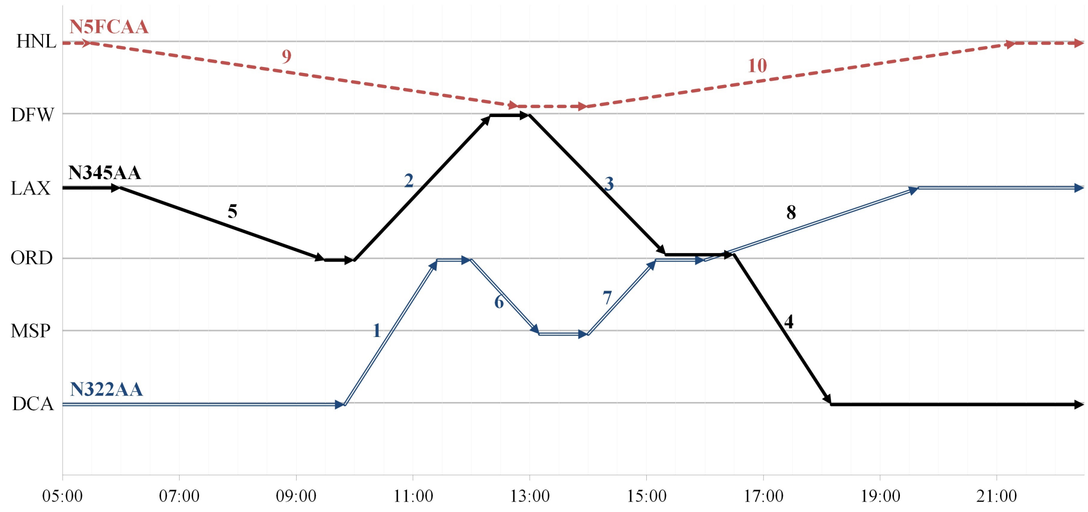


Figure 3.6: APR solution

## 3.2 Mathematical Formulation

The objective of the aircraft and passenger recovery problem is to find the minimum cost recovery plan given aircraft routings and aircraft characteristics, passenger itineraries and number of assigned passengers of each passenger class, and a set of departure or arrival delays of any severity. The decisions include which flights to postpone and how much to postpone, which passenger itineraries to maintain and which to cancel, aircraft of which routings to swap and which flights to speed up. Objective function consists of five cost terms: delay cost for aircraft and passengers, spill cost, swap cost and additional fuel cost. Derivation of the general expression for the nonlinear fuel cost will be given in this section, and conic quadratic reformulation will be explained in Section 3.3.

### 3.2.1 Fuel Cost Function

Since we aim to use cruise speed control of flights in recovery, a realistic fuel cost function is required. In the technical documentations of [2] and [19], it is reported that direct operating costs of a flight consists of a fixed cost, and variable fuel and time related costs depending on cruise speed and time. However, in the aircraft and

Table 3.4: Performance measures of solutions

| Performance measure | PB1 | PB2 | APR |
|---|---|---|---|
| # of delayed flights | 4 | 6 | 1 |
| Total delay (min.) | 420 | 570 | 115.2 |
| # of disrupted itineraries | 3 | 0 | 2 |
| # of spilled passengers | 95 | 0 | 59 |
| Total compression (min.) | 0 | 0 | 14.8 |
| Aircraft delay cost | 8,400 | 11,400 | 2,304 |
| Passenger delay cost | 27,974 | 34,276 | 8,467 |
| Total spill cost | 5,950 | 0 | 4,000 |
| Additional fuel cost | 0 | 0 | 5,757 |
| Swap cost | 0 | 0 | 5,000 |
| Total recovery cost | 42,324 | 45,676 | 25,528 |

passenger recovery problem, time related costs of flights should be evaluated considering the superimposition of aircraft and passenger networks instead of considering flights separately. Therefore, we calculate these costs in the mathematical model as aircraft and passenger delay costs. On the other hand, we develop the fuel cost function of a flight with respect to the technical report of [3]. It is reported that airlines generally operate their aircraft at maximum range cruise (MRC) speeds that result in minimum fuel consumption. The relation of fuel cost with deviation from MRC speed is increasing and convex as displayed in Figure 3.7. In (3.1), we express actual fuel cost of flight $k$ ($fc_{kt}$) when operated with the aircraft of routing $t$ as the deviation from the planned (minimum) fuel cost ($FC_{kt}$), where $V_{kt}$ is the MRC speed, $\Delta v_k$ is the increase in the cruise speed and $K_{kt}$ is flight and aircraft related parameter.

$$fc_{kt}(\Delta v_k) = FC_{kt} \times \left( \frac{V_{kt} + \Delta v_k}{V_{kt}} \right)^{K_{kt}} \tag{3.1}$$

where $K_{kt} \geq 1$.

In (3.2) we express the relation between the cruise speed ($v_k$) and cruise time ($crt_k$) of flight $k$, where $DIST_k$ is the cruise stage distance. A similar approximation was also employed in [57] to analyze the potential of cruise fuel burn savings.

$$DIST_k = v_k \times crt_k \tag{3.2}$$

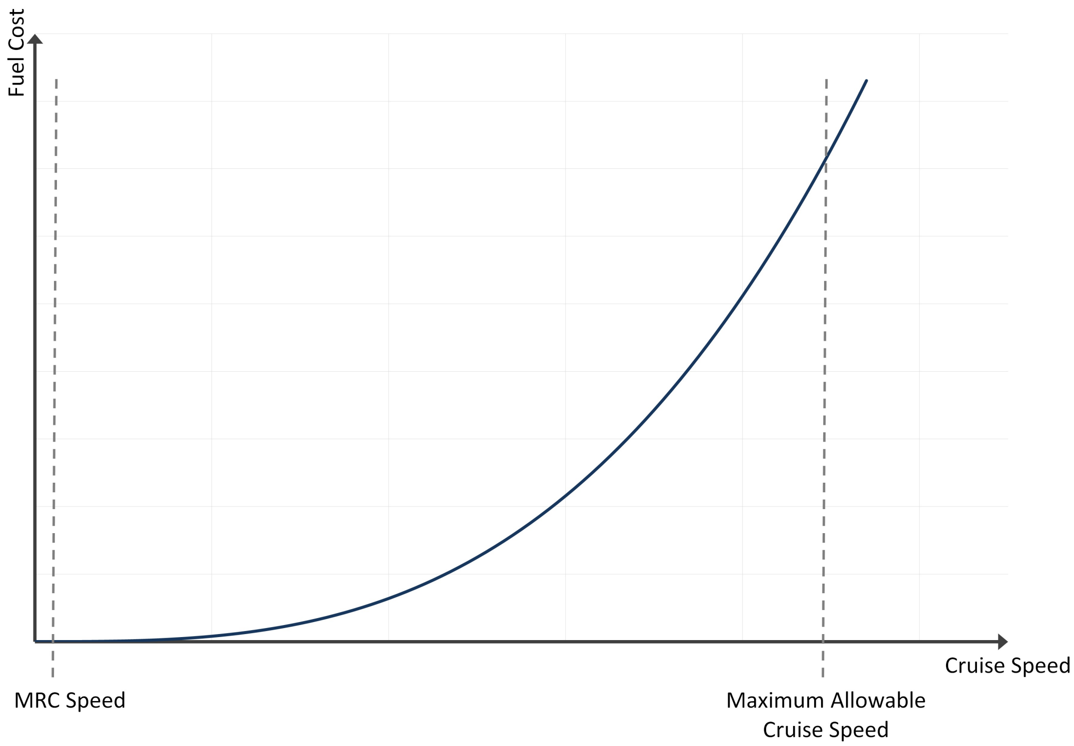


Figure 3.7: Cruise speed - fuel cost relation

Using relations in equations (3.1) and (3.2), Proposition 3.2.1 expresses fuel cost as a function of cruise time.

**Proposition 3.2.1.** *Cruise stage fuel cost of flight $k$, when operated with the aircraft of routing $t$ ($fc_{kt}$) and when the cruise time is compressed by $\Delta t_k$, can be expressed as a function of the amount of compression as follows:*

$$f_{kt}(\Delta t_k) = FC_{kt} \times T_{kt}^{K_{kt}} \times \left(\frac{1}{T_{kt} - \Delta t_k}\right)^{K_{kt}} \tag{3.3}$$

*where $T_{kt}$ is the cruise time of the flight at planned cruise speed.*

*Proof.* Using the relation in (3.2), we have both $DIST_k = V_{kt} \times T_{kt}$ and $DIST_k = (V_{kt} + \Delta v_k) \times (T_{kt} - \Delta t_k)$. Therefore, increase in cruise speed can be expressed in terms of the amount of compression.

$$\Delta v_k = \frac{V_{kt} \times \Delta t_k}{T_{kt} - \Delta t_k} \tag{3.4}$$

Substituting (3.4) in (3.1), actual cruise stage fuel cost of a flight can be expressed in terms of the amount of compression in the cruise time. □

### 3.2.2 Notation

Before going through the formulation, notation used throughout this chapter is described below.

**Sets:**

| | | |
|---|---|---|
| $F$ | Set of all scheduled flights | |
| $R$ | Set of all scheduled aircraft routings | |
| $F_t$ | Set of flights in $t^{th}$ aircraft routing | $t \in R$ |
| $I$ | Set of passenger itineraries | |
| $I(k)$ | Set of passenger itineraries including flight $k$ | $k \in F$ |
| $F(i)$ | Set of flights in passenger itinerary $i$ | $i \in I$ |
| $S$ | Set of pairs of flights that can be swapped | |
| $D$ | Set of flights that have departure or arrival time disruption | |
| $Pre_k$ | Set of flights including flight $k$ and all flights that precede flight $k$ in its routing, | $k \in F$ |
| $C$ | Set of passenger classes | |

In general, aircraft of all flights departing from the same airport can be swapped:

$$S = \{(k, m) : k, m \in F, m > k, R_k \neq R_m, ORI_k = ORI_m\}.$$

$R_k$ is the tail number of the $k^{th}$ flight and the parameter $ORI_k$ denotes the origin airport of flight $k$. Flight pairs in set $S$ can be determined by considering various factors. If two aircraft have a long time gap between their arrivals to the airport where swap occurs, then such a swap may be excluded from $S$. If a swap has a very high cost, it causes infeasibilities in crew schedules or in aircraft maintenance restrictions, then it should be excluded from $S$ as well.

**Parameters:**

| | | |
|---|---|---|
| $R_k$: | Routing (tail) number of flight $k$ | $k \in F$ |
| $DT_k$: | Scheduled departure time of flight $k$ | $k \in F$ |
| $AT_k$: | Scheduled arrival time of flight $k$ | $k \in F$ |
| $FT_{kt}$: | Flight time of flight $k$ if it is flown at planned speed with aircraft of routing $t$ | $k \in F, t \in R$ |

| | | |
|---|---|---|
| $T_{kt}$: | Cruise time of flight $k$ if it is flown at planned speed with aircraft of routing $t$ | $k \in F, t \in R$ |
| $\Gamma_k$: | Aircraft turn time for flight $k$ | $k \in F$ |
| $\Theta_{ik}$: | Minimum connection time for passengers in itinerary $i$ before flight $k$ | $i \in I, k \in F(i)$ |
| $CAP_t^c$: | Seat capacity for passengers of class $c$ in the aircraft assigned to the routing $t$ | $t \in R, c \in C$ |
| $APre_k$: | Flight that immediately precedes flight $k$ in its routing | $k \in F$ |
| $First_i$: | First flight in passenger itinerary $i$ | $i \in I$ |
| $PPre_{ik}$: | Flight that immediately precedes flight $k$ in passenger itinerary $i$ | $i \in I, k \in F(i)$ |
| $\epsilon_i$: | Difference between the ready time of passengers in itinerary $i$ and scheduled departure time of their first flight | $i \in I$ |
| $NPass_k^c$: | Total number of class c passengers in flight $k$ | $k \in F, c \in C$ |
| $NP_i^c$: | Number of class $c$ passengers in itinerary $i$ | $i \in I, c \in C$ |
| $K_{kt}$: | Fuel cost exponent of aircraft of routing $t$ for operating flight $k$ | $t \in R$ |
| $All_{kt}$: | Percentage of cruise time of flight $k$ that can be compressed with aircraft of routing $t$ | $k \in F, t \in R$ |
| $FC_{kt}$: | Minimum (planned) fuel cost of flight $k$ if operated with aircraft of routing $t$ | $k \in F, t \in R$ |
| $Y_{t_1 t_2}$: | Swap cost incurred if the aircraft of routings $t_1$ and $t_2$ are swapped | $t_1, t_2 \in R$ |
| $ADC_k$: | Aircraft delay cost per flight minute of aircraft | $k \in F$ |
| $PDC_k^c$: | Passenger delay cost per minute per passenger of class $c$ for flight $k$ | $k \in F, c \in C$ |
| $SC_i^c$: | Spill cost per passenger of class $c$ in itinerary $i$ | $i \in I, c \in C$ |
| $DD_k$: | Amount of departure delay of disrupted flight $k$ | $k \in D$ |
| $AD_k$: | Amount of arrival delay of disrupted flight $k$ | $k \in D$ |
| $M$: | A sufficiently large amount of time | |

**Decision Variables:**

| | | |
|---|---|---|
| $dt_k$: | Departure time of flight $k$ in the recovered schedule | $k \in F$ |
| $at_k$: | Arrival time of flight $k$ in the recovered schedule | $k \in F$ |
| $\Delta t_k$: | Amount of compression in cruise time of flight $k$ in the recovered schedule | $k \in F$ |
| $delay_k$: | Amount of arrival delay of flight $k$ in minutes in the recovered schedule | $k \in F$ |
| $sp_i^c$ | Number of class $c$ passengers in itinerary $i$ that are spilled due to itinerary disruption or capacity insufficiency | $i \in I, c \in C$ |
| $pass_k^c$: | Number of class $c$ passengers that are not disrupted and available for flight $k$ | $k \in F, c \in C$ |
| $z_i$= | 1, if passenger itinerary $i$ is disrupted; 0 otherwise | $i \in I$ |
| $x_{km}$= | 1, if aircraft of flights $k$ and $m$ are swapped; 0 otherwise | $(k, m) \in S$ |
| $y_{kt}$= | 1, if flight $k$ is performed with aircraft of routing $t$; 0 otherwise | $k \in F, t \in R$ |

### 3.2.3 Constraints

Constraints of the aircraft and passenger recovery problem are discussed in six groups.

#### 3.2.3.1 Swap Constraints

We limit the number of swaps for the flights in a routing to one in (3.5) and introduce the auxiliary variable $y_{kt}$ in (3.6) and (3.7) in order to indicate that a flight is swapped or not.

$$\sum_{k\in F_t}\left(\sum_{m:(k,m)\in S} x_{km} + \sum_{m:(m,k)\in S} x_{mk}\right) \leq \quad 1 \qquad ;t \in R \tag{3.5}$$

$$\sum_{i\in Pre_k}\left(\sum_{\substack{j:(i,j)\in S,\\ j\in F_t}} x_{ij} + \sum_{\substack{j:(i,j)\in S,\\ j\in F_t}} x_{ji}\right) = \; y_{kt}; k \in F, t \in R \setminus \{R_k\} \tag{3.6}$$

$$1 - \sum_{i\in Pre_k}\left(\sum_{j:(i,j)\in S} x_{ij} + \sum_{j:(i,j)\in S} x_{ji}\right) = y_{kR_k} \qquad ;k \in F \tag{3.7}$$

The auxiliary variable $y_{kt}$ takes value one if flight $k$ is operated with the aircraft of routing $t$. Value of $y_{kR_k}$ will be equal to one if the flight is not swapped; i.e., if the flight is operated with the originally scheduled aircraft.

#### 3.2.3.2 Departure Time Constraints

Departure time constraints are developed using the idea that an operation cannot start until all necessary resources are available. Constraints allow early and late departures in order to create a greater feasible region; however, operational restrictions on departure and arrival times can easily be inserted into the model.

$$dt_k \geq \quad (DT_k - \epsilon_i) \times (1 - z_i) \qquad ;i \in I, k = First_i \tag{3.8}$$

$$dt_k \geq \quad (at_j + \Theta_{ij}) - M \times z_i \quad ;i \in I, k \in F(i) \setminus First_i, j = PPre_{ik} \tag{3.9}$$

$$dt_k \geq (at_j + \Gamma_j) - M \times \left(\sum_{m:(k,m)\in S} x_{km} + \sum_{m:(m,k)\in S} x_{mk}\right) \quad ;k \in F, j = APre_k \tag{3.10}$$

$$dt_k \geq \quad (at_j + \Gamma_j) - M \times (1 - x_{km}) \quad ;k \in F, j = APre_m \ni (k,m) \in S \tag{3.11}$$

$$dt_k \geq \quad (at_j + \Gamma_j) - M \times (1 - x_{mk}) \quad ;k \in F, j = APre_m \ni (m,k) \in S \tag{3.12}$$

We allow flights to depart before the ready time of its passengers. Although it seems an undesirable situation, it might be necessary due to a swapping decision. In such a case, constraint (3.8) ensures that the passengers having that flight as the first flight in their itineraries are spilled. Passenger connections are modeled with constraint (3.9). If a flight departs before passengers who have a preceding flight in their itineraries are ready ($at_j + \Theta_{ij}$), the variable $z_i$ takes value one denoting that the connection is disrupted. Constraint (3.10) satisfies the aircraft flow balance of two successive flights in a routing considering turn times if the aircraft of these flights are not swapped. In case of swaps, balance is satisfied with constraints (3.11) and (3.12). For instance, consider constraint (3.11) and suppose that the aircraft of flights $k$ and $m$ are swapped. Then, the constraint states that flight $k$ cannot depart before the ready time of the aircraft after flight $j$ which is the predecessor of flight $m$ in its routing in the original schedule.

#### 3.2.3.3 Arrival Time and Delay Constraints

If the cruise speed is not controlled, arrival time of a flight would be expressed as the sum of the departure time and the planned flight time. Taking compression and swap decisions into consideration, arrival time constraint is given in (3.13). Since customer convenience is related with on-time arrival performance of flights, we express delay as the difference between actual and planned arrival time in (3.14). In order not to promote early arrivals, we force delay to be nonnegative in (3.15).

$$at_k \geq dt_k + \left( \sum_{t \in R} FT_{kt} \times y_{kt} \right) - \Delta t_k \qquad ; k \in F \tag{3.13}$$

$$delay_k \geq at_k - AT_k \qquad ; k \in F \tag{3.14}$$

$$delay_k \geq 0 \qquad ; k \in F \tag{3.15}$$

#### 3.2.3.4 Allowable Compression Constraints

Maximum compression in the cruise time of a flight is limited and we express this limitation as percentage of the cruise time. Limitation on the compression depends both on the aircraft type and the flight. Constraint (3.16) defines the upper bound for compression in cruise time. We assume that an airline plans its flight times at maximum range cruise speed which gives minimum fuel cost. Therefore, a longer cruise time would be worse both in terms of time and cost. Constraint (3.17) enforces compression variable to take nonnegative values.

$$\Delta t_k \leq \sum_{t \in R} T_{kt} \times All_{kt} \times y_{kt} \qquad ; k \in F \tag{3.16}$$

$$\Delta t \geq 0 \qquad ; k \in F \tag{3.17}$$

#### 3.2.3.5 Passenger Itinerary Disruption and Capacity Shortage Constraints

Constraints (3.8) and (3.9) ensure that binary variable $z_i$ takes value one if itinerary $i$ is disrupted. Constraint (3.18) states that passengers in disrupted itineraries are spilled.

$$sp_i^c \geq NP_i^c \times z_i \qquad ; i \in I, c \in C \tag{3.18}$$

Since we allow swapping aircraft, seat capacities of flights may change and passengers may also be spilled due to insufficient capacity. In capacity shortage cases, airline should decide on which passengers to spill considering different spill costs of different itineraries. For a given flight $k$, number of class $c$ passengers to be spilled due to capacity shortage is given by $max\{0, NPass_k^c - \sum_{t \in R} CAP_t^c \times y_{kt}\}$. These passengers should be selected from the itineraries that include flight $k$, and some of them may already been spilled due to itinerary disruptions. Constraint (3.19) ensures that number of passengers assigned to each flight do not exceed the seat capacities of its aircraft.

$$\sum_{i \in I(k)} sp_i^c \geq NPass_k^c - \sum_{t \in R} CAP_t^c \times y_{kt} \qquad ; k \in F \tag{3.19}$$

#### 3.2.3.6 Disruption Constraints

In aircraft and passenger recovery problem, we are dealing with delays of any severity. One can represent a disruption by entering departure delay ($DD_k$) or arrival delay ($AD_k$) for a flight. Constraint (3.20) states that a disrupted flight cannot depart before its ready time. Note that the ready time of these flights after disruption is equal to the sum of scheduled departure time and departure delay. Similarly, constraint (3.21) expresses arrival time disruptions.

$$dt_k \geq DT_k + DD_k \qquad ; k \in D \tag{3.20}$$

$$at_k \geq AT_k + AD_k \qquad ; k \in D \tag{3.21}$$

#### 3.2.3.7 Objective Function

The objective of the problem is to recover from the disruptions with minimum operating and inconvenience costs. Cost of recovery includes five terms: delay cost for aircraft and passengers, spill cost, swap cost, and increase in the fuel cost. Delay costs include flight and passenger related costs. Passenger related delay costs are based on the number of passengers in the original schedule. For each disrupted passenger, a spill cost is incurred. Both disrupted passenger and passenger delay costs depend on passenger classes. For pairs of flights that are swapped, a swap cost is incurred in order to return to the original schedule before the next day of operations. Cruise stage fuel cost of a flight considering both compression and swap decisions is expressed in (3.3). Using auxiliary variable $y_{kt}$, additional fuel cost of a particular flight, say $k$, is obtained by $\sum_{t \in R} f_{kt}(\Delta t_k) \times y_{kt} - FC_{kR_k}$. The objective function corresponding to the recovery cost is defined in (3.22) and the complete model is given below:

Minimize

$$\sum_{k \in F} delay_k \times \left( FDC_k + \sum_{c \in C} NPass_k^c \times PDC_k^c \right) +$$
$$\sum_{c \in C} \sum_{i \in I} SC_i^c \times sp_i^c +$$
$$\sum_{(k,m) \in S} Y_{R_k R_m} \times x_{km} +$$
$$\left( \sum_{k \in F} \sum_{t \in R} f_{kt}(\Delta t_k) \times y_{kt} - FC_{kR_k} \right) \tag{3.22}$$

subject to

| | |
|---|---|
| Swap constraints | (3.5) – (3.7) |
| Departure time constraints | (3.8) – (3.12) |
| Arrival time and delay constraints | (3.13) – (3.15) |
| Allowable compression constraints | (3.16) – (3.17) |
| Passenger itinerary disruption and capacity shortage constraints | (3.18) – (3.19) |
| Disruption constraints | (3.20) – (3.21) |
| $z_i \in \{0, 1\}$ | $i \in I$ |
| $x_{km} \in \{0, 1\}$ | $(k, m) \in S$ |

Binary decisions included in the proposed model are whether to maintain a passenger connection or not, and whether to swap aircraft of pairs of flights or not, as stated above. Note that APR is a MINLP model having a nonlinear objective function. Fuel cost term of the objective function includes products of convex functions with binary variables. In Section 3.3, we linearize the objective function and then show that the resulting model is SOCP-representable. Finally, we explain generation of conic quadratic constraints so that the model can be solved with commercial CQMIP solvers.

## 3.3 Conic Quadratic Mixed Integer Programming Model

In the objective function given in section 3.2.3.7 for each flight-aircraft pair there exists a nonlinear fuel cost term

$$f_{kt}(\Delta t_k) \times y_{kt} \tag{3.23}$$

which is the product of fuel cost function ($f_{kt}$) and flight-aircraft assignment variable ($y_{kt}$). This term is nonlinear and nonconvex. In the following, we will first move this function to the constraint set, and then show that the resulting constraint can be reformulated in a way to represent a convex set. We next observe that the resulting convex set can be represented using conic quadratic inequalities.

Substituting the function (3.3) in (3.23) we get

$$\Omega_{kt} \times \left(\frac{1}{T_{kt} - \Delta t_k}\right)^{K_{kt}} \times y_{kt} \tag{3.24}$$

where $\Omega_{kt} = FC_{kt} \times T_{kt}^{K_{kt}}$.

We next introduce an auxiliary variable $crt_k$ that represents the actual cruise time of flight $k$ as below:

$$crt_k = T_{kt} - \Delta t_k \quad k \in F \text{ and } t \in R \tag{3.25}$$

The next step in representing (3.23) via conic quadratic inequalities is to replace each term (3.23) in the objective function with the expression $\Omega_{kt} \times q_{kt}$ where $q_{kt}$ is an auxiliary variable and add the following inequality into the constraint set

$$\frac{y_{kt}}{crt_k^{K_{kt}}} \leq q_{kt} \tag{3.26}$$

This reformulation is obviously equivalent to the original one since it is a minimization problem. In its new form the objective is linear but now we have nonlinear constraints in the constraint set.

Next, we first reformulate (3.26) so that the reformulation will represent the hypograph of the geometric mean of $2^l$ nonnegative variables, which is a convex set. In the following, we drop $k, t$ indices of variables and consider $K = k_1/k_2$ where $k_1$, $k_2$ are integers.

**Proposition 3.3.1.** *Inequality* (3.26)*,*

$$\frac{y}{crt^K} \leq q$$

*can be equivalently written as*

$$y^{2^l} \leq q^{k_2} \times crt^{k_1} \tag{3.27}$$

*where* $l = \lceil \log_2(k_1 + k_2) \rceil$.

*Proof.* Inequality

$$\frac{y}{crt^K} \leq q$$

can be first written as

$$y \leq crt^{k_1/k_2} \times q$$

taking $k_2^{th}$ power of both sides we get

$$y^{k_2} \leq crt^{k_1} \times q^{k_2}$$

Exploiting the fact that $y$ is a 0-1 decision variable in the model, the exponent of $y$ can be increased and set to $2^l$ and we get:

$$y^{2^l} \leq crt^{k_1} \times q^{k_2} \times 1^{(2^l - k_1 - k_2)} \tag{3.28}$$

An inequality of the form $r \leq (s_1 s_2 \cdots s_{2^l})^{1/2^l}$ with $s_i \geq 0$ restrictions defines the hypograph of the geometric mean of variables $s_1, \ldots, s_2$ which is a convex set. Inequality (3.28) is of the same form with some restrictions on the variables. $k_1$ of them are identical and equal to $crt_k$, $k_2$ of them equal to $q$ and $(2^l - k_1 - k_2)$ of them equal to 1. $\square$

Proposition 3.3.1 shows that inequality (3.26) can be reformulated as an inequality which defines a convex set, namely a hypograph of geometric mean of $2^l$ variables. This leads to the following result.

**Proposition 3.3.2.** *Inequality* (3.26)*,*

$$\frac{y}{crt_k^K} \leq q$$

*with restrictions* $y \in \{0, 1\}, crt_k \geq 0, q \geq 0$*, can be represented using conic quadratic inequalities.*

*Proof.* As given in [17], for a positive integer $l$, an inequality of the form

$$r^{2^l} \leq s_1 s_2 \cdots s_{2^l}, \tag{3.29}$$

for $r, s_1, \ldots, s_{2^l} \geq 0$, i.e. a hypograph of geometric mean of $2^l$ variables, can be expressed equivalently using $O(2^l)$ variables and $O(2^l)$ hyperbolic inequalities of the form

$$u^2 \leq v_1 v_2,\ u, v_1, v_2 \geq 0 \tag{3.30}$$

Furthermore, each constraint $u^2 \leq v_1 v_2$ can be written as a conic quadratic constraint

$$\|(2u, v_1 - v_2)\| \leq v_1 + v_2. \tag{3.31}$$

□

This reformulation enables us to model the MINLP problem initiated in Section 3.2 as a CQMIP problem. The modified model with linear objective function, linear and conic quadratic constraints can be handled by fast algorithms of commercial CQMIP solvers. We finally illustrate the generation of conic quadratic constraints with an example.

**Example 3.3.1.** *Suppose that $K = 2.5$; and hence, $k_1 = 5$ and $k_2 = 2$ for a particular flight-aircraft pairing. By Proposition 3.3.1, inequality*

$$\frac{y}{crt^{2.5}} \leq q$$

*is first expressed as*

$$y^2 \leq crt^5 q^2$$

*Then, reformulated as*

$$y^8 \leq crt^5 q^2 1^1$$

*Obtained inequality can be expressed with the following three inequalities introducing two nonnegative variables:*

$$\begin{aligned} w_1^2 \leq& crt \times 1 \\ w_2^2 \leq& w_1 \times q \\ y^2 \leq& w_2 \times crt \end{aligned}$$

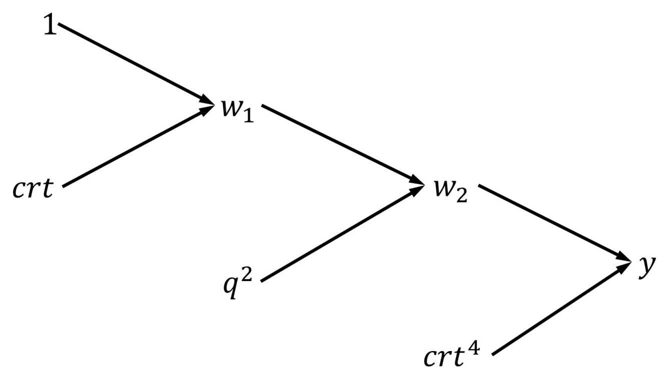


Figure 3.8: Illustration of generation of conic quadratic constraints

*Generation of the inequalities is illustrated in Figure 3.8.*

*These constraints can be represented by the following conic quadratic inequalities:*

$$\begin{aligned} 4w_1^2+(crt-1)^2 &\leq (crt+1)^2 \\ 4w_2^2+(w_1-q)^2 &\leq (w_1+q)^2 \\ 4y^2+(w_2-crt)^2 &\leq (w_2+crt)^2 \end{aligned}$$

In the next section, we discuss the computational performance of proposed reformulation.

### 3.4 Computational Results

In order to validate the practicality of the proposed approach for airline and passenger recovery operations, and investigate the effects of several factors on solution quality and performance of the approach with respect to solution time, we have set up an experimental design. Bureau of Transportation Statistics (BTS) provides airline on-time performance data (http://www.transtats.bts.gov/ DataIndex.asp). We have extracted the aircraft schedules of a major U.S. airline for three days of operations in April, 2011. Extracted data includes tail numbers, departure and arrival times, flight and cruise times, origins and destinations, and distances of flights. For each aircraft, we randomly selected the number of seats and values of fuel cost parameters. Each flight is included as a single-flight itinerary. Moreover, pair of flights that arrives to a common airport and satisfies minimum passenger connection time are considered as possible passenger connections. The second rule on these pairs to be connecting flights is that the origin of the prior flight should be a different airport than the destination of the latter one. These possible connecting flights are used to generate two-flight passenger itineraries. In a similar manner, again considering the minimum connection times, and origins and destinations of the flights, three-flight itineraries are constructed. Number of passengers in each flight is assigned randomly, considering the seat capacities and generated itineraries using these flights. Six different aircraft seat capacities are used: 150, 160, 180, 200, 260 and 300, where about 10% of the seats are assigned to business class passengers. Finally, the number of passengers in a flight is allocated randomly to the passenger itineraries that include the flight in consideration. BTS also provides departure delays. Actually disrupted flights are rated with respect to the severity of disturbation on the operations (length of delay, number of succeeding flights in the routing of the disrupted flight, and number of connected flights) and the most severe disruptions are selected for recovery instances.

We have analyzed five factors: number of hubs, number of disruptions, passenger delay cost, fuel cost and swap cost. Number of hubs is directly related to the size of the network in consideration and may affect the performance of the approach. Number of disruptions is another factor affecting the complexity of the problem; and hence, should be analyzed. Instances are created having one, two and five disruptions at a given time frame. Two levels of each of the three cost parameters (passenger delay cost, fuel cost and swap cost parameters) are used in the experimentation. Passenger delay cost is set to \$1.09 and \$1.5 per minute per passenger for the low and high values, respectively as proposed by [59]. As the cost estimates proposed by [20] suggest, we have doubled the delay costs for business class passengers. We have included a small flight delay cost of \$10 to prevent unnecessary delays. For low and high values of swap cost, we have used \$500 (proposed by [59]) and \$1000, respectively. Low and high fuel cost constants are generated from U(0.5, 1) and U(1.5, 3), respectively, where U($a$, $b$) is a uniform distribution in interval ($a$, $b$). Fuel cost exponents vary from 1.5 to 3.5. Spill costs are set to \$457.8 that is evaluated by the method proposed by [59]. For high and low levels of the factors, + and - signs will be used. 15 instances are created for each of the 72 experimental settings. Therefore, a total of 1080 instances are solved using version 12.1 of IBM ILOG CPLEX. In order to test the practicality of our approach, solution times are limited to 300 seconds.

Generated instances include one, two or four hubs. Sizes of the corresponding networks are displayed in Table 3.5. Number of routings is obtained from the data by counting the number of routings that visit the selected hubs. All flights in these routings and all airports visited are relevant to our disruption problem; and hence, are included in the model.

Solution times and gaps are summarized in Table 3.6. Gaps are calculated by dividing the difference between the best integer solution and best lower bound to the best integer solution. When we increase the number of hubs or disruptions, the corresponding solution times and gaps increase as well. We deal with realistic size problem instances, and still solve 1059 of 1080 test instances to optimality using the proposed APR approach. Solution times are greater than one minute only in four-hub instances; however, average solution time is still less than one minute in general. When we analyze the existing schedules, we observed that the number of disruptions was usually less than two in a given time frame. For this particular case, we could solve all of the problem instances optimally in less than one minute of computation time. This is a very important contribution since we can now use optimization techniques instead of relying on ad hoc approaches for such a critical problem.

Table 3.5: Effect of number of hubs on the size of the network

| # of hubs | Average # of flights | Average # of routings | Average # of airports |
|---|---|---|---|
| 1 | 352 | 99 | 41 |
| 2 | 619 | 211 | 49 |
| 4 | 1429 | 419 | 76 |

Table 3.7 summarizes characteristics of the recovery plans generated by APR with respect to the levels of cost factors (+ and - signs indicate high and low values, respectively). Magnitude of delay cost has the greatest effect on the recovery plan. It can be seen from the table that in the first four settings, number of delayed flights and total delay are less than those for the last four settings permitting disruptions in more passenger itineraries. APR tries to minimize delay propagation as much as possible

Table 3.6: Effect of # of hubs and # of disruptions on solution performance of APR

| # of hubs | # of disruptions | Average CPU time (sec.) | Average gap (%) | Percentage of optimal solutions |
|---|---|---|---|---|
| 1 | 1 | 4 | - | 100 |
| | 2 | 7 | - | 100 |
| | 5 | 23 | - | 100 |
| 2 | 1 | 6 | - | 100 |
| | 2 | 10 | - | 100 |
| | 5 | 55 | 1.7 | 97 |
| 4 | 1 | 44 | - | 100 |
| | 2 | 72 | - | 100 |
| | 5 | 142 | 3.8 | 82 |

swapping more flights and using more cruise time compression when delay cost is set to its high level. Therefore, we do not observe a statistically significant effect of the other cost parameters in these settings. When we consider the last four settings separately, we can observe a small effect of fuel cost on the amount of total compression. We do not observe a similar effect for swap cost; however, we can state that swap opportunities help delay mitigation as the number of swaps is proportional to total delay.

Table 3.7: Effects of cost factors on recovery actions with APR

| Delay cost | Fuel cost | Swap cost | # of flights with delay | Delay (min.) | # of spills | # of swaps | Compr. (min) |
|---|---|---|---|---|---|---|---|
| + | + | + | 4.07 | 726 | 57.2 | 1.88 | 165 |
| + | + | - | 4.02 | 731 | 38.0 | 1.98 | 228 |
| + | - | + | 4.00 | 718 | 41.1 | 1.92 | 179 |
| + | - | - | 3.89 | 710 | 39.2 | 1.84 | 187 |
| - | + | + | 4.22 | 764 | 43.8 | 1.42 | 148 |
| - | + | - | 4.10 | 747 | 38.4 | 1.46 | 153 |
| - | - | + | 4.24 | 748 | 39.6 | 1.54 | 168 |
| - | - | - | 4.11 | 734 | 30.6 | 1.78 | 171 |

Trying to find the optimal balance between five cost terms, APR is expected to outperform methods PB1 and PB2; however, test instances are also solved with these push-back recovery policies in order to observe the improvement in total recovery costs.

Table 3.8: Comparison of APR with other approaches

| # of hubs | # of disruptions | PB1 (%) | PB2 (%) | APR2 (%) |
|---|---|---|---|---|
| 1 | 1 | 28.1 | 28.1 | 6.3 |
| | 2 | 30.0 | 203.2 | 7.9 |
| | 5 | 18.5 | 135.1 | 5.2 |
| 2 | 1 | 29.2 | 29.5 | 6.5 |
| | 2 | 42.4 | 129.2 | 15.7 |
| | 5 | 18.2 | 180.5 | 4.3 |
| 4 | 1 | 31.0 | 31.2 | 9.4 |
| | 2 | 42.0 | 130.6 | 6.8 |
| | 5 | 29.8 | 170.9 | 19.4 |
| Average | | 28.3 | 130.6 | 8.4 |

Moreover, in our experiments, we have observed that complexity of the problem is greatly depending on the number of swap opportunities. Without swap decision, all test instances are solved to optimality in less than a minute (about 15 seconds on the average). Therefore, we have named this alternative solution procedure as APR2 and included in our analysis. On the other hand, PB1 and PB2 plans are created within a second. Table 3.8 summarizes the percentage cost deviations of alternative approaches with respect to APR solutions. Deviations are calculated by dividing the cost difference to the recovery cost obtained by APR. Greatly propagating delays, PB2 results in most costly recovery plans. PB1 performs better than PB2, but the cost deviation from APR solution is 28.3%. We also observe that APR2 provides great cost improvements compared to push-back recovery policies. On the other hand, 8.4% deviation indicates the importance of swap opportunities. It is important to note that even in instances where APR stopped with positive gaps, it provided better solutions than APR2. This finding suggests that swap opportunities should be evaluated even they make the problem more complex. Moreover, including the cruise speed control into the integrated recovery process greatly enhances the solution quality as seen in APR2 compared to PB1 or PB2.

Since APR2 finds the optimal solution in very short solution times compared to APR, it may be preferred to APR in scenarios with less available solution times. In order to compare solution qualities of these two approaches within shorter solution times, we have experimented same instances with time limits of one and two minutes. Table 3.9 summarizes the percentage cost deviations of APR2 with respect to APR within 60 and 120 seconds. Considering overall deviations, we can state that APR still outperforms APR2. However, in five disruptions scenario of two-hub case and more than one disruption scenarios of four-hub cases, the deviations are negative when we need to take recovery actions within one minute. The results suggest to use APR2 with these settings. On the other hand, we observe that 120 seconds is sufficient to evaluate and utilize swap opportunities, since APR outperforms APR2 in all settings within two minutes.

Table 3.9: Comparison of APR with APR2 within shorter solution time limits

| # of hubs | # of disruptions | 60 seconds | 120 seconds |
|---|---|---|---|
| 1 | 1 | 6.3 | 6.3 |
| | 2 | 7.9 | 7.9 |
| | 5 | 5.2 | 5.2 |
| 2 | 1 | 6.5 | 6.5 |
| | 2 | 15.7 | 15.7 |
| | 5 | -1.4 | 4.2 |
| 4 | 1 | 5.3 | 9.4 |
| | 2 | -4.9 | 6.8 |
| | 5 | -6.2 | 4.8 |
| Average | | 2.1 | 6.9 |

## 3.5 Conclusions

We develop a mathematical model for passenger and aircraft recovery problem. Main focus of the study in this chapter is to integrate cruise speed control with other recovery actions such as retiming departure and arrival times, and swapping aircraft. Airlines generally operate their flights at cruise speeds that result in minimum fuel consumptions, which is lower than the maximum speed that the aircraft can reach. Our experiments have shown that cruise time compression with cruise speed control can greatly mitigate delays. On the other hand, fuel consumption increases as the aircraft speed up. In accordance with the airline manufacturers' technical specifications, we present a convex and increasing function to express the change in fuel cost as the speed increases. Proposed formulation of the problem is originally a mixed integer nonlinear programming (MINLP) model. We first linearize the nonlinear cost term in the objective function and then show that the resulting problem is second-order cone programming (SOCP) representable. Finally, we create conic quadratic constraints for the nonlinear constraints to be able to solve the problem with commercial MIP solvers such as IBM ILOG CPLEX. We also place special emphasis on passenger-related costs. In addition to itineraries, we also discretize passengers with respect to their classes. Proposed model decides passengers in which itineraries will be delayed and how much; and which passenger itineraries will be disrupted, if necessary. Objective of our model is to minimize total recovery cost in case of disruptions. Recovery cost consists of five terms: flight delay cost, passenger delay cost, disrupted itinerary cost, swap cost, and increased fuel cost. Proposed model is able to create minimum cost recovery plans by finding the optimal tradeoff between these cost terms. We have performed an extensive computational study for five factors, i.e., number of hubs, number of disruptions, delay cost, fuel cost, and swap cost. Number of hubs determines the size of the network under consideration; and hence, is significant on solution times and gaps. Number of disruptions that will be handled is also one of the most important factors affecting the complexity of the problem. Due to the nature of the problem we expect to handle one or two disruptions at a given time frame. Our computational experiments have shown that proposed approach is able to handle two disruptions on a four-hub network of a major U.S. airline within less than one minute on the average. Moreover, 97% of the instances including problems dealing with five disruptions at a given time frame are solved to optimality in real time. In the solution approach, we only allow swapping aircraft that satisfy crew and maintenance related constraints, although this assumption can be relaxed to enlarge the solution space.

In the next chapter, we manage to integrate aircraft, crew and passenger recovery by an alternative problem representation. Proposed solution space includes all recovery actions for each entity type. Furthermore, we propose realistic passenger delay cost calculations in order to evaluate recovery actions more accurately.

# CHAPTER 4

# A NETWORK FLOW APPROACH FOR INTEGRATED AIRLINE RECOVERY WITH CRUISE SPEED CONTROL

In this chapter, we try to achieve full integration of airline recovery problems. Recall that integrated airline recovery approaches suffer from huge problem sizes and intractability within real time. Therefore, we initially focus on an alternative problem representation. In Section 4.1, we propose an alternative connection network (CN) representation that is advantageous in both size of problem representation, and ease of integrating different entity types and recovery actions. Proposed representation is illustrated on an example recovery problem. In Section 4.2, we propose a network flow based formulation that integrates recovery actions, restrictions and disruption costs of aircraft, crew members and passengers. Furthermore, problem representation and formulation allows to integrate any other entity type (such as luggage) that is important for the decision maker in the same manner. We propose four different passenger delay cost calculation methods. These include approximation methods that can achieve faster solutions and more realistic formulations. Since all possible recovery actions are included in the solution space, proposed formulation contains a nonlinear cost term in the objective function due to the nonlinear tradeoff between cruise speed and fuel cost of flights. In Section 4.3, we propose a similar scheme as discussed in Section 3.3 to reformulate the proposed MINLP model as a CQMIP model. Integrated airline recovery formulation with actual passenger delay cost calculations and cruise speed control action is very complex. In Section 4.4, we present two important preprocessing methods to reduce the problem size and complexity without sacrificing optimality. We test the efficiency of our solution approach with practical-sized problems and major disruption types. Results of the experimentations are presented in 4.5. Final remarks on the integrated airline recovery approach are given in 4.6.

## 4.1 Problem Representation

In this section, we give the problem definition and briefly review two important network representations discussed in Section 2.3. Afterwards, we present the proposed network structure and redefine the problem on this network.

An original schedule of an airline is given. A set of disruptions occur on the schedule. We consider a recovery horizon, $[t_0, t_1]$. The aim of *integrated airline recovery problem* is to find the minimum-cost recovery actions by altering operations of aircraft, crew members and passengers within the recovery horizon while providing that the original schedule will be caught up by $t_1$ the latest.

An effective representation of disruption management problem is crucial due to the size of flight networks, complexity of the problem and limited solution times. There are two important representations in the literature. [62] utilize flight strings which are sequences of flights with timing decisions. Same sequence of flights might be present in multiple strings each with a different set of retiming decisions. In order to associate rerouting options, eligible flight strings with different flights and retiming decisions for each entity need to be generated. String-based models have the advantage of ignoring ground time requirements in the formulation since each flight string already satisfies this restriction. On the other hand, since feasibility of flight strings are evaluated by constant flight times, it is not easy to incorporate cruise speed control.

The second important and widely used approach uses a time-space network representation ([20], [59]) where nodes are associated with both time and location. Flights are represented by arcs between two nodes belonging to different locations. In order to satisfy the ground time between any two consecutive flights, ground arcs starting and ending at the same location are included. Departure time decisions are evaluated by creating flight copies at different departure time alternatives. [59] manage to incorporate cruise speed control, or flight planning, with time-space network by generating a second set of flight copies at each departure time alternative of each flight where each copy corresponds to a different cruise speed alternative. However, this requires discretization of cruise speed options and a huge network to be generated.

We propose an alternative network representation which may be classified as a connection network (CN), also known as a flight network or an activity-on-node network. CNs represent the problem with a much smaller number of nodes and arcs since each scheduled flight is represented by a single node. Moreover, they provide a greater flexibility in retiming decisions and they are able to generate all possible paths without enumeration. [70] propose a CN representation for integrated schedule design, fleet assignment and aircraft routing problem in which aircraft are transported through the CN starting and ending at a station. The authors describe the advantages of CN representation in detail.

We start our approach by defining state parameters that capture the true state of any entity. These definitions allow modeling all entity types (aircraft, crew member, or a passenger) in a similar manner. Then, we propose a general CN representation that allows to integrate any entity type. Therefore, not only aircraft, but all entities are transported through a CN. By integration on a common CN, interdependencies among different entity types are easily defined. Moreover, all recovery actions including cruise speed control are included in the model to ensure optimality. Since activity is kept on nodes, departure time, arrival time and cruise speed decisions can be represented by continuous variables instead of a set of discrete alternatives.

### 4.1.1 Network Structure

We start with the notation required to understand the network structure. For the ease of reading, we use overscores and underscores to denote parameters as upper and lower bounds, respectively. All parameters begin with an upper case letter while decision variables start with a lower case letter throughout the text. Parameters of scheduled flights are defined below.

| | | |
|---|---|---|
| $Ori_f(Des_f)$ | : | Origin (destination) airport of flight $f$ |
| $SDT_f(SAT_f)$ | : | Scheduled departure (arrival) time of flight $f$ in the original schedule |

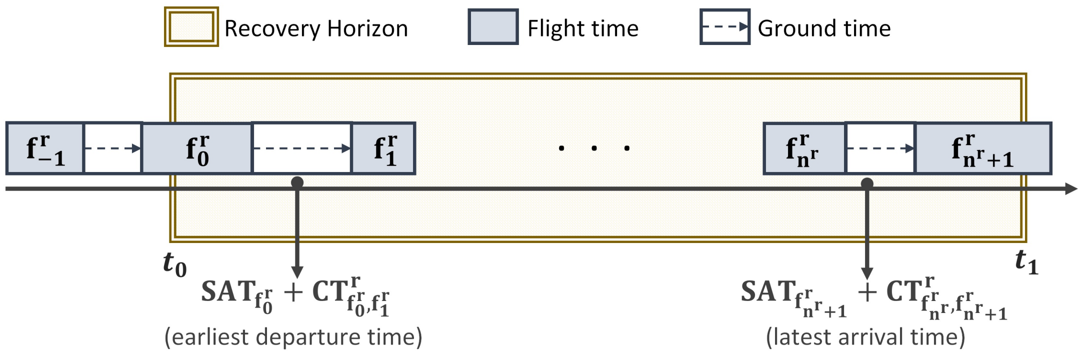


Figure 4.1: Earliest departure time and latest arrival time of an entity

| | | |
|---|---|---|
| $\overline{DT_f}(\overline{AT_f})$ | : | Latest allowable departure (arrival) time of flight $f$ |
| $\overline{\Delta T_f}$ | : | Maximum of cruise time compression limits for flight $f$ among all aircraft types |
| $\underline{FT_f}$ | : | Minimum of flight times of flight $f$ at max-range cruise speed among all aircraft types |
| $\underline{AT_f}$ | : | Earliest possible arrival time of flight $f$ |
| $\underline{CT_{fg}}$ | : | Minimum of minimum connection times among all entities between flights $f$ and $g$ |

An airline may have different types of aircraft in its fleet. As a recovery action, aircraft swaps may occur between flights. Therefore, while obtaining the values of $\overline{\Delta T_f}$ and $\underline{FT_f}$, we consider all possible aircraft types that can operate the flight. Maximum-range cruise speed is the speed of an aircraft that results in minimum fuel consumption which will be discussed in detail in Section 4.3. In order to generate all possible rerouting options, we set the value of $\underline{CT_{fg}}$ to the minimum of required connection times among all entities. Note that there are two limitations on the earliest arrival time of a flight. The first one is determined by time slot availability. On the other hand, a flight cannot arrive before $SDT_f + \underline{FT_f} - \overline{\Delta T_f}$. Therefore, $\underline{AT_f}$ is set to the maximum of these limitations.

#### 4.1.1.1 Nodes

Proposed network contains four types of nodes: scheduled flight nodes, source nodes, sink nodes and must-visit-nodes (or must-nodes). For each entity there is a source node which represent the initial state of the entity at $t_0$ and a sink node which represent the final status of entity at $t_1$. For each entity, there might be certain must-nodes. A must-node might represent a maintenance activity of an aircraft at a certain airport at a certain time period. Each node has a demand parameter corresponding to each entity.

Let $\mathcal{F}$ be the set of all scheduled flights of the airline. Then, the set of **flight nodes**, $F$, relevant to the problem are obtained as follows:

$$F = \{f \in \mathcal{F} : SDT_f \geq t_0 \text{ and } \underline{AT_f} \leq t_1\}$$

which defines all flights scheduled to depart after $t_0$ and scheduled to arrive before $t_1$ as illustrated in Figure 4.1. Let $T$ be the set of entity types (aircraft, crew member or passenger) relevant to our problem, $r \in R^t$ be an entity of type $t$ and $R = \bigcup_{t \in T} R^t$ be the set of all entities. Demand at flight nodes for all entities are zero, i.e. $D_f^r = 0, \forall r, f \in F$.

Dynamic state of an entity is obtained and defined by the parameters below. Earliest departure time and latest arrival time parameters that guarantee that operations outside the recovery horizon will be operated as scheduled are illustrated in Figure 4.1.

$\mathcal{F}^r$ : Set of scheduled flights originally assigned to entity $r$ ordered by departure times

$\mathcal{F}^r = \{, ..., f_{-1}^r, f_0^r, f_1^r, ..., f_{n^r}^r, f_{n^r+1}^r, ...\}$

$F^r$ : Set of scheduled flights of entity $r$ within the recovery horizon ordered by departure times

$F^r = \{f \in \mathcal{F}^r : SDT_f \geq t_0, \underline{AT_f} \leq t_1\} = \{f_1^r, f_2^r, ..., f_{n^r}^r\}$

$FT_f^r$ : Flight time of flight $f$ when operated by aircraft $r$ at max-range cruise speed

$CT_{fg}^r$ : Minimum connection time required for entity $r$ between flights $f$ and $g$

$Ori^r$ : Location of entity $r$ at the beginning of the recovery horizon
(e.g., $Ori^r = Ori^r_{f_1}$)

$\underline{DT^r}$ : Earliest time that the first flight of entity $r$ can depart (ready time)
$\underline{DT^r} = max\{t_0, SAT_{f_0^r} + CT^r_{f_0^r f_1^r}\}$

$Des^r$ : Planned destination of entity $r$ at the end of the recovery horizon
(e.g., $Des^r = Des_{f^r_{n^r}}$)

$\overline{AT^r}$ : Latest time that entity $r$ needs to arrive at $Des^r$ to catch up its schedule
$\overline{AT^r} = min\{t_1, SDT_{f^r_{n^r+1}} - CT^r_{f^r_{n^r} f^r_{n^r+1}}\}$

Recovery actions such as reserve aircraft and standby crew can be included in the solution space by simply inserting these entities in set $R$ with corresponding entity parameters. These entities can be generalized as operating resources that can be used within the recovery horizon and have $F^r = \emptyset$.

**Source node** for entity $r$ is designated by $s^r$ and has the following parameters to represent the initial state of the entity:

$$Des_{s^r} = Ori^r, \quad \underline{AT_{s^r}} = \underline{DT^r}, \quad CT^r_{s^r,g} = 0, \forall g \in F, \quad D^r_{s^r} = -1.$$

**Sink node** for entity $r$ is designated by $t^r$ and has the following parameters to represent the final status of the entity:

$$Ori_{t^r} = Des^r, \quad \overline{DT_{t^r}} = \overline{AT^r}, \quad CT^r_{f,t^r} = 0, \forall f \in F, \quad D^r_{t^r} = +1.$$

Finally, we insert **must-nodes** to model the restrictions of entities within the recovery horizon such as scheduled aircraft maintenances or away-from-home limitations of crew members. In the proposed solution approach, we will force entities with such restrictions to visit these nodes. Let $M^r$ be the set of must-nodes of entity $r$, and $M = \bigcup_{r \in R} M^r$. For each must-node of entity $r$, we have:

$Ori_m = Des_m$ : location of the activity

$\overline{DT_m}(\underline{AT_m})$ : scheduled start (completion) time of the activity

$$CT^r_{fm} = CT^r_{mg} = 0, f, g \in F, \quad D^r_m = 0.$$

Then, the set of nodes of the network is $\mathcal{N} = F \cup \left(\bigcup_{r \in R}\{s^r, t^r\}\right) \cup M$.

#### 4.1.1.2 Arcs

An arc $(f, g)$ may correspond to a flight connection (if $f, g \in F$), beginning of the operations of an entity (if $f = s^r$), end of the operations of an entity (if $g = t^r$), or connections with must-nodes (if $f$ or $g \in M^r$). Set of arcs is obtained using node parameters as follows:

$$A = \{(f, g) : f, g \in \mathcal{N}, Des_f = Ori_g \text{ and } \overline{DT_g} \geq \underline{AT_f} + \underline{CT_{fg}}\}.$$

This rule allows to include all possible connections considering the allowed flexibility in departure and arrival times by time slots and by cruise speed options. Therefore, all possible paths can be generated through the proposed network. Moreover,

In order to incorporate recovery actions such as ferrying aircraft or deadheading crew members, we insert **external arcs**, i.e. $(f, g) \notin A$, whose arc costs are equal to the costs of the corresponding actions. An external arc from $s^r$ to $t^r$ may represent ferrying the aircraft (deadheading the crew member) from its origin to its destination. An aircraft can also be ferried after operating some flights. This action can be modeled by an external arc from a flight node to the sink. In terms of passengers, cancelling the ticket of a passenger or reallocating to other means of transportation is modeled by an external arc from source to the sink. On the other hand, a passenger with two or more scheduled flights may be spilled after its first/second flight which may be modeled by an external arc from a flight node to the sink. In general, any path that is external to the CN of the airline can be modeled by external arcs. Therefore, cooperation between airline companies in terms of passenger recovery can also be modeled using external arcs. Let $E^r$ be the set of external arcs available for entity $r$ and $E = \bigcup_{\forall r} E^r$. Then, the set of arcs of the proposed network is $\mathcal{A} = A \cup E$.

Proposed network structure $G = (\mathcal{N}, \mathcal{A})$ is illustrated in Figure 4.2. Source and

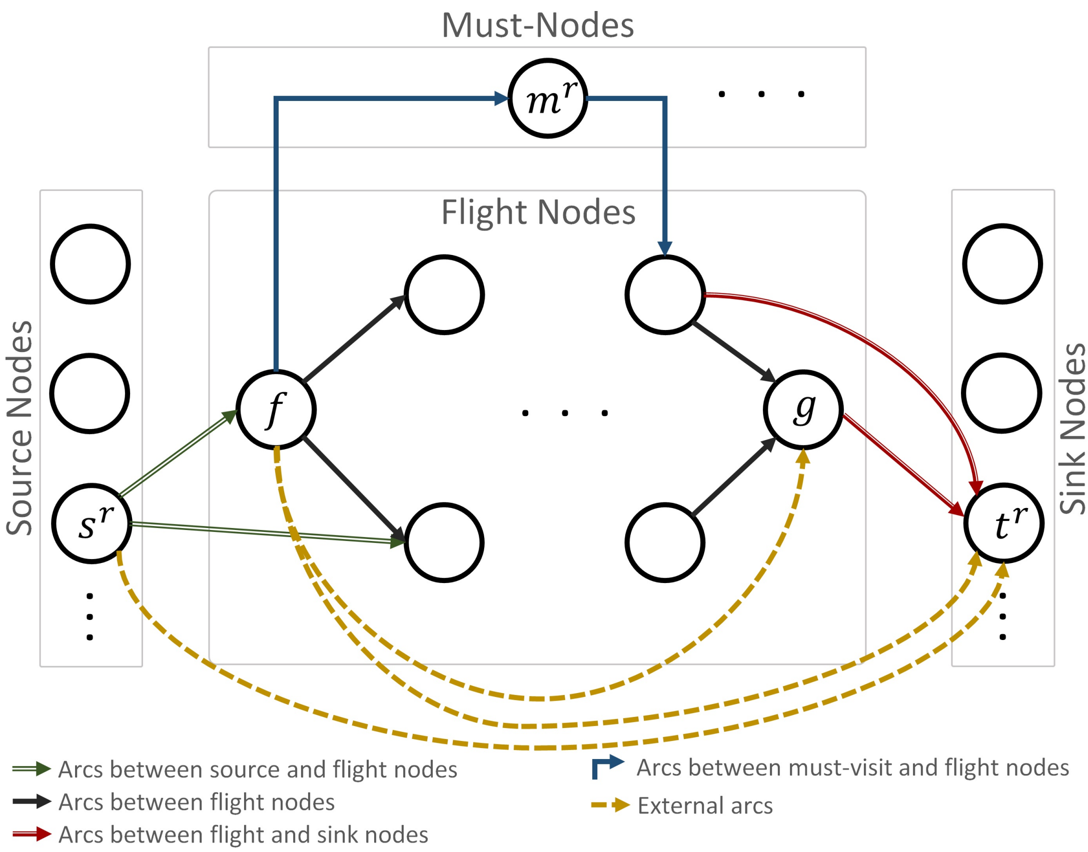


Figure 4.2: Network structure of the proposed representation.

sink nodes are displayed on the left and right of the network, respectively. The arcs emanating from source nodes (incoming to sink nodes) represent the connection to the first (from the last) flight for the particular entity. For entities with restrictions, we have a set of must-nodes displayed at the top of the network. The nodes within the box in the middle of the network correspond to scheduled flights with incoming and emanating flight connection arcs. All connections are created with respect to the arc generation rule given above. Finally, three external arcs are displayed at the bottom of the network (dashed) which may correspond to different recovery actions. We have -1 (+1) demand in the source (sink) nodes for the corresponding entities, while all flight and must-visit nodes have zero demand.

### 4.1.2 Disruption Types

All disruptions are modeled by updating parameters of entities and specific parts of the network, i.e. no constraints need to be added to the formulation. We have selected and experimented four disruption types which are of major importance with respect to their frequency or severity. After describing how these disruptions are represented, we redefine the problem with the proposed network structure.

#### 4.1.2.1 Flight Departure Delay

Scheduled departure time of a flight may be delayed due to various reasons such as airport congestion or irregularities in ground operations. These disruptions are represented by updating $SDT_f$ as $SDT_f + DD_f$, if flight $f$ experiences a departure delay of $DD_f$ in minutes.

#### 4.1.2.2 Flight Cancellation

If a flight experiences a severe departure delay, the airline may have no other option but cancel the flight. Let $D^c$ be the set of canceled flights. Then, all nodes in $D^c$ are removed from the network together with all arcs incoming to and emanating from these nodes.

#### 4.1.2.3 Delayed Ready Time

Aircraft experiencing an unscheduled maintenance or late arrivals of crew members are examples of this type of disruptions. Note that considering these as flight departure delays would eliminate many feasible recovery options and lead to sub-optimal solutions. In particular, even if the ready time of an aircraft is delayed, its first flight could still be operated on-time by another available aircraft. These disruptions are modeled by updating $\underline{DT^r}$ as $\underline{DT^r} + RD^r$ if entity $r$ experiences a ready time delay of $RD^r$ in minutes.

#### 4.1.2.4 Airport Closure

Poor weather condition is one of the major reasons for an airport to cancel all departures and arrivals for a while. Let $D^{[ac]}$ be the set of closed airports and $a \in D^{[ac]}$ be an airport experiencing a closure during $[ST_a, ET_a]$. The consequences of this closure are handled in two parts. Firstly, due to the closure of airport $a$, some flights need to be canceled. We insert these flights into set $D^{[c]}$ with the following operations:

$$D^{[c]} = D^{[c]} \cup \{f : Ori_f = a, SDT_f > ST_a \text{ and } \overline{DT_f} < ET_a\}$$

$$D^{[c]} = D^{[c]} \cup \{f : Des_f = a, \underline{AT_f} > ST_a \text{ and } \overline{AT_f} < ET_a\}.$$

On the other hand, some flights affected from this closure may still be operated by rescheduling the departure times or increasing their cruise speeds. These flights are obtained in two subsets:

- $\{f : Ori_f = a, SDT_f < ST_a \text{ and } \overline{DT_f} > ST_a\}$, and
- $\{f : Des_f = a, \underline{AT_f} < ET_a \text{ and } \overline{AT_f} > ET_a\}$

The first subset includes flights which are scheduled to depart prior to the closure, however, departure times may be held beyond the starting time of closure as $\overline{DT_f} > ST_a$. Therefore, we update $\overline{DT_f} = ST_a$ to guarantee that these flights do not depart during closure. Ending time of closure falls within the arrival time slots of flights in the second subset. For these flights, we update $\underline{AT_f} = ET_a$ so that they do not arrive during closure.

Given the network representation, the aim of *disruption management problem* is to find the minimum-cost flow of entities from their source nodes to their sink nodes provided that must-visit nodes will be visited by the corresponding entities.

### 4.1.3 Numerical Example

We illustrate the problem representation on a small-sized numerical example. The flight schedule of an airline within the recovery horizon is tabulated in Table 4.1, and details of these flights are presented in Table 4.2. Three aircraft and four crew teams are involved in the problem. In this example, we assume that each flight is operated by a crew team; however, the proposed approach can handle different requirements. Scheduled flights of crew teams C1, C2, C3, and C4 are 1-2-3-4-9, 5, 6-7-8-13, and 10-11-12, respectively. The aircraft with tail numbers N322AA and N345AA have a seat capacity of 180, while the seat capacity of N5FCAA is set to 210.

Table 4.1: Original aircraft and crew schedules of the example

| Tail # | Flight # | Crew Id | From | To | SDT | SAT |
|---|---|---|---|---|---|---|
| N322AA | 1 | C1 | ORD | DCA | 5:30 | 7:10 |
| | 2 | C1 | DCA | ORD | 7:50 | 9:30 |
| | 3 | C1 | ORD | DFW | 10:00 | 12:20 |
| | 4 | C1 | DFW | ORD | 13:00 | 15:20 |
| | 5 | C2 | ORD | DCA | 16:30 | 18:10 |
| N345AA | 6 | C3 | LAX | ORD | 6:00 | 9:40 |
| | 7 | C3 | ORD | MSP | 12:00 | 13:10 |
| | 8 | C3 | MSP | ORD | 14:00 | 15:10 |
| | 9 | C1 | ORD | LAX | 16:00 | 19:40 |
| N5FCAA | 10 | C4 | DCA | ORD | 9:00 | 10:40 |
| | 11 | C4 | ORD | MSP | 11:10 | 12:20 |
| | 12 | C4 | MSP | ORD | 13:00 | 14:10 |
| | 13 | C3 | ORD | DCA | 16:00 | 17:40 |

The original routing of N322AA is 1-2-3-4-5. However, it may be rerouted through many alternative paths to reach DCA from ORD. For instance, it may only operate

Table 4.2: Details of flight schedule of the example

| Tail # | Flight # | Cruise Time | Distance | Number of Passengers |
|---|---|---|---|---|
| N322AA | 1 | 70 | 610 | 126 |
| | 2 | 70 | 610 | 149 |
| | 3 | 110 | 800 | 111 |
| | 4 | 110 | 800 | 166 |
| | 5 | 70 | 610 | 153 |
| N345AA | 6 | 190 | 1745 | 170 |
| | 7 | 40 | 335 | 172 |
| | 8 | 40 | 335 | 135 |
| | 9 | 190 | 1745 | 139 |
| N5FCAA | 10 | 70 | 610 | 170 |
| | 11 | 40 | 335 | 196 |
| | 12 | 40 | 335 | 200 |
| | 13 | 70 | 610 | 154 |

flight 1 in cases of severe disruptions, or follow the path 1-2-5 if flight 3 or 4 is cancelled. Moreover, the aircraft may use the flights scheduled to any other aircraft, i.e. it can follow the path 1-10-11-12-5. On the other hand, only a subset of flight nodes and connections can be used by this entity to construct a feasible path from its origin to its destination. For instance, flight 6 cannot be operated by N322AA since the aircraft is currently located at ORD and cannot arrive at LAX before the latest departure time of this flight. The part of the proposed network related to N322AA is given in Figure 4.3. This partial network is able to generate all possible paths for the particular entity with an additional external arc (dashed) corresponding to ferrying action. In Section 4.4.1, the importance of partial networks and an efficient algorithm to generate them will be described.

In Figure 4.4, partial network of crew team C3 is displayed. The original schedule of C3 which is transported from LAX to DCA is 6-7-8-13. All possible paths such as 6-7-12-13 or 6-13 can be generated through this network with an additional external arc for deadheading. Consider flight connection arc between flights 7 and 12 which is infeasible in the original schedule. Scheduled arrival time of flight 7 is 13:10 while scheduled departure time of flight 12 is 13:00. However, there exists a possibility to provide required connection time between these flights by speeding up flight 7 and holding the departure time of flight 12. Therefore, we include this connection in our

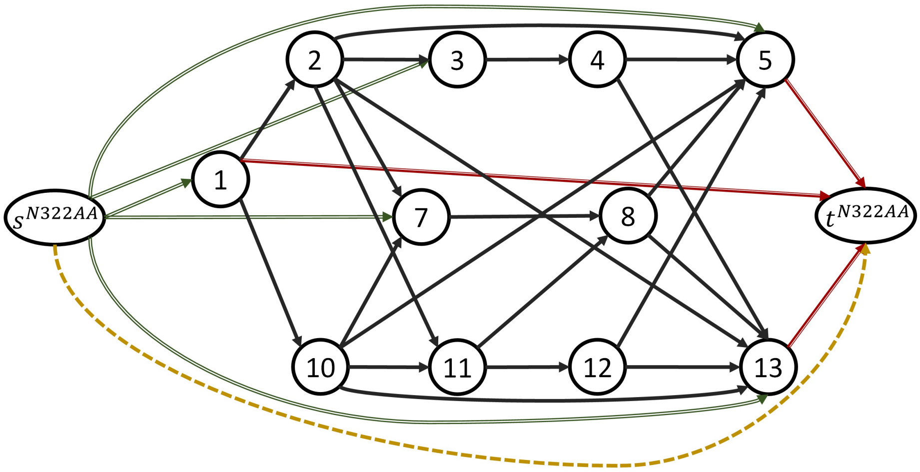


Figure 4.3: Partial network of aircraft N322AA.

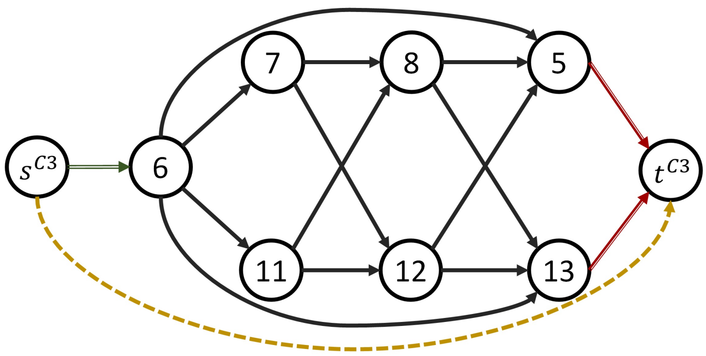


Figure 4.4: Partial network of crew team C3.

solution space as well.

In this example, there exists 13 single-flight itineraries corresponding to each scheduled flight, and six two-flight itineraries. All itineraries and number of assigned passengers are tabulated in Table 4.3. Partial network of a passenger in itinerary 2-7 is illustrated in Figure 4.5. The external arc from source to sink corresponds to spilling the passenger, while the other one from 2 to the sink corresponds to reallocating this passenger to other means of transportation at ORD.

In the disruption scenario, flight 1 experiences a departure delay of 90 minutes, i.e. it cannot depart before 7:00. This disruption is handled by updating the scheduled departure time of flight 1.

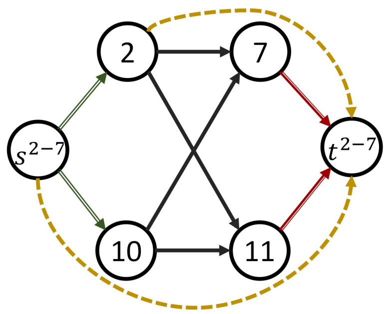


Figure 4.5: Partial network of passengers in itinerary 2-7.

Table 4.3: Numbers of passengers in passenger itineraries

| Itin. | # Pass'rs | Itin. | # Pass'rs | Itin. | # Pass'rs | Itin. | # Pass'rs |
|---|---|---|---|---|---|---|---|
| 1 | 126 | 4 | 166 | 7 | 67 | 10-11 | 91 |
| 2 | 51 | 5 | 88 | 8 | 70 | 11 | 105 |
| 2-3 | 53 | 6 | 55 | 8-5 | 65 | 12 | 200 |
| 2-7 | 45 | 6-7 | 60 | 9 | 139 | 13 | 99 |
| 3 | 58 | 6-13 | 55 | 10 | 79 | | |

Note that without rerouting options, 90 minutes delay in flight 1 would propagate through the downstream flights of aircraft N322AA and through those of crew team C1. In the optimal solution of this example, aircraft N322AA follows the path $s$-1-10-11-12-13-$t$ while the path of aircraft N5FCAA is $s$-2-3-4-5-$t$. Since N5FCAA is available in DCA at 7:50, flight 2 does not wait for arrival of the delayed flight. This swap action prevents the delay of flight 1 to propagate. Since destinations of both flights 13 and 5 are DCA, each aircraft is positioned at their expected locations by the end of recovery horizon.

Crew rerouting actions are more complicated in this example. Note that the crew team that is originally assigned to flight 2 also operates flight 1. Since, flight 2 does not wait for the arrival of flight 1, flight 2 is assigned to another crew team. In this example, we assume that each crew team is ready at the beginning of the recovery horizon and can operate each of the flights; however, such technical limitations can easily be inserted in the proposed approach. In the optimal solution, crew team C1, which is originally located at ORD and needs to arrive at LAX, operates only flight 9. Crew team C2 operates flights 1-10-7-8-13 and reaches its destination (DCA). Flights 6-11-12-5 are operated by C3 with an origin-destination pair LAX-DCA. Finally, flights 2-3-4 are operated by C4. Note that C4 is available in DCA at 7:50, and therefore, flight 2 is not delayed. Also notice that flight 10 can still be delayed since flight 1 is experiencing a departure delay and C2 is transported through the arc 1-10.

Allowing inter-fleet reassignments has two consequences. Firstly, the speed capabilities of different aircraft may be different and this affects the maximum amount of compression of flights, consequently additional fuel and carbon emission costs incurred due to the speed increases. In this example, we have assumed that each aircraft have similar speed capabilities. In the optimal solution, flight 1 is compressed by 7 minutes for both decreasing the arrival delay of this flight and preventing propagation through the connection 1-10. With the given departure delay and 7 minutes of compression, flight 1 departs at 7:00 and arrives at 8:33. In this example, crew connection times are set to 30 minutes. Then, due to connection 1-10 used by C2, flight 10 with a scheduled departure time of 9:00 departs at 9:03. In the optimal solution, the speed of this flight is also increased so that it arrives on time at 10:40.

Secondly, the seat capacities of aircraft may be different and inter-fleet swaps may result in shortages. In this example, shortages may occur in flights 10, 11, 12 and 13 since the seat capacities of these flights are reduced by 30 seats after the swap action. When we analyze the passenger assignments, we observe that flights 11 and 12 will have shortages of 16 and 20 seats, respectively. In Figure 4.5, it can be seen that 10-7 is an alternative path for passengers in itinerary 10-11. However, there are only 8 empty seats available in flight 7. Therefore, 8 passengers of itinerary 10-11 are rerouted through path 10-7 and arrive at MSP. However, since flight 7 arrives at 13:10, these passengers experience 50 minutes of arrival delay. Remaining 8 passengers are spilled (transported through an external arc). Finally, 20 passengers of itinerary 12 are assigned to flight 8 with 60 minutes delay.

This example illustrates the complexity of the problem due to the interrelation among entity types and the necessity of an integrated approach. Moreover, we try to illustrate how passengers in an itinerary may be separated to different paths and how cruise speed control can be integrated with other recovery actions. Proposed problem representation is capable of generating all possible recovery options for each entity while they are evaluated simultaneously with the proposed mathematical formulation.

## 4.2 Mathematical Formulation

The constraints will be constructed in five groups and calculation of cost terms will be explained after the constraints.

### 4.2.1 Flow Balance Constraints

Decision variable $x^r_{fg}$ equals one if entity $r$ flows through arc $(f, g)$, and zero otherwise. Flow balance is satisfied by equation (4.1).

$$\sum_{f:(f,g)\in\mathcal{A}} x^r_{fg} - \sum_{h:(g,h)\in\mathcal{A}} x^r_{gh} = D^r_g \qquad , r \in R, g \in \mathcal{N} \tag{4.1}$$

$$\text{where} \quad D^r_g = \begin{cases} -1 & \text{if } g = s^r\text{, source node of } r \\ 0 & \text{if } g \text{ is a flight or must-visit node} \\ +1 & \text{if } g = t^r\text{, sink node of } r \end{cases}$$

For the sake of generality, we define constraint (4.1) for all aircraft and flight pairs. However, an aircraft may not be appropriate to operate all flights due to technological limitations. These limitations can easily be represented by eliminating flow variables corresponding to infeasible matches.

### 4.2.2 Node Closure Constraints

In order to operate a flight, operating entities should be assigned. For instance, a flight may require an aircraft and a crew team to be operated. We define subset $T^{OP} \subseteq T$ as the set of operating entity types and the parameter $Req^t_f$ as the number of entities of type $t$ needed to operate flight $f$. Decision variable $z_f$ equals one if flight $f$ is cancelled (or node $f$ is closed) and zero otherwise. Constraint (4.2) provides that a flight will be cancelled if the required number of operating entities does not flow through the corresponding node. Constraint (4.3) guarantees that other entities cannot flow through a closed node as well.

$$\sum_{r \in R^t} \left( \sum_{g:(f,g) \in \mathcal{A}} x^r_{fg} \right) = (1 - z_f) Req^t_f \qquad , t \in T^{OP}, f \in F \tag{4.2}$$

$$\sum_{g:(f,g) \in \mathcal{A}} x^r_{fg} \leq (1 - z_f) D^r_f \qquad , t \in T \setminus T^{OP}, r \in R^t, f \in F \tag{4.3}$$

### 4.2.3 Flight Time Constraints

Flight time of a flight node depends on the type of the assigned aircraft. Moreover, flight time can be reduced to some extent by increasing the speed of the assigned aircraft. Let nonnegative continuous decision variables $dt_f$ and $at_f$ represent the actual departure and arrival time of flight $f$, respectively, where $dt_f \in \left[SDT_f, \overline{DT_f}\right]$ and $at_f \in \left(\left[\underline{AT_f}, \overline{AT_f}\right]\right)$. Note that $\underline{AT_f}$ can be smaller when we utilize cruise speed control option. Therefore, arrival time variable $at_f$ is defined over a greater interval resulting in a greater solution space. Finally, let nonnegative continuous variable $\delta t_f$ be the amount of cruise time compression of flight $f$. Then, the relation between actual departure and arrival time, and compression is constructed with equation (4.4).

$$at_f = dt_f + \left( \sum_{r \in R^{aircr.}} \sum_{g:(f,g) \in \mathcal{A}} x^r_{fg} \right) FT^r_f - \delta t_f \qquad , f \in F \tag{4.4}$$

### 4.2.4 Arc Feasibility Constraints

We have four constraints in order to construct arc feasibility such that each corresponding to a different operational rule.

#### 4.2.4.1 Arcs emanating from source nodes

These arcs end in flight nodes that may be assigned to an entity as its first flight in the recovered schedule. An entity will use one of these arcs and reach its first flight node, say $f_{first}$. In this case, $f_{first}$ needs to wait for the ready time of this entity to depart. Therefore, we need a constraint to ensure that the entity is available at the departure time of its first flight. However, only a subset of these arcs are critical for feasibility. They are defined as the set of departure-critical arcs, $DC^r = \{(s^r, g) \in A : SDT_g < \underline{DT_r}\}$, and the constraint for each entity $r$ is defined over $DC^r$ in (4.5).

$$dt_g \geq \underline{DT_r} x^r_{s^r g} \qquad , r \in R, (s^r, g) \in DC^r \tag{4.5}$$

#### 4.2.4.2 Arcs incoming to sink nodes

Similarly, the last flight assigned to entity $r$ cannot arrive later than the latest arrival time of the entity, $\overline{AT^r}$, in order to catch up the original schedule. Constraint (4.6) is limited to the arrival-critical arcs for entity $r$, $AC^r = \{(f, t^r) \in A : \overline{AT_f} > \overline{AT^r}\}$.

$$at_f \leq \overline{AT_f} + \left[\overline{AT^r} - \overline{AT_f}\right] x^r_{ft^r} \qquad , r \in R, (f, t^r) \in AC^r \tag{4.6}$$

#### 4.2.4.3 Intermediate arcs

Intermediate arcs consist of arcs between two flight nodes, and arcs between a flight node and a must-node. If there is a positive flow of entity $r$ between nodes $f$ and $g$, minimum connection time, $CT^r_{fg}$, should be provided between these flights. Set of connection-critical arcs for entity $r$ is defined as $CC^r = \{(f, g) \in \mathcal{A} : f, g \in F \cup M, \overline{AT_f} + CT^r_{fg} > SDT_g\}$, and connection time rule is modeled with Constraint (4.7).

$$dt_g \geq at_f + CT^r_{fg} x^r_{fg} - \overline{AT_f}(1 - x^r_{fg}) \qquad , r \in R, (f, g) \in CC^r \tag{4.7}$$

#### 4.2.4.4 Arcs emanating from or incoming to must-nodes

Recall that must-nodes represent restrictions of entities. Therefore, entities with such restrictions should visit these nodes as formulated in Constraint (4.8).

$$\sum_{g:(m,g)\in\mathcal{A}} x^r_{mg} = 1 \qquad\qquad , r \in R, m \in M^r \tag{4.8}$$

### 4.2.5 Aircraft Properties

Some properties of flights depend on the type of assigned aircraft if inter-fleet aircraft-flight assignments are allowed. Otherwise, these properties would be constant. First such property is the seat capacity. Left-hand-side of constraint (4.9) is the number of passengers assigned to flight $f$. This number is limited by the seat capacity of the assigned aircraft (right-hand-side).

$$\sum_{r\in R^{pass.}} \sum_{g:(f,g)\in\mathcal{A}} x^r_{fg} \leq \sum_{r\in R^{aircr.}} \sum_{g:(f,g)\in\mathcal{A}} x^r_{fg} SCAP^r \qquad , f \in F \tag{4.9}$$

The second property is the limitation on cruise speed. Each aircraft type may speed up to different extents for a particular flight. Maximum cruise speed can be determined by technological constraints or airline policy. This limit can be expressed with an upper bound on cruise speed or equivalently on cruise time compression. We define $\Delta T^r_f$ to be the maximum amount of decrease in cruise time of $f$ if it is operated by aircraft $r$. Cruise time compression variable is bounded by constraint (4.10).

$$\delta t_f \leq \sum_{r\in R^{aircr.}} \sum_{g:(f,g)\in\mathcal{A}} x^r_{fg} \Delta T^r_f \qquad\qquad , f \in F \tag{4.10}$$

### 4.2.6 External Arc Costs

We define $tc^{[e]}$ to be the total cost of flow on external arcs. Recall that $tc^{[e]}$ represents the sum of costs of actions such as ferrying aircraft, deadheading crew members, spilling and allocating passengers to other means of transportation, and ticket cancellation. Let $C_e^{[e]}$ be the cost of unit flow on arc $e$. Then, this cost term is evaluated in (4.11).

$$tc^{[e]} = \sum_{r \in R} \sum_{e \in E^r} C_e^{[e]} x_e^r \tag{4.11}$$

### 4.2.7 Flight Cancellation Costs

Let $C_f^{[c]}$ be the flight cancellation cost of flight $f$. Total flight cancellation cost of the solution, $tc^{[c]}$ is evaluated by (4.12).

$$tc^{[c]} = \sum_{f \in F} C_f^{[c]} z_f \tag{4.12}$$

### 4.2.8 Additional Fuel Costs

Airlines generally operate their aircraft at maximum range cruise (MRC) speeds that result in minimum fuel consumption. $\underline{C_{r,f}^{[f]}}$ is defined as the minimum fuel cost for flight $f$ that can be achieved by aircraft $r$ at MRC speed. Increase in the cruise speed of a flight and inter-fleet reassignments may result in a change in the fuel cost. Defining $T_f^r$ as the cruise time of flight $f$ when operated with aircraft $r$ at MRC speed, and $K_f^r$ as the flight-and-aircraft-dependent exponential, the fuel cost of flight $f$ when operated by aircraft $r$, $fc_f^r$, is expressed as a function of the amount of compression with the following equation:

$$fc_f^r(\delta t_f) = \underline{C_{r,f}^{[f]}} T_f^{K_f^r} \left( \frac{1}{T_f - \delta t_f} \right)^{K_f^r}$$

Letting $r_f$ as the aircraft that is originally assigned to flight $f$, total additional fuel cost of the solution, $tc^{[f]}$, is calculated by (4.13)

$$tc^{[f]} = \sum_{f \in F} \left( \sum_{r \in R^{aircr.}} \sum_{g:(f,g) \in \mathcal{A}} x^r_{fg} fc^r_f(\delta t_f) - \underline{C^{[f]}_{r,f}} \right) \tag{4.13}$$

Derivation of fuel cost function and conic quadratic reformulation scheme to handle nonlinearity in this constraint will be discussed in Section 4.3.

### 4.2.9 Passenger Delay Costs

Passenger delay cost includes cost of goodwill loss, and hence, is difficult to calculate in practice. A straightforward calculation method used in many studies is to use a continuous linear delay cost function by utilizing delay cost per passenger per minute. On the other hand, there is also a belief that the relation between goodwill loss and the amount of delay is nonlinear; and hence, a piecewise cost function would be more appropriate. Due to the complexity of the problem, approximate delay costs are utilized in the literature. In this study, we model and experiment approximate and exact delay cost calculation methods for both linear and piecewise functions.

#### 4.2.9.1 Linear Function with Flight Delay Approximation

Passengers may arrive to their destinations through a set of possible alternative flights due to rerouting decisions. Therefore, each possible final flight for a passenger should be investigated to calculate the actual delay, which increases complexity. A common approximation method is to use flight delay instead of using actual delay of individuals. $N^{arr}_f$ is defined to be the number of passengers that arrive to their destinations through flight $f$ in the original schedule.

$$N^{arr}_f = \sum_{r \in T^{pass.}: Des^r = Des_f} Nb^r$$

Total passenger delay cost, $tc^{[pd]}$, is approximated with constraints (4.14) and (4.15), where decision variable $delay_f$ is the arrival delay of flight $f$ and $C^{[pd]}_f$ is per minute delay of a passenger whose last scheduled flight is $f$.

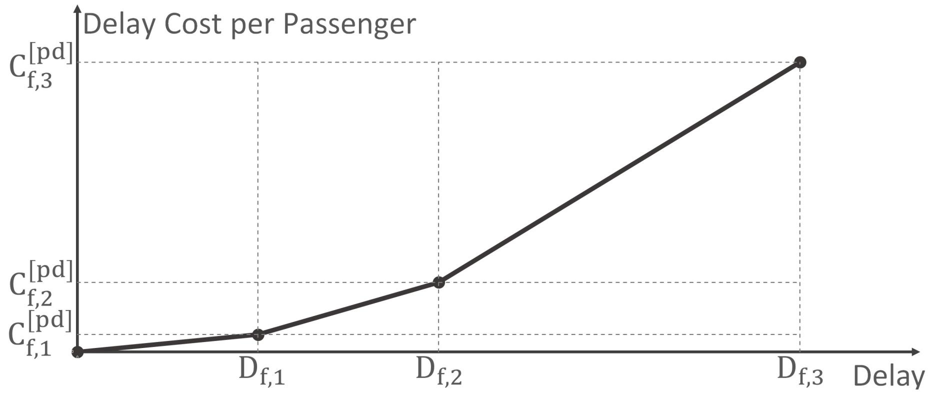


Figure 4.6: A convex piecewise passenger delay cost function.

$$dly_f \geq at_f - SAT_f \qquad , f \in F \tag{4.14}$$

$$tc^{[pd]} = \sum_{f \in F} N_f^{arr} C_f^{[pd]} delay_f \tag{4.15}$$

This is the least complex method; however, it overestimates the delay cost for flight $f$ if some or all passengers in itineraries with final flight $f$ are spilled or rerouted. On the other hand, it underestimates the delay cost if passengers from other itineraries are rerouted and arrive their destination through $f$.

#### 4.2.9.2 Piecewise Function with Flight Delay Approximation

In this method, a convex piecewise delay cost function is used instead of a linear function. An example delay cost function is illustrated in Figure 4.6. For flight $f$, the function is defined by delay points $D_{f,i}$ $(D_{f,0} = 0)$ and corresponding delay costs $C_{f,i}^{[pd]}$ $\left(C_{f,0}^{[pd]} = 0\right)$ where $I_f$ is the number of points that the function changes its slope. Let continuous decision variable $delay_f^i$ be defined over [0,1] for each interval $i$ of flight $f$. Total passenger delay cost is approximated with constraints (4.16) and (4.17).

$$\sum_{i=1}^{I_f}(D_{f,i} - D_{f,i-1})delay_f^i \geq at_f - SAT_f \qquad , f \in F \quad (4.16)$$

$$tc^{[pd]} = \sum_{f \in F} N_f^{arr} \sum_{i=1}^{I_f}(C_{f,i}^{[pd]} - C_{f,i-1}^{[pd]})delay_f^i \quad (4.17)$$

#### 4.2.9.3 Linear Function with Actual Passenger Delay

In the remaining two methods, we propose actual passenger delay cost formulations. To the best of our knowledge, these are the first formulations that both consider rerouting decisions of passengers and realized arrival times of flights in an exact approach. Let $C_r^{[pd]}$ be per minute delay cost of passenger $r$ and decision variable $delay_r$ be the realized delay of this passenger. Then, total linear delay cost of passengers with actual delays is calculated with constraints (4.18) and (4.19), where $SAT^r$ is the scheduled arrival time of passenger $r$.

$$delay_r \geq at_f - SAT^r - \left(\overline{AT_f} - SAT^r\right)\left(1 - x_{ft^r}^r\right)$$
$$, r \in R^{pass.}, f \in F \ni (f, t^r) \in \mathcal{A} \quad (4.18)$$
$$tc^{[pd]} = \sum_{r \in R^{pass.}} C_r^{[pd]} delay_r \quad (4.19)$$

#### 4.2.9.4 Piecewise Function with Actual Passenger Delay

A piecewise delay cost function can be defined for each passenger in a similar manner. Let $D_{r,i}$ be the delay points that the function changes its slope ($D_{r,0} = 0$) and $C_{r,i}^{[pd]}$ be the corresponding delay costs ($C_{r,0}^{[pd]} = 0$) where there are $I_r$ such points for passenger $r$. Continuous decision variable $delay_r^i \in [0, 1]$ is defined for each interval $i$ and total passenger delay cost is calculated with constraints (4.20) and (4.21).

$$\sum_{i=1}^{I_r}(D_{r,i} - D_{r,i-1})delay_r^i \geq at_f - SAT^r - \left(\overline{AT_f} - SAT^r\right)\left(1 - x_{ft^r}^r\right)$$

$$, r \in R^{pass.}, f \in F \ni (f, t^r) \in \mathcal{A} \tag{4.20}$$

$$tc^{[pd]} = \sum_{r \in R^{pass.}} \sum_{i=1}^{I_r}(C_{r,i}^{[pd]} - C_{r,i-1}^{[pd]})delay_r^i \tag{4.21}$$

### 4.2.10 Mathematical Model

The complete mathematical formulation is given below:

Minimize $tc^{[e]} + tc^{[c]} + tc^{[f]} + tc^{[pd]}$

subject to

(4.1) - (4.13)

(4.14) - (4.15) or (4.16) - (4.17) or (4.18) - (4.19) or (4.20) - (4.21)

Proposed formulation is a mixed integer nonlinear programming (MINLP) model since the nonlinear fuel cost function is multiplied with the flow variables in (4.13). In the next section, we will reformulate the problem as a conic quadratic mixed integer programming (CQMIP) problem.

## 4.3 Conic Quadratic Reformulation

In the technical documentation of [2] and [19], it is reported that direct operating costs of a flight consist of a fixed cost, and variable fuel and time related costs depending on cruise speed and time. Considering downstream effects of disruptions and recovery actions on all types of entities, we have already modeled time related costs without isolating the decision to a single flight. For the variable fuel cost term, we develop the fuel cost function with respect to the technical report of [3]. It is stated that airlines generally operate their aircraft at maximum range cruise (MRC) speeds that lead to minimum fuel consumption. Fuel cost is increasing and convex in deviation from MRC speed. $V_f^r$ is defined as the MRC speed of flight $f$ for aircraft $r$, and $\delta v_f$ as the increase in cruise speed. We express the fuel cost of flight $f$ when operated with aircraft $r$ as a function $\delta v_f$ by (4.22), where $K_f^r$ is flight-and-aircraft-dependent parameter.

$$fc_f^r\left(\delta v_f\right) = \underline{C_{r,f}^{[f]}} T_f^{K_f^r} \left( \frac{V_f^r + \delta v_f}{V_f^r} \right)^{K_f^r} \qquad , K_f^r \geq 1 \tag{4.22}$$

Cruise stage distance of flight $f$ can be expressed by $DIST_f = V_f^r T_f^r$ for any aircraft $r$. Note that since distance is constant, $DIST_f = \left(V_f^r + \delta v_f\right)\left(T_f^r - \delta t_f\right)$ should also hold. Using these equalities, we express $\delta v_f$ in terms of the amount of compression in cruise time, $\delta t_f$, with (4.23).

$$\delta v_f = \frac{V_f^r \delta t_f}{T_f^r - \delta t_f} \tag{4.23}$$

Now substituting (4.23) in (4.22), we obtain the fuel cost function given in Section 4.2.8:

$$fc_f^r(\delta t_f) = \underline{C_{r,f}^{[f]}} T_f^{K_f^r} \left( \frac{1}{T_f - \delta t_f} \right)^{K_f^r} \tag{4.24}$$

Let auxiliary variable $y_f^r$ be equal to 1 if flight $f$ is operated by aircraft $r$ in the recovered schedule and 0 otherwise. We also define nonnegative auxiliary decision variables $crt_f^r$ as the actual cruise time of flight $f$ when operated with aircraft $r$, and $\beta_f^r$ as a fuel cost variable. Finally, letting the parameter $\Omega_f^r$ be equal to $\underline{C_{r,f}^{[f]}} {T_f^r}^{K_f^r}$, we show that the nonlinear constraint (4.13) can be linearized as:

$$tc^{[f]} = \sum_{f \in F} \left( \sum_{r \in R^{aircr}} \Omega_f^r \beta_f^r - \underline{C_{r,f}^{[f]}} \right) \tag{4.25}$$

by addition of the following constraints:

$$y_f^r = \sum_{g \ni (f,g) \in \mathcal{A}} x_{fg}^r \qquad , f \in F, r \in R^{aircr.} \tag{4.26}$$

$$crt_f^r = T_f^r - \delta t_f \qquad , f \in F, r \in R^{aircr.} \tag{4.27}$$

$$\beta_f^r \geq \frac{y_f^r}{{crt_f^r}^{K_f^r}} \qquad , f \in F, r \in R^{aircr.} \tag{4.28}$$

Inequality (4.28) is valid since it is a minimization problem. In the current form of the problem we have the nonlinear constraint set (4.28).

Next, we reformulate (4.28) so that the reformulation will represent the hypograph of the geometric mean of $2^l$ nonnegative variables, which is a convex set. In the following, we drop $f$ and $r$ indices of variables and consider $K = k_1/k_2$ where $k_1$, $k_2$ are integers.

**Proposition 4.3.1.** *Inequality* (4.28)*,*

$$\beta \geq \frac{y}{crt^K}$$

*can be equivalently written as*

$$y^{2^l} \leq \beta^{k_2} \times crt^{k_1} \tag{4.29}$$

*where* $l = \lceil \log_2(k_1 + k_2) \rceil$.

*Proof.* Inequality

$$\beta \geq \frac{y}{crt^K}$$

can be first written as

$$y \leq crt^{k_1/k_2} \times \beta$$

taking $k_2^{th}$ power of both sides we get

$$y^{k_2} \leq crt^{k_1} \times \beta^{k_2}$$

Exploiting the fact that $y$ is a 0-1 decision variable in the model, the exponent of $y$ can be increased and set to $2^l$ and we get:

$$y^{2^l} \leq crt^{k_1} \times \beta^{k_2} \times 1^{(2^l - k_1 - k_2)} \tag{4.30}$$

An inequality of the form $r \leq (s_1 s_2 \cdots s_{2^l})^{1/2^l}$ with $s_i \geq 0$ restrictions defines the hypograph of the geometric mean of variables $s_1, \ldots, s_2$, which is a convex set. Inequality (4.30) is of the same form with some restrictions on the variables. $k_1$ of them are identical and equal to $crt_k$, $k_2$ of them equal to $\beta$ and $(2^l - k_1 - k_2)$ of them equal to 1. $\square$

Proposition 4.3.1 shows that inequality (4.28) can be reformulated as an inequality which defines a convex set, namely a hypograph of geometric mean of $2^l$ variables. This leads to the following result.

**Proposition 4.3.2.** *Inequality* (4.28)*,*

$$\beta \geq \frac{y}{crt^K}$$

*with restrictions* $y \in \{0, 1\}, crt \geq 0, \beta \geq 0$*, can be represented using conic quadratic inequalities.*

*Proof.* As given in [17], for a positive integer $l$, an inequality of the form

$$r^{2^l} \leq s_1 s_2 \cdots s_{2^l}, \tag{4.31}$$

for $r, s_1, \ldots, s_{2^l} \geq 0$, i.e. a hypograph of geometric mean of $2^l$ variables, can be expressed equivalently using $O(2^l)$ variables and $O(2^l)$ hyperbolic inequalities of the form

$$u^2 \leq v_1 v_2,\ u, v_1, v_2 \geq 0 \tag{4.32}$$

Furthermore, each constraint $u^2 \leq v_1 v_2$ can be written as a conic quadratic constraint

$$\|(2u, v_1 - v_2)\| \leq v_1 + v_2. \tag{4.33}$$

$\square$

This reformulation enables to model the MINLP problem initiated in Section 4.2 as a CQMIP problem. The modified model with a linear objective function, linear and conic quadratic constraints can be handled by fast algorithms of commercial CQMIP solvers.

### 4.4 Preprocessing

As mentioned before, CNs are large and quick solutions are required in disruption management. Therefore, it is important to eliminate unnecessary variables and constraints without sacrificing optimality. In this section, we describe two preprocessing methods. In partial network approach, we propose an algorithm to obtain the partial networks of entities. Partial network of an entity is a subset of the complete network which excludes nodes and arcs that will not be visited by the entity even after rerouting. Therefore, we can reduce the number of variables and constraints significantly. In the second method, we propose an entity aggregation rule without losing any information and sacrificing optimality.

#### 4.4.1 Partial Networks

A partial network of entity $r$ is the subset of the complete graph, $G = (\mathcal{N}, \mathcal{A})$, which includes the source and sink nodes ($s^r$, $t^r$), and must-visit nodes ($M^r$) of the entity together with the flight nodes that it can visit in a feasible solution. The idea is to reduce the number of flow variables using the fact that not all arcs can be used to transport a particular entity from its origin to its destination. For instance, consider an entity whose destination is LAC and latest arrival time is 17:00. Then, a flight from ORD to DFW with an earliest arrival time of 17:00 is irrelevant to this entity, as well as all arcs incoming to and outgoing from this node. We propose Partial Network Generation Algorithm for efficiently generating the partial network of an entity which is capable of generating all feasible paths that can be used by the entity to reach its destination; and does not include any flight nodes that would not be visited.

The steps of the algorithm for entity $r$ is given in Algorithm 1. The algorithm starts with an empty network. Partial network for $r$ is obtained in line 3 by calling GeneratePath sub-procedure with $N^{curr} = \{s^r\}$, where $s^r$ defines the initial state of the entity. $N^{curr}$ is a temporary path that is updated throughout the algorithm. Finally, external arcs related to entity $r$ are included in line 4 and the partial network is returned.

GeneratePath sub-procedure starts with a temporary path, $N^{curr}$, and tries to connect a flight to the final flight of this path. Sub-procedure stops at line 10 if the desired destination is reached and return to destination is not allowed. Returning to destination is not allowed for passengers, while it is allowed for aircraft and crew members. For instance, the path ORD-DCA-DFW-DCA would not be realistic if the entity in consideration is a passenger (or luggage) that will be transported from ORD to DCA. If the destination has not been reached yet (or the entity may leave and return to its destination), $N^{next}$ is created in line 12, which is the set of candidate flights that can be connected to the last flight of the temporary path. The stopping condition in line 15 is crucial for the efficiency of the algorithm. If a flight is already inserted in the partial network of the entity ($g \in \mathcal{N}^r$), we are sure that all sub-paths emanating from this node to the sink have already been discovered. Therefore, $N^{curr}$ can be inserted without any further search.

Insert sub-procedure simply inserts the nodes and arcs in the temporary set $N^{curr}$ into the partial network of the entity. Note that this sub-procedure is called either in line 16 or in line 19. In the latter one, the temporary path is a complete path from the origin to the destination of the entity. All nodes and arcs in the temporary path are inserted into the partial network. On the other hand, in the prior one, the temporary path is connected to an already inserted node. Since we know that there is a sub-path from the already inserted node ($g$) to the destination, the flights and connections in the temporary path may exist in a feasible path. Therefore, we insert the nodes and arcs of this sub-path to the network, as well. Since we do not insert the nodes and arcs of any other path, generated partial network excludes all nodes and arcs that can not be visited by the entity through a feasible path.

Finally note that partial network generation algorithm of an entity does not depend on the partial networks of the other entities. Therefore, partial network generation process can fully be parallelized with respect to entities. Figures 4.3, 4.4 and 4.5 are example partial networks of a complete network that involves 13 flight nodes.

### 4.4.2 Entity Aggregation

Each individual (aircraft, crew member, and passenger) is defined as an entity so far. By careful aggregation, number of entities can be reduced. It is easy to notice that

**Algorithm 1** Partial Network Generation Algorithm

**procedure** PNGA($r$)
    **Initialization:** $\mathcal{N}^r = \emptyset, A^r = \emptyset, N^{curr} = \{s^r\}$
    $G^r = (\mathcal{N}^r, A^r) \leftarrow$ GeneratePath($N^{curr}$)
    $A^r \leftarrow A^r \cup E^r$
    **return** $G^r = (\mathcal{N}^r, A^r)$
**end procedure**
**procedure** GENERATEPATH($N^{curr}$)
    $f \leftarrow$ last element of $N^{curr}$
    **if** $Des_f = Des^r$ and return to destination is not allowed **then**
        **exit procedure**
    **else**
        $N^{next} \leftarrow \left\{g \in F'(r) : Ori_g = Des_f \text{ and } SDT_g \geq \underline{AT_f} + CT^r_{fg}\right\}$
        **for** each $g \in N^{next}$ **do**
            $N^{curr} \leftarrow N^{curr} \cup \{g\}$
            **if** $g \in \mathcal{N}^r$ **then**
                Insert($N^{curr}$)
            **else**
                **if** $Des_f = Des^r$ **then**
                    Insert($N^{curr} \cup \{t^r\}$)
                **end if**
                GeneratePath($N^{curr}$)
            **end if**
        **end for**
    **end if**
**end procedure**
**procedure** INSERT($N^{curr}$)
    $\mathcal{N}^r \leftarrow \mathcal{N}^r \cup N^{curr}$
    Let $f_i$ be the $i^{th}$ element of $N^{curr}$
    **for** $i = 1$ to $|N^{curr}| - 1$ **do**
        $A^r \leftarrow A^r \cup \{f_i, f_{i+1}\}$
    **end for**
**end procedure**

individuals of an aggregated entity need to have *exactly the same partial network* in order to prevent any loss of information. By this observation, we can extend the rule for aggregation of entities as follows:

*Aggregation Rule: Individuals with common ready time, latest arrival time, origin, destination, connection time between flights, must-visit nodes, technical properties (such as aircraft speed and seat capacity) and delay cost parameters can be aggregated without sacrificing optimality.*

Proposed mathematical formulation can easily be modified to aggregated entities. For instance, let entity, $r$, be defined as the aggregation of $Nb^r$ individuals. In this case, binary decision variables $x^r_{fg}$ need to be defined as nonnegative integer variables with an upper bound of $|Nb^r|$. Similarly, the demand of source and sink nodes of this entity should be changed as $-Nb^r$ and $Nb^r$, respectively.

It can easily be noticed that passengers in an itinerary with common delay cost parameters (in the same fare class) can be aggregated without violating the proposed rule. However, for linear passenger delay cost function with actual passenger delay (Sections 4.2.9.3 and 4.2.9.4), passengers should not be aggregated. On the other hand, it is possible to incorporate a piecewise-step function utilizing realized delays with aggregated passengers. Finally, we need to note that individuals of an aggregated entity can still be transported through different paths.

## 4.5 Computational Results

We test the practicality of the proposed representation and formulation using data provided by Bureau of Transportation Statistics (http://www.transtats.bts.gov /DataIndex.asp). We extract aircraft schedules of a major U.S. airline in January, 2012. Extracted data includes tail numbers, departure and arrival times, flight and cruise times, origins, destinations, and distances of flights. We randomly assign seat capacities and fuel cost parameters to each aircraft. Six different seat capacities are used in the experimentation: 150, 160, 180, 200, 260 and 300. For itinerary generation, we have first defined each flight as a single-flight itinerary. Then, we designate flight pairs with destination-origin match and available passenger connection time as possible passenger connections. We generate two-flight and three-flight itineraries using these connection alternatives. We randomly assign the number of passengers in each flight and assign these passengers to itineraries including these flights. Finally, we generate scheduled routings of crew teams in a similar manner. We have used a recovery horizon of 2000 minutes. Spill cost per passenger is set to $457.8 that is evaluated by the method proposed by [59]. Flight cancellation, aircraft ferrying, and crew deadheading costs are set to $20000, $10000, and $500, respectively. Fuel cost coefficients are randomly selected from U(1.5, 3.5) and assigned to each aircraft, where U(a, b) is a uniform distribution in interval (a, b). For linear passenger delay cost function, we have used $1.09 per passenger per minute as proposed by [59]. For piecewise passenger delay cost function, we have used four steps: $D_{f,i} = 30, 60, 120$, and $240$, for $i = 1, 2, 3, 4$. The corresponding delay costs per passenger are set to $25, $60, $130, and $300, respectively.

We have tested our approach in three different networks. In Table 4.4, we summarize the hubs, number of flights and number of entities included in these networks. Abbreviations *ac*, *cr*, *it*, and *ps* are used for aircraft, crew team, itinerary, and passenger, respectively.

Table 4.4: Characteristics of the networks

| Network | Hubs | $\lvert\mathcal{F}\rvert$ | $\lvert R^{ac}\rvert$ | $\lvert R^{cr}\rvert$ | $\lvert R^{it}\rvert$ | $\lvert R^{ps}\rvert$ |
|---|---|---|---|---|---|---|
| N1 | DCA | 103 | 27 | 38 | 139 | 16,393 |
| N2 | ORD | 233 | 83 | 105 | 574 | 39,405 |
| N3 | ORD,DCA,LAX | 504 | 168 | 225 | 1,177 | 85,141 |

We apply partial network approach in all instances. We generate partial networks both by using the flexibility of cruise speed control and excluding it. In Table 4.5 partial network structures for aircraft are summarized. Avg($\lvert\mathcal{N}^{ac}\rvert$) and Avg($\lvert A^{ac}\rvert$) are the average number of nodes and arcs in partial networks of aircraft, respectively. Firstly, we observe a significant reduction in the number of decision variables and constraints with partial network approach, as the average size of partial networks is much smaller than the complete network. Secondly, we observe an increase in the number of nodes and arcs with cruise speed control option. This indicates that a significant number of new rerouting opportunities can be created by speeding up some flights.

Table 4.5: Effect of cruise speed control on partial networks

| | without speeding up flights | | with speeding up flights | |
|---|---|---|---|---|
| Network | Avg($\lvert\mathcal{N}^{ac}\rvert$) | Avg($\lvert A^{ac}\rvert$) | Avg($\lvert\mathcal{N}^{ac}\rvert$) | Avg($\lvert A^{ac}\rvert$) |
| N1 | 15.11 | 27.93 | 16.56 | 29.89 |
| N2 | 15.47 | 27.71 | 17.02 | 29.81 |
| N3 | 21.33 | 37.48 | 24.54 | 43.15 |

We have experimented four proposed delay cost calculation methods: linear function with flight delay approximation ($L^-$), piecewise function with flight delay approximation ($PW^-$), linear function with actual delay ($L^+$) and piecewise function with actual delay ($PW^+$). For experiments with actual delay costs ($L^+$ and $PW^+$), passengers are modeled explicitly while for approximations we have used aggregation approach for passengers in each itinerary. It is expected that cruise speed control option will increase solution times. On the other hand, it provides a significant growth in solution space. In order to observe the tradeoff between its burden in solution time and improvement in costs, we define $CS^+$ as the proposed approach and $CS^-$ as the proposed approach without using cruise speed control option (note that $CS^-$ is a mixed integer programming (MIP) model).

Four of the disruption types are tested. For each network, we have created departure delay scenarios including 1 to 5 departure delays each ranging from 45 to 120 minutes. Similarly 1 to 5 randomly selected flights are cancelled for cancellation scenarios. For delayed ready time instances, we have randomly selected an aircraft and delayed its ready time by 60, 120, 180, 240 and 300 minutes. Finally, a hub is closed for 60, 120, 180, 240 and 300 minutes in hub closure scenarios. Solution time is set to 30 minutes for hub closure instances and to 15 minutes for the remaining.

### 4.5.1 N1 - Single-hub (DCA) scenarios

About 86% of all instances are optimized within the given time limit while the maximum gap is 2.2%. Gaps are calculated by dividing the difference between the best integer solution and best lower bound to the best integer solution. Average solution times are displayed in Table 4.6. Solutions are obtained less than 10 seconds (four minutes) on the average without (with) cruise speed control option. In order to verify whether it is worth to use realized delays, we check passenger delay costs of the solutions. We observe that total passenger delay cost is underestimated with $L^-$ and $PW^-$ by about $5,495 on the average. This amount may be negligible for severe disruptions such as closure of a hub with a long duration, while it probably affects decisions in minor disruptions.

Table 4.6: Solution times for N1 (in seconds)

| | | Flight Delay | Cancellation | Aircraft Delay | Hub Closure | Average |
|---|---|---|---|---|---|---|
| $CS^-$ | | | | | | |
| | $L^-$ | 1.3 | 1.4 | 1.3 | 2.4 | 1.6 |
| | $PW^-$ | 1.3 | 1.4 | 2.3 | 1.4 | 1.6 |
| | $L^+$ | 6.5 | 12.1 | 7.3 | 8.6 | 8.6 |
| | $PW^+$ | 8.5 | 27.7 | 15.4 | 16.3 | 17.0 |
| $CS^+$ | | | | | | |
| | $L^-$ | 357.6 | 5.5 | 61.1 | 11.5 | 108.9 |
| | $PW^-$ | 556.2 | 7.1 | 118.3 | 22.2 | 176.0 |
| | $L^+$ | 320.1 | 33.7 | 72.2 | 27.2 | 113.3 |
| | $PW^+$ | 563.3 | 585.5 | 215.9 | 780.6 | 536.3 |

Total disruption and recovery costs are tabulated in Table 4.7. Despite the increased solution times, we observe a significant improvement in costs with cruise speed control option. Percent improvements are calculated by dividing the difference in objective functions to the cost with $CS^-$. Opposed to delay propagation, speeding up early flights help mitigate delays. Therefore, improvement is expected in delay scenarios. On the other hand, main reason of improvement in cancellation and hub closure scenarios is the availability of new rerouting and swap opportunities. On the average, cruise speed option provides about 6.98% reduction in costs.

Table 4.7: Total disruption and recovery costs for N1

| | $CS^-$ | $CS^+$ | % Improvement |
|---|---|---|---|
| Flight Delay | 63,792 | 53,643 | 15.9% |
| Cancellation | 435,924 | 405,348 | 7.0% |
| Aircraft Delay | 160,262 | 137,337 | 14.3% |
| Hub Closure | 602,374 | 577,854 | 4.1% |

### 4.5.2 N2 - Single-hub (ORD) scenarios

Solution times and costs of N2 instances are tabulated in Table 4.8 and Table 4.9, respectively. About 78.7% of all instances are solved to optimality. Average gap is only 0.74%. However, maximum gap reaches 10.13% for instances with $L^+$ or $PW^+$ and $CS^+$. On the other hand, maximum gap is less than 0.5% for approximations or for actual delay cost calculation with $CS^-$. Therefore, we believe that size of N2 is near the upper bound for which real-time solutions can be provided using actual delay cost calculation and cruise speed control option with the proposed approach. As in N1 scenarios, we observe a significant improvement in costs with cruise speed control option.

Table 4.8: Solution times for N2 (in seconds)

| | | Flight Delay | Cancellation | Aircraft Delay | Hub Closure | Average |
|---|---|---|---|---|---|---|
| $CS^-$ | | | | | | |
| | $L^-$ | 1.8 | 2.1 | 2.8 | 4.9 | 2.9 |
| | $PW^-$ | 2.9 | 2.3 | 2.7 | 3.8 | 2.9 |
| | $L^+$ | 14.3 | 67.4 | 32.8 | 88.8 | 50.8 |
| | $PW^+$ | 22.4 | 75.4 | 322.4 | 923.3 | 335.9 |
| $CS^+$ | | | | | | |
| | $L^-$ | 470.2 | 159.3 | 11.9 | 218.7 | 215.0 |
| | $PW^-$ | 448.9 | 103.2 | 128.6 | 447.4 | 282.0 |
| | $L^+$ | 268.9 | 177.2 | 370.3 | 918.4 | 433.7 |
| | $PW^+$ | 500.8 | 484.1 | 555.4 | 1517.1 | 764.4 |

Table 4.9: Total disruption and recovery costs for N2

| | $CS^{-}$ | $CS^{+}$ | % Improvement |
|---|---|---|---|
| Flight Delay | 30,280 | 27,282 | 9.9% |
| Cancellation | 795,223 | 729,457 | 8.3% |
| Aircraft Delay | 645,555 | 627,299 | 2.8% |
| Hub Closure | 2,275,324 | 2,242,558 | 1.4% |

### 4.5.3 N3 - Three-hub scenarios

For N3 scenarios, we have aggregated passengers in an itinerary, and hence, used $L^{-}$ and $PW^{-}$ methods to estimate passenger delay costs. Solution times and costs of N3 instances are tabulated in Table 4.10 and Table 4.11, respectively. About 79.3% of all instances are solved to optimality while the average (maximum) gap is 0.3% (2.6%). Average solution time with cruise speed control option is about ten minutes while about 6.9% reduction in costs is provided.

Table 4.10: Solution times for N3 (in seconds)

| | | Flight Delay | Cancellation | Aircraft Delay | Hub Closure | Average |
|---|---|---|---|---|---|---|
| $CS^{-}$ | | | | | | |
| | $L^{-}$ | 3.8 | 72.7 | 13.9 | 95.3 | 46.4 |
| | $PW^{-}$ | 5.9 | 57.9 | 14.1 | 85.1 | 40.8 |
| $CS^{+}$ | | | | | | |
| | $L^{-}$ | 407.1 | 891.8 | 233.4 | 932.2 | 616.1 |
| | $PW^{-}$ | 720.8 | 532.1 | 461.7 | 1638.4 | 838.3 |

In order to observe the relation between the improvement provided by cruise speed control option and severity of disruptions, we further investigate instances. In Figure 4.7, costs with respect to the number of delayed flights are illustrated. Similar results are observed for the remaining disruption types as well. The observations suggest that cruise speed control option becomes more valuable as the disruption scenario gets more complicated.

Table 4.11: Total disruption and recovery costs for N3

| | $CS^-$ | $CS^+$ | % Improvement |
|---|---|---|---|
| Flight Delay | 61,652 | 56,563 | 8.3% |
| Cancellation | 245,612 | 219,290 | 10.7% |
| Aircraft Delay | 173,651 | 161,442 | 7.0% |
| Hub Closure | 1,492,851 | 1,400,328 | 6.2% |

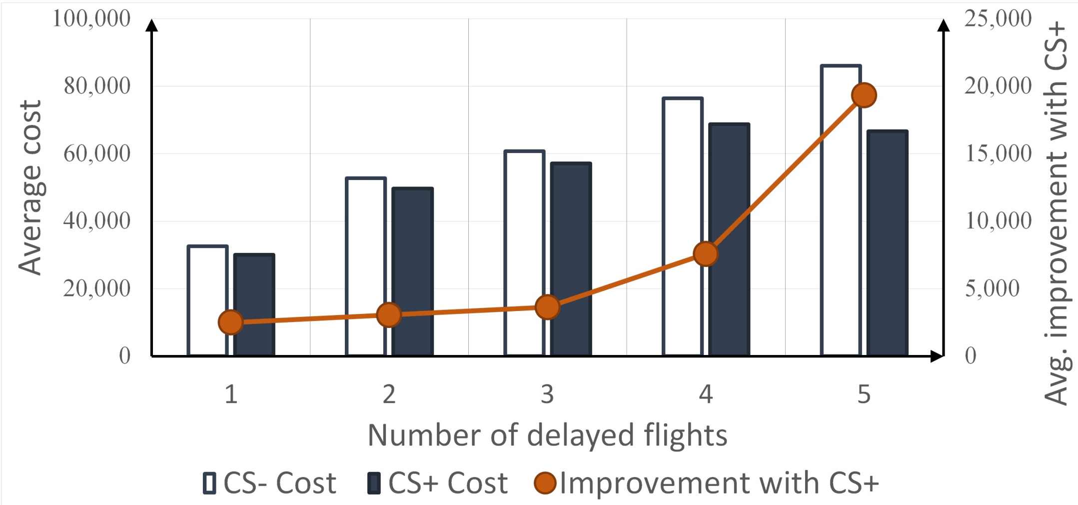


Figure 4.7: Total costs with respect to number of delayed flights in N3

In Figure 4.8, we observe percent improvements in cost terms by cruise speed control option with respect to different disruption types. We observe improvement in passenger delay costs for all disruption types as cruise speed control helps mitigate delays. Moreover, it helps maintain passenger connections so there is an improvement in external arc costs (spilling costs). In hub closure and aircraft delay scenarios, we also observe that there is a reduction in the total number of ferried aircraft and deadheaded crew members. Infeasibilities by flight cancellations may spread through the schedules of aircraft and crew members, and result in severe disruptions. Hub closure scenarios are obviously the most complex scenarios resulting in many cancelled flights. Therefore, network connectivity becomes more valuable than delay mitigation in cancellation and hub closure scenarios. Reduction in the number of cancelled flights by cruise speed control option is 0.6 and 1.4 on the average for cancellation and hub closure scenarios, respectively.

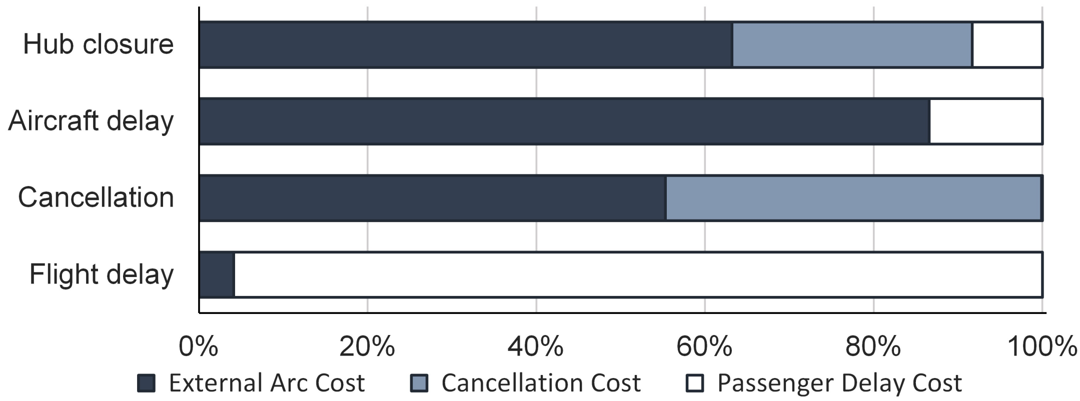


Figure 4.8: Percent improvement in cost terms with cruise speed control

Analysis of the effect of severity of disruptions on disruption costs is illustrated in Figure 4.9. Relation between the number of departure delays and cancellations with total cost resembles a linear dependency. A similar relation exists between the amount of ready time delay of aircraft and total cost for delays less than three hours. However, the increase in total cost decreases for greater delays. The reason of this relation is that a three hour delay is severe enough to force the airline utilize costly recovery actions already. In hub closure scenarios, we observe a jump in the increase in total cost for closures greater than two hours. Many affected flights in scenarios with a closure less than or equal to two hours may still be operated using departure holding and speeding up. However, the number of flights that needs to be cancelled increases greatly for longer closures.

## 4.6 Conclusion

Recently, there is an increasing effort in integrated airline recovery approaches for airline disruption management problem due to high passenger inconvenience and crew recovery costs with sequential approaches. Main challenge in integration is the increased problem size while airlines require real-time solutions. In this study, we propose a general network representation for the problem that captures the state of the entities and allows integration of any entity type in the same manner while keeping the problem size in reasonable limits. Another advantage of the proposed representation is that all recovery actions including all rerouting possibilities for each entity can

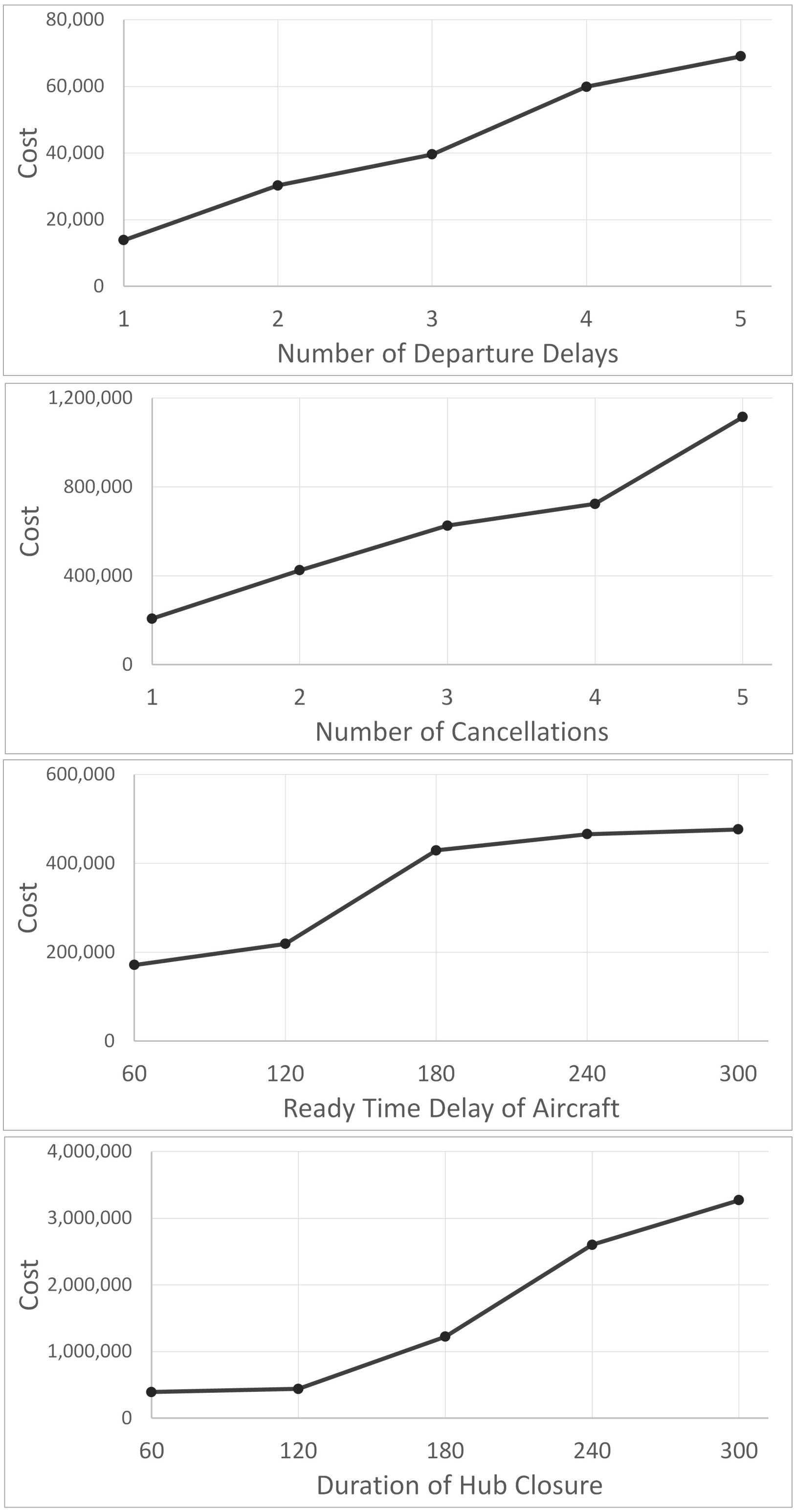


Figure 4.9: Total disruption and recovery costs with respect to the severity of disruptions.

easily be integrated to ensure optimality.

Service quality is becoming more important due to the high competition in the industry. Therefore, evaluating all possible passenger recovery actions in cases of disruptions is crucial. In the literature, there are several passenger delay cost calculation methods to evaluate passenger recovery options, however, they do not fully capture the dynamic nature of operations. To the best of our knowledge, this study is the first to model each passenger explicitly and evaluate passenger delay costs by considering both the rerouting decisions and realized delays of flights in the recovered schedule. We propose a linear and a piecewise passenger delay cost function. For larger problems, we also propose approximation approaches similar to the ones proposed in the literature.

In addition to common recovery actions, we also integrate cruise speed control action in our solution space. Our experiments have shown that speeding up flights may be very beneficial to mitigate delays and preserve passenger connections in cases of disruptions. Moreover, we observe an improvement in the connectivity of the network as new swap and rerouting options are created. However, speeding up a flight increases fuel consumption, and hence, additional fuel cost is incurred. There is a nonlinear tradeoff between fuel consumption and aircraft speed. However, the resulting formulation is second-order cone programming representable. Therefore, we can create conic quadratic constraints for the nonlinear constraints and solve the problem with commercial MIP solvers such as IBM ILOG CPLEX. With the proposed reformulation, solution times have increased but kept within reasonable limits. On the other hand, significant improvements in disruption and recovery costs are observed.

Finally, we propose two important preprocessing approaches for enhancing the performance of the proposed approach without sacrificing optimality. In the first method, an efficient algorithm to generate partial networks of entities is proposed in order to eliminate unnecessary variables and constraints. In the second one, we propose a rule to aggregate entities that needs to be satisfied to preserve optimality. In our experimentations, we managed to optimize single-hub scenarios without aggregation while for three-hub scenarios we have aggregated passengers in the same itinerary.

Aggregating passengers in the same itinerary is a common approach in the literature and reduces the problem size significantly. On the other hand, explicitly modeling each passenger has several advantages. Firstly, it allows calculating passenger delay costs using the realized delays in the recovered solution. Moreover, each individual can be treated differently. For instance, different cost coefficients and different recovery actions may be assigned to different individuals. Despite the increased problem size, we have managed to achieve real-time solutions by explicitly modeling each passenger in single-hub scenarios.

Our alternative problem representation and preprocessing approaches enhance solution times and enable fast solutions to complex formulations. However, solving instances related with huge networks is still challenging. In this chapter, we have used flight delay approximation method to deal with large instances. In the next chapter, we propose a heuristic approach to solve large instances with complex formulations where actual passenger delay is evaluated and cruise speed control action is utilized. The approach reduces the solution space rather than simplifying the formulation.

# CHAPTER 5

# A REAL TIME RECOVERY APPROACH TO INTEGRATED AIRLINE DISRUPTION MANAGEMENT PROBLEM

In this chapter we propose a heuristic approach, *Isolation Heuristic*, that tries to handle limited solution time requirement in airline disruption management. Integrated airline recovery problem is very complex and practical instances are large. In the previous chapter, we propose four different passenger delay cost formulations where the more realistic ones are more complex. Furthermore, integrating cruise speed control action adds complexity to the formulation. We have tested four of the delay cost formulations with and without utilizing cruise speed control action. We observe that the most complex formulations in three-hub networks of a major U.S. airline cannot be solved in real-time, while we achieve good results (having an average gap of 0.3%) with approximation models. In order to deal with larger instances with complex formulations, problem size should carefully be reduced. Most integrated airline recovery approaches in the literature point out the importance of controlling problem size and complexity. Common techniques to achieve tractability in real time include:

- solving individual or partially integrated recovery problems sequentially, and
- using approximations for passenger delay cost.

Sequential recovery approaches generally result in high passenger inconvenience. On the other hand, approximation models underestimate or overestimate actual passenger delay cost.

The aim of the heuristic approach is to provide real time solutions with complex formulations that maintains integration and realistic delay cost calculation. We propose an alternative way to reduce problem size. Dispatchers in AOCCs alter only a subset of operations while recovering schedules against disruptions in order to reduce the complexity of the problem, and provide *stability* of the schedules. Note that enlarging this subset also enlarges the solution space and may provide lower cost recovery decisions. Proposed heuristic tries to control the tradeoff of stability and solution time with the quality of the solutions by mimicking the intuitive decision making process of dispatchers in a systematic way.

In Section 5.1, we propose a connection network (CN) representation similar to the one proposed in Chapter 4. We propose to generate a CN for each entity that is capable of generating all possible recovery actions within a planning horizon. The planning horizon is generally set to one day since there is enough time at night for recovery against disruptions. An efficient algorithm to create CNs of entities is proposed and the idea is illustrated on an example problem. In Section 5.2, we proposed an algorithm to revise the CNs quickly at the moment a disruption occurs. Revised CNs are actually subsets of the original ones and reflect the state of the entities at the beginning of the recovery horizon. *Isolation Heuristic* is presented in Section 5.3. Parameters used to balance the problem size and quality of solutions are explained in detail. An important aspect of the proposed approach is that it only reduces the problem size independent of the optimization approach. Therefore, it can be integrated with any methodology. The outline of the proposed real time recovery approach to integrated airline disruption management problem is illustrated in Figure 5.1. The processes after the revision of CNs could be parallelized as well, if necessary. In our experimentations which are presented in 5.4, we use four mathematical models proposed in Chapter 4 each with a different level of complexity. We give our remarks on the proposed heuristic in 5.5.

## 5.1 Problem Representation

The problem addressed in this chapter is real time airline disruption management problem. Problem in consideration may involve aircraft only (aircraft recovery prob-

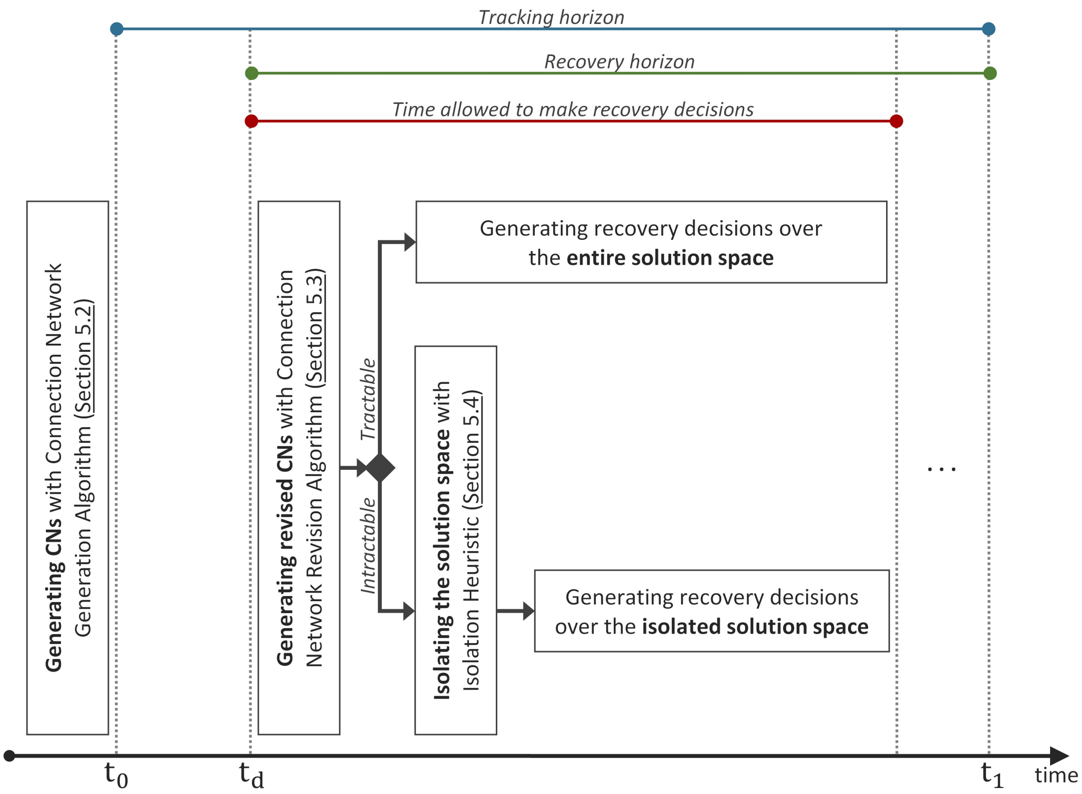


Figure 5.1: Solution Procedure

lem). However, proposed solution approach is more beneficial when rerouting decisions of other entities such as crew members and passengers are also important. There exists a planned schedule for each entity. A planned schedule consists of a series or a path of flights together with departure and arrival time decisions. Scheduled flights are continuously controlled by the dispatchers in AOCCs during the tracking horizon, $[t_0, t_1]$. A common option for $t_1$ is the end of day of operations since generally there is enough slack time at night to resume original schedules. Finally, we are given a disruption at time $t_d \in [t_0, t_1]$ that is severe enough to prevent operating the original schedules. The objective function of airline disruption management problem may vary. A common objective is to minimize disruption and recovery costs as in the formulations proposed in Chapter 3 and 4. Another objective may be to catch up the original schedule as early as possible. One can achieve lower cost solutions with a longer recovery horizon, however, it is generally desirable to limit recovery horizon length in order to reduce the disturbance of the disruption on scheduled operations. Moreover, since new disruptions may occur, long-term recovery plans are less likely to be operated. Finally, *stability* is an important aspect in recovery problems. Decision makers generally desire that the deviation between the original and recovered schedules is little.

We propose a connection network, or activity-on-node, representation which has several advantages. Firstly, number of nodes is limited with the number of flights. Similarly, number of connection arcs between nodes is directly related to the number of rerouting opportunities. Therefore, size of the problem is kept in its natural limits. We propose to model entities of all types with CNs. Integrated recovery problems deal with the superimposition of CNs of all entities, and structure of CNs is very suitable to define the interdependencies among entities of different types.

### 5.1.1 Entities

In the proposed representation, each entity will have a CN that is capable of generating all alternative paths in addition to the scheduled path. By modeling entities of all types with CNs, we guarantee that generated solution space will contain all possible recovery actions. Let $E$ be the set of all entities related to our recovery problem (aircraft, passengers, crew members, etc.). Each of these entities have the following properties:

| | | |
|---|---|---|
| $Ori_e(Des_e)$ | : | Origin (destination) of entity $e$ |
| $RT_e(LAT_e)$ | : | Ready time (latest arrival time) of entity $e$ |

Origin of an entity is the starting location of the entity at the beginning of the original schedule, and the destination is the location that the entity is expected to reach at the end of the schedule. Origin and destination of an aircraft may be a maintenance station. On the other hand, the O-D pair of a passenger would be the origin of the first flight and destination of the last flight of his/her itinerary.

Ready time is related to the availability of the entity. A passenger would be available at its origin just before the departure time of his/her first flight. On the other hand, an aircraft may be available since the beginning of tracking horizon. If the entity is a crew member, ready time is related with the work rules. Latest arrival time designates the latest time that the entity needs to reach its destination. This parameter would be evaluated according to the maximum allowable delay for passengers and according to the work rules for crew members. For aircraft, latest arrival time may be the end of day if there are no additional restrictions such as scheduled maintenances.

### 5.1.2 Nodes

All activities having the following properties are modeled with nodes:

- having defined origin and destination (these locations may be the same)
- having a time window for the starting time
- having a scheduled duration

Some examples satisfying these conditions are flights, scheduled aircraft maintenances, away-from-home limitations of crew members, etc. Notation for nodes is given below:

$$
\begin{array}{rcl}
Ori_i(Des_i) & : & \text{Origin (destination) of activity } i \\
A_i(B_i) & : & \text{Earliest (latest) time that activity } i \text{ can start at } Ori_i \\
D_i & : & \text{Duration of activity } i \\
S_i(C_i) & : & \text{Scheduled starting (completion) time of activity } i \\
& & (s_i \in [A_i, B_i] \text{ and } C_i \in [A_i + D_i, B_i, D_i])
\end{array}
$$

If the activity is a flight, the origin and destination would be different, and they may be the same location if the activity is the maintenance of an aircraft. We have time windows for the starting time of activities and a constant service duration. However, variable service times may easily be incorporated with the proposed approach. We define $\mathcal{N}$ to be the set of all scheduled activities (nodes) in the problem. In addition to location and time window constraints, there may be some additional restrictions between activity-entity pairs stating that the entity cannot visit the activity node. For instance, an aircraft $e$ may not be appropriate to operate a flight $i$ due to technological limitations. Moreover, available time window of the entity may restrict some activity assignments. In particular, entity $e$ cannot visit any of the activity nodes in the following subset due to time window restrictions:

$$\{i \in N : B_i < RT_e \text{ or } A_i + D_i > LAT_e\}$$

We define $\mathcal{N}_e^{feas} \subseteq \mathcal{N}$ to be the set of nodes that entity $e$ can visit.

### 5.1.3 Arcs

Arcs represent feasible connections among nodes with respect to geographical constraints and time window requirements. Existence of an arc in the CN of an entity states that the entity may take place in the activities at the starting and ending nodes of the arc sequentially. An arc from node $i$ to node $j$ has the following characteristics:

- has a duration, $D_{ij}$ ($D_{ij}$ may be equal to $0$)
- there is a possibility to connect these activities with respect to time window

restrictions, i.e.,
$B_j \geq A_i + D_i + D_{ij}$

- either destination-origin (D-O) match ($Des_i = Ori_j$) is satisfied or it is possible for the entity to travel from $Des_i$ to $Ori_j$ (in the latter case, $D_{ij}$ contains the duration of the trip)

For the sake of generality, if destination-origin match is not satisfied between activities $i$ and $j$ and it is not possible to travel between these locations, we set the duration of these connections to a large value, $D_{ij} = t_1 - t_0$. If D-O match is satisfied and activity $j$ can be started immediately after the completion of activity $i$, we set $D_{ij} = 0$. On the other hand, even if D-O match is satisfied, there may be a required time between the completion time of the prior activity and the starting time of the latter. For instance, between two consecutive flights assigned to an aircraft, a turnaround time needs to be provided. This duration is also expressed with the arc length, i.e. $D_{ij}$.

### 5.1.4 Source and Sink Nodes

For each entity $e$, we insert a source node ($s_e$), and a sink ($t_e$) node. The aim of the proposed approach is to generate all possible paths from $s_e$ to $t_e, \forall e \in E$, so that the entire solution space is obtained. These artificial nodes have the following properties:

- $Des_{s_e} = Ori_e$ and $Ori_{t_e} = Des_e$
- $A_{s_e} = RT_e$ and $B_{t_e} = LAT_e$

### 5.1.5 Connection Network Generation Algorithm

For each entity $e \in E$, we propose to create a CN having the structure explained above. CN of entity $e$ is designated with $G_e = (\mathcal{N}_e, A_e)$. We propose *Connection Network Generation Algorithm* (CNGA) to efficiently generate these networks. This algorithm may be regarded as a generalized version of the Partial Network Generation Algorithm proposed in Section 4.4.1.

In the *Initialization* step, we create empty sets for nodes and arcs, $N_e = \emptyset, A_e = \emptyset$.

**Algorithm 2** Connection Network Generation Algorithm

**procedure** CNGA($e$)<br>
    **Initialization:** $\mathcal{N}_e = \emptyset, A_e = \emptyset, \mathcal{N}_e^{feas} = \mathcal{N}_e^{feas} \cup \{t_e\}, N^{curr} = \{s_e\}$<br>
    $G_e = (\mathcal{N}_e, A_e) \leftarrow$ GeneratePath($N^{curr}$)<br>
**end procedure**<br>
**procedure** GENERATEPATH($N^{curr}$)<br>
    $i \leftarrow$ last element of $N^{curr}$<br>
    **if** $i = t_e$ **then**<br>
        **exit procedure**<br>
    **else**<br>
        $N^{next} \leftarrow \left\{ j \in \mathcal{N}_e^{feas} : B_j \geq A_i + D_i + D_{ij} \right\}$<br>
        **for** each $j \in N^{next}$ **do**<br>
            $N^{curr} \leftarrow N^{curr} \cup \{j\}$<br>
            **if** $j \in \mathcal{N}_e$ **then**<br>
                Insert($N^{curr}$)<br>
            **else**<br>
                GeneratePath($N^{curr}$)<br>
            **end if**<br>
        **end for**<br>
    **end if**<br>
**end procedure**<br>
**procedure** INSERT($N^{curr}$)<br>
    $\mathcal{N}_e \leftarrow \mathcal{N}_e \cup N^{curr}$<br>
    Let $i_k$ be the $k^{th}$ element of $N^{curr}$<br>
    **for** $k = 1$ to $|N^{curr}| - 1$ **do**<br>
        $A_e \leftarrow A_e \cup \{(i_k, i_{k+1})\}$<br>
    **end for**<br>
**end procedure**

We insert the sink in $N_e^{feas}$ which will be the last node of all paths for entity $e$. $N^{curr}$ is a temporary set of activities and we start by inserting the source node, $s_e$.

In *GeneratePath* subprocedure, we investigate the last activity in $N^{curr}$. If the last activity is the sink (if $i = t_e$), we stop. Otherwise, we obtain another temporary set of nodes, $N^{next}$, that may be connected to node $i$. From each activity $j \in N^{next}$, we try to find a path reaching $t_e$.

Condition in line 13 is important for the efficiency of the algorithm. During the execution of the algorithm, if node $j$ that is currently being investigated is already included in the CN of the entity, we are sure that all paths emanating from $j$ and reaching $t_e$ are already constructed. Therefore, we insert the path from $s_e$ to $j$ which is represented by $N^{curr}$ into the CN and stop. This guarantees that none of the connection arcs in the CN are checked for more than once. If the condition is not satisfied, we are on a path that is not discovered yet, and hence, we check further connections.

*Insert* is a simple subprocedure that inserts the current path represented by $N^{curr}$ into the CN of the entity.

Note that each entity has a scheduled path, which is feasible prior to the disruption. Since the CN constructs all feasible paths for the entity, scheduled path is included in the network, as well. We designate the scheduled network of entity $e$ (which actually is a single path from $s_e$ to $t_e$) by $SG^e = (SN^e, SA^e)$, where $SN^e \subseteq N^e$ and $SA^e \subseteq A^e$.

### 5.1.6 Example

In order to demonstrate the idea of CNGA and *Connection Network Revision Algorithm* that will be explained in Section 5.2, we introduce an example. Scheduled flights are modeled as nodes and the arcs represent flight connections. Arc durations are related with entity types. For simplicity, we set the required turnaround time for aircraft and connection time for crew / passenger to 30 minutes.

Original flight schedules of three aircraft are tabulated in Table 5.1. Note that the scheduled start ($S_i$) and completion times ($C_i$) of nodes correspond to scheduled de-

Table 5.1: Flight schedule of the example

| Tail number | $i$ (Flight number) | $Ori_i$ | $Des_i$ | $S_i$ | $C_i$ |
|---|---|---|---|---|---|
| N322AA | 1 | ORD | DCA | 5:30 | 7:10 |
| | 2 | DCA | ORD | 7:50 | 9:30 |
| | 3 | ORD | DFW | 10:00 | 12:20 |
| | 4 | DFW | ORD | 13:00 | 15:20 |
| | 5 | ORD | DCA | 16:30 | 18:10 |
| N345AA | 6 | LAX | ORD | 6:00 | 9:40 |
| | 7 | ORD | MSP | 12:00 | 13:10 |
| | 8 | MSP | ORD | 14:00 | 15:10 |
| | 9 | ORD | LAX | 16:00 | 19:40 |
| N5FCAA | 10 | DCA | ORD | 9:00 | 10:40 |
| | 11 | ORD | MSP | 11:10 | 12:20 |
| | 12 | MSP | ORD | 13:00 | 14:10 |
| | 13 | ORD | DCA | 16:00 | 17:40 |

parture and arrival times of flights, respectively. Since a flight cannot depart before its scheduled departure time, we also set $A_i = S_i$. $B_i$, on the other hand, is related with the maximum departure delay allowed.

For illustration, consider the first aircraft with tail number N322AA. We assume that the origin of this aircraft is ORD ($Ori_{N322AA}$ =ORD) and it can operate its first flight any time after 4:00 ($RT_{N322AA}$ =4:00). We set the destination to DCA ($Des_{N322AA}$=DCA) and assume that it needs to arrive before 19:00 ($LAT_{N322AA}$ = 19:00). The original routing is 1-2-3-4-5. CN created by CNGA is displayed in Figure 5.2. Each arc represents a feasible flight connection (with respect to time windows and locations), and each path from source to sink in the network represents a feasible aircraft routing.

Now let us consider a passenger with two consecutive flights, 2 and 7. We can set the ready time to the scheduled departure time of flight 2 ($RT_{2-7}$=7:50). On the other hand, latest arrival time is related with airline policy. Origin and destination are DCA and MSP, respectively. CN of this passenger created by CNGA is displayed in Figure 5.3, which includes all reallocation alternatives: 2-7, 2-11, 10-7, and 10-11.

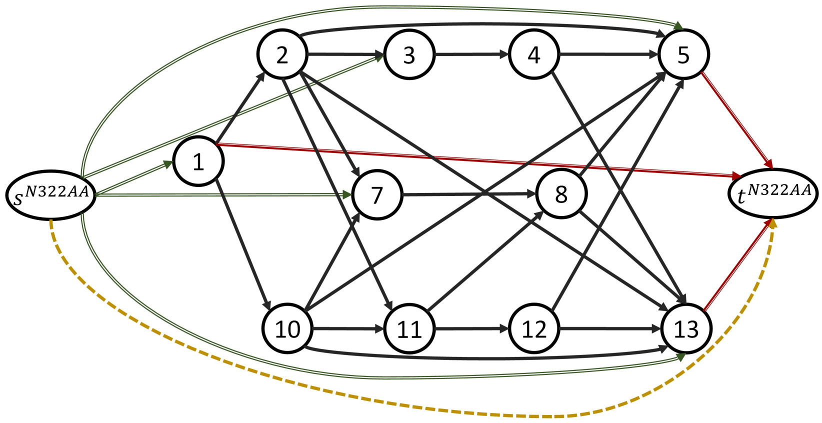


Figure 5.2: CN of aircraft N322AA

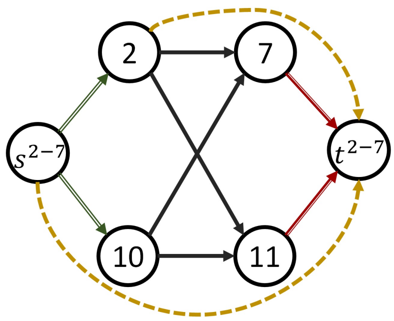


Figure 5.3: CN of a passenger (itinerary) with scheduled flights 2-7

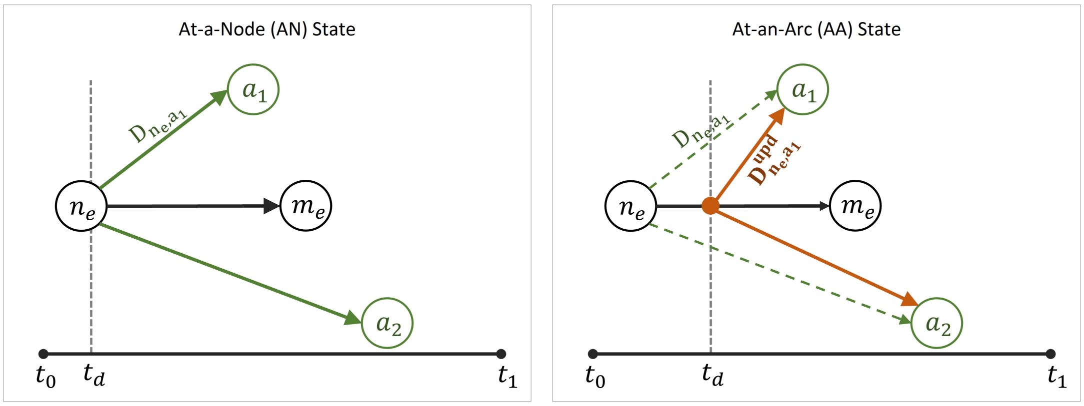


Figure 5.4: State of an entity at the time disruption occurs.

## 5.2 Revised Connection Networks After Disruption

Whenever a disruption occurs, some activities may have been completed. Obviously, we are interested in the remaining. Structure of CNs allows to quickly obtain states of entities and rerouting alternatives that are still feasible. At the time disruption occurs, $t_d \in [t_0, t_1]$, an entity is in one of the following states:

- At-a-Node (AN): has started an activity and has not finished its service yet
- At-an-Arc (AA): has completed an activity (or left the source) and currently on a connection arc towards another activity (or sink).

These two states are illustrated in Figure 5.4. Flight nodes $n_e$ and $m_e$ are on the scheduled path of the entity while $a_1$ and $a_2$ are alternative flights that the entity can be rerouted through. Horizontal line is a time-line representing the planning horizon. If there exist a node $i \in SN_e$ such that $t_d \in [S_i, C_i]$, we say that the state of entity $e$ is AN. We designate the current node with $n_e = i$. Otherwise, we state that the entity is in state AA and designate the current arc with $(n_e, m_e)$, where $n_e = \underset{i \in SN_e}{argmax}\{C_i : C_i < t_d\}$ and $m_e$ is the end node of the arc emanating from $n_e$ in $SG_e$.

For the first case (AN), we assume that the operation of the service at the current node cannot be altered. Recall that these activities may be flights or aircraft maintenances, and hence, it would not be easy to cancel or postpone an already started activity.

For the second case (AA), there are different possibilities: the entity may be idle or preparing for its next service (fueling, luggage loading etc.) at $Des_{n_e}$. In any case, we roll-back to node $n_e$ and check the feasibility of alternative arcs emanating from this node. Recall the feasibility rule for a connection between $i$ and $j$ used in CNGA. All arcs satisfy $B_j \geq A_i + D_i + D_{ij}$. However since $t_d > C_i = S_i + D_i$ and $S_i \geq A_i$, some arcs may be infeasible at $t_d$. Updated feasibility rule for arcs emanating from $n_e$ and incoming to $j$ where $j \neq m_e$ (alternative) is:

$$B_j \geq t_d + D^{upd}_{n_e j} \qquad , j \in N_e : (n_e, j) \in A_e, j \neq m_e$$

$D^{upd}_{n_e j}$ is the updated arc duration as illustrated in Figure 5.4 which depends on activity type:

- If the arc represents an idle time at $Des_{n_e}$, i.e. if $D_{n_e j} = 0$, $D^{upd}_{n_e j}$ is set to $0$ as well.
- If the arc represents a preparation particularly for its next scheduled service which is $m_e$, then $D^{upd}_{n_e j}$ is set to the original arc duration, $D_{n_e j}$.
- If the arc represents a preparation for its next (any) service, $D^{upd}_{n_e j}$ is set to the remaining time to complete the preparation, $D_{n_e j} - (t_d - C_{n_e})$.

We represent the state of entity $e$ at the moment of a disruption with a connection network, $\overline{G}_e = \left(\overline{N}_e, \overline{A}_e\right)$. This network holds the following information:

- Current position of the entity in the original CN.
- Possible rerouting opportunities within $[t_d, t_1]$ to transport the entity from its current location to its sink, $t_e$.

We also generate $\overline{SG}_e = \left(\overline{SN}_e, \overline{SA}_e\right)$ to represent the scheduled activities assigned to entity $e$ and that have not been operated yet.

We propose *Connection Network Revision Algorithm* (CNRA) to create $\overline{G}_e$ and $\overline{SG}_e$ at the moment of disruption. The algorithm starts by obtaining the current node or last

**Algorithm 3** Connection Network Revision Algorithm

**procedure** CNRA($e$)
    **if** $\exists n_e \in SN_e \ni t_d \in [S_i, C_i]$ **then**
        Set $s_e \leftarrow n_e$
    **else**
        Set $s_e \leftarrow \underset{i \in SN_e}{argmax}\{C_i : C_i < t_d\}$
        Set $m_e \leftarrow j \ni (s_e, j) \in SA_e$
        **for** each $j : (s_e, j) \in A_e \setminus SA_e$ **do**
            Calculate $D^{upd}_{s_e j}$
            **if** If $t_d + D^{upd}_{s_e j} > B_j$ **then**
                mark $(s_e, j)$ as *Infeasible*
            **end if**
        **end for**
    **end if**
    Update $RT_e = max\{C_{s_e}, t_d\}$
    Set $\overline{N}_e = \overline{SN}_e = \{s_e\}$ and $\overline{A}_e = \overline{SA}_e = \emptyset$
    SearchForward($s_e$)
**end procedure**

**Algorithm 4** Search Forward

**procedure** SEARCHFORWARD($i$)
    **if** $i = t_e$ **then**
        Stop
    **else**
        $N^{next} \leftarrow \{j \in N_e : (i, j) \in A_e$ and $(i, j)$ is not *Infeasible*$\}$
        **for** each $j \in N^{next}$ **do**
            $\overline{N}_e \leftarrow \overline{N}_e \cup \{j\}$
            $\overline{A}_e \leftarrow \overline{A}_e \cup \{(i, j)\}$
            **if** $j \in SN_e$ **then**
                $\overline{SN}_e \leftarrow \overline{SN}_e \cup \{j\}$
            **end if**
            **if** $(i, j) \in SA_e$ **then**
                $\overline{SA}_e \leftarrow \overline{SA}_e \cup \{(i, j)\}$
            **end if**
            SearchForward($j$)
        **end for**
    **end if**
**end procedure**

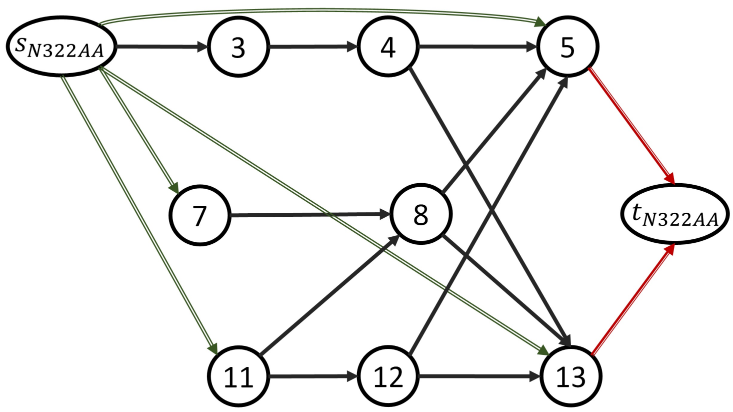


Figure 5.5: State of NN322AA if disruption occurs at 8:15.

visited node, and updates arc lengths emanating from this node. New source node is set to the current node (line 3 or 5) and ready time of the entity is updated as the first time that the entity will be available (line 14). *SearchForward* procedure inserts all nodes and arcs that are reachable from the new source node into the updated networks $\overline{G}_e$ and $\overline{SG}_e$.

For illustration, we consider the revised CN of aircraft in the example described in Section 5.1.6. Assume that a disruption occurs at 8:15. Note that the aircraft is operating flight 2 (AN) at this moment. We update the source node as flight 2 ($s_{N322AA} = 2$) and search forward from this node. Revised CN of this entity, $\overline{G}_{N322AA}$, is displayed in Figure 5.5. Note that this is a subset of the CN generated by CNGA in Figure 5.2, corresponding to the subset of rerouting alternatives which are still feasible.

Now assume that the disruption occurs at 9:50. The aircraft is on the connection arc $2 \rightarrow 3$ (AA). 20 minutes of the required turnaround time have passed and there is 10 minutes left for the scheduled departure of flight 3. We roll back to flight 2 and check feasibility of arcs emanating from this node. There are three alternative connections $2 \rightarrow 7$, $2 \rightarrow 11$, $2 \rightarrow 13$. Since there is enough slack time between the scheduled departures of flights 7, 11 and 13, we designate alternative connections as still feasible. Therefore, we obtain the same CN displayed in Figure 5.5.

Finally, assume that time of the disruption is 16:20. The aircraft is on the connection arc $4 \rightarrow 5$ (AA). We roll back to flight 4, set $s_{N322AA} = 4$, and check the feasibility

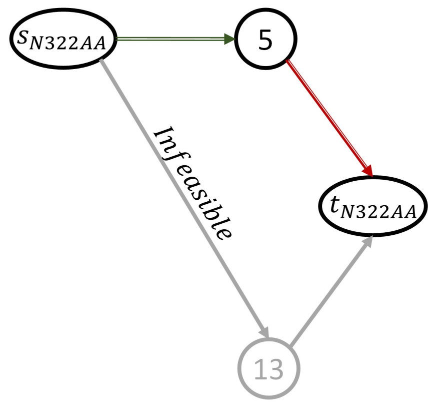


Figure 5.6: State of NN322AA if disruption occurs at 16:20.

of the alternative connection $4 \rightarrow 13$. Assume that if we decide to divert the aircraft, another 30 minutes of turnaround time is required. In other words, updated duration, $D^{upd}_{4,13}$, is 30 minutes. Assume also that upper bound on departure time of flight 13 is 16:45 (maximum allowable delay is 45 minutes). Since $t_d + D^{upd}_{4,13} =$ 16:50 $> B_5 = 16:45$ , we mark this connection arc as *Infeasible*. Resulting revised CN of the aircraft is displayed in Figure 5.6.

## 5.3 Isolation Heuristic

CNs generated by CNGA and updated by CNRA represent states of entities at the moment of a disruption, $t_d$. In addition to the scheduled paths, these CNs are capable of generating all possible rerouting alternatives. In other words, complete solution space is represented. Therefore one can optimize the problem using the revised CNs. In this section, we propose a heuristic approach that carefully limits problem size when dealing with large instances.

*Isolation Heuristic* (IH) exploits the fact that only a subset of the operations will be altered in the optimal recovery. Moreover, operations of only a subset of entities will be altered in the optimal solution. Remaining nodes will be visited by originally assigned entities and will be operated as planned. Objective of the proposed heuristic is to find the subset of entities and nodes, schedules of which would be changed if the instance were optimized over the complete solution space. If these entities and nodes can be isolated, it is possible to reach optimal solution over a smaller solution space.

We represent disruptions by designating directly affected nodes. Let $DN$ be the set of nodes that are directly affected from the disruption. A scheduled activity may be disrupted due to various reasons such as flight cancellation. In this case, we insert the related node in $DN$. There may also be a disruption on a connection arc. For instance, turnaround time of an aircraft may be increased due to problems in ground operations. In this case, we set $DN = \{i, j\}$. Finally, reason of the disruption may be related to an entity. A crew member may show up late for a flight or may not show up during the tracking horizon. Similarly, technical problems of an aircraft may prevent it from following its original path. In this final case, all nodes originally scheduled to the disrupted entity, say $e$, are designated as disrupted, and hence, we set $DN = SN_e$.

The algorithm expands through the CNs of entities based on a reference node set denoted by $IN$. At each *expansion step*, we select new entities which:

- are not included in the solution space yet, and
- are related to the current reference node set.

When we include these entities in our solution space, the reference node set expands as well by the inclusion of new related flight nodes. In order to illustrate the idea of the heuristic, suppose that flight $i$ is cancelled, and hence, we set $DN = \{i\}$. Furthermore, we initiate the reference node set, $IN = DN$. Nodes included in $IN$ also correspond to activities that will be included in the isolated solution space. If a flight is cancelled due to the weather conditions at the destination airport, the aircraft that is planned to operate flight $i$, say $e$, could be rerouted to another destination instead of waiting idle at the origin airport. Flights originally assigned to this aircraft, $\overline{SN}_e$, may be altered. Alternative flight nodes in its CN that are originally assigned to other aircraft may also be altered due to rerouting or swapping actions. We would set $IN = IN \cup \overline{N}_e$ to include all rerouting options of this aircraft in the solution space.

Schedules of these $|\overline{N}_e|$ flights may be altered in the solution. Therefore, crew members operating these flights need to be included in the solution space. In order to include rerouting options for these crew members, we also need to insert flight nodes in their CNs.

Passengers are severely affected from disruptions and recovery decisions. Therefore, we also need to consider passengers originally assigned to flights in $IN$ and corresponding reallocation alternatives. As the example illustrates, whenever new flight nodes are inserted in $IN$, new aircraft, crew members and passengers need to be considered. This process probably continues until all entities and nodes are included in the solution space since the operations of different entity types do not overlap.

Assume that we iterate in this manner by including new aircraft, crew members, passengers and new flight nodes at each step. Also assume that flight $j$ is inserted into $IN$ at $n^{th}$ step. Recall that $i$ is the cancelled flight, and hence, inserted into $IN$ at step 1. Step numbers at which these nodes are inserted into $IN$ expresses the level of relational distance of these nodes from the disruption. Note that this level is not only related with the geographical distance (between origin and destination) and time related relation (between the time windows), but also captures proximity information with respect to the existence in rerouting alternatives of the included entities.

We intuitively state that as the level of relational distance between a node and disruption increases, it becomes less likely that original schedule of this node will be altered in the optimal recovery. Moreover, if scheduled activities with a high level of relational distance are altered, *stability* of the solution will be low. IH starts from the disrupted node(s) and expands through the CNs of entities that are *close to* the disruption point so that good recovery actions are included in the isolated solution space and stability of the solution is controlled.

### 5.3.1 Control Parameters

In this section, parameters to control the expansion strategy, problem size and stability are explained.

$\alpha$ : main expansion strategy

$$\alpha = \begin{cases} \textit{Balanced} \\ \textit{Independent} \end{cases}$$

$K_t$ : number of expansion steps that will be executed for entity type $t$.

$\alpha$ determines the main strategy of the algorithm. If *Balanced* version is used, expansion steps of all entity types are based on a common reference node set. In this manner, recovery actions of all entity types are considered simultaneously. On the other hand, in *Independent* strategy, expansion steps of different entity types are executed independently as if we are dealing with dedicated recovery problems, such as aircraft recovery, crew recovery or passenger recovery problems. Note that integration will still be maintained at the Stabilization stage that will be executed at the end of expansion process. If $K_t$ is set to a large number, more entities of type $t$ will be included in the solution space, and vice versa.

$\beta_{tk}$ : entity selection strategy at step $k$ for type $t$ entities

$$\beta_{tk} = \begin{cases} \textit{Affected} \\ \textit{Alternative} \end{cases}$$

Consider the $k^{th}$ expansion step of entity type $t$ while the set of already included nodes is $IN$. If $\beta_{tk}$ is set to *Affected*, entities having at least one scheduled activity that is already included in $IN$ are selected. Note that these entities will be directly affected if operations in $IN$ are altered. If $\beta_{tk}$ is set to *Alternative*, the algorithm will be searching for entities that are not only directly affected, but also are candidates to be rerouted through the nodes in $IN$. To illustrate, consider an aircraft that is idle during the recovery horizon, and hence, is not directly affected. However, this aircraft may generate many alternative solutions. For instance, it may operate flights of another aircraft experiencing mechanical problems and also may enrich swap opportunities. In *Alternative* strategy, entities having at least one node, not necessarily scheduled, in $IN$ are selected.

$n_{tk}$ : limit on the number of type $t$ entities at step $k$ if entity selection strategy is *Alternative*

$\theta_{tk}$ : sorting strategy for selected type $t$ entities at step $k$

$$\theta_{tk} = \begin{cases} \textit{Idleness} \\ \textit{Relevance} \end{cases}$$

When we use *Alternative* strategy for selecting new entities, number of candidate entities may be enormously high. In order to limit problem size, we limit the number of entities of type $t$ to be included at step $k$ by $n_{tk}$. In other words, we only include a subset of possible alternatives. $\theta_{tk}$ is the criterion to sort selected entities, and we select the first $n_{tk}$ entities from this sorted list. In *Idleness* strategy, selected entities are sorted with descending order of their idle times within $[t_d, t_1]$. Since these entities represent alternative resources to recover disrupted nodes, it is more preferable that they have idle time. On the other hand, in *Relevance* strategy, entities are sorted with descending number of nodes in their CNs that have already been included in $IN$. More relevant entities may be used to recover more nodes in the reduced solution space.

$\gamma_{tk}$ : Network inclusion strategy for type $t$ entities at step $k$

$$\gamma_{tk} = \begin{cases} \textit{Scheduled} \\ \textit{Complete} \end{cases}$$

If an entity is included with *Complete* strategy, $\overline{G}_e$ will be included in the isolated solution space. Therefore, all rerouting alternatives will be available. In *Scheduled* option, the entity is included with $\overline{SG}_e$, and hence, only retiming and ferrying/deadheading/spilling decisions will be available.

### 5.3.2 Algorithm

IH involves three main steps: (1) Initialization, (2) Expansion and (3) Stabilization, and main heuristic is displayed in Algorithm 5.

**Algorithm 5** Isolation Heuristic

```
1: procedure ISOLATION
2:     Initialization
3:     Expansion
4:     Stabilization
5: end procedure
```

In initialization step given in Algorithm 6, starting point of the heuristic is determined. Expansion process is based on a common set of nodes ($IN$) in *Balanced* strategy, and on discrete sets for each entity type ($IN_t$) in *Independent* strategy. In this step, we simply include disrupted nodes, $DN$, in the corresponding reference sets. In order to keep track of entities included in the solution space, we also initiate empty sets of entities for each type, i.e. $IE_t = \emptyset, t \in T$.

**Algorithm 6** Initialization

**procedure** INITIALIZATION<br>
    $IN = DN$<br>
    $IN_t = DN, t \in T$<br>
    $IE_t = \emptyset, t \in T$<br>
**end procedure**

Pseudo codes of *Balanced* and *Independent* strategies are presented in Algorithm 7. In *Balanced* strategy, $K = max\{K_t\}$ expansion steps are carried out. At each step, we check entity types for which an expansion step will be carried out (line 5). Reference node set for the expansion process is $IN$.

In *Independent* strategy, expansion steps of each entity type are carried out based on their own reference node sets, $IN_t$. We simply search for recovery options for each entity type separately. Although the idea seems contrary to integration, dependencies among entity types will still be constructed in Stabilization step and the solution space will be integrated.

**Algorithm 7** Expansion

1: **procedure** EXPANSION<br>
2:     **if** $\alpha =$*Balanced* **then**<br>
3:         **for** $k = 1$ to $K = max\{K_t\}$ **do**<br>
4:             **for all** $t \in T$ **do**<br>
5:                 **if** $k \leq K_t$ **then**<br>
6:                     Expand($k, t, IN$)<br>
7:                 **end if**<br>
8:             **end for**<br>
9:         **end for**<br>
10:     **end if**<br>
11:     **if** $\alpha =$ *Independent* **then**<br>
12:         **for all** $t \in T$ **do**<br>
13:             **for** $k = 1$ to $K_t$ **do**<br>
14:                 Expand($k, t, IN_t$)<br>
15:             **end for**<br>
16:         **end for**<br>
17:     **end if**<br>
18: **end procedure**

Main expansion step for entity type $t$ at step $k$ is presented in Algorithm 8. Expansion will be based on the reference node set $B$ ($IN$ for *Balanced* and $IN_t$ for *Independent*). The procedure first creates set $\varepsilon$ which is the set of entities of type $t$ that will be included in the current step. Then, each of these entities are included in the solution space in line 3. According to the value of parameter $\gamma_{tk}$, we either include the entity with $\overline{SG}_e$ (line 5), or with $\overline{G}_e$ (line 8). In the latter one, we include all rerouting options of the entity in solution space, while in the prior one only retiming decisions will be included. Expansion is reflected to the reference sets in lines 10 and 11.

**Algorithm 8** Expand

1: **procedure** EXPAND($k, t, B$)
2: $\quad$ $\varepsilon =$ SelectEntities($t, B, \beta_{tk}, \theta_{tk}, n_{tk}$)
3: $\quad$ $IE_t \leftarrow IE_t \cup \varepsilon$
4: $\quad$ **if** $\gamma_{tk} =$*Scheduled* **then**
5: $\quad\quad$ $IG_e = \overline{SG}_e, \forall e \in \varepsilon$
6: $\quad$ **end if**
7: $\quad$ **if** $\gamma_{tk} =$*Complete* **then**
8: $\quad\quad$ $IG_e = \overline{G}_e, \forall e \in \varepsilon$
9: $\quad$ **end if**
10: $\quad$ $IN_t \leftarrow IN_t \cup \left( \bigcup_{e \in \varepsilon} IN_e \right)$
11: $\quad$ $IN \leftarrow IN \cup \left( \bigcup_{e \in \varepsilon} IN_e \right)$
12: **end procedure**

Entities that will be included at an expansion step are selected with Select Entities procedure presented in Algorithm 9. Selection process is based on reference node set $B$. In line 2, set of directly affected, $\varepsilon_1$, entities are selected. Note that these entities have at least one scheduled node which is already included in the reference set. If cardinality of $\varepsilon_1$ is greater than or equal to the upper bound on the number of entities that may be included at this step ($n$), or if the entity selection strategy *Affected*, the procedure returns $\varepsilon_1$. Otherwise (if entity selection strategy is *Alternative* and $|\varepsilon| < n$), we search for entities which have no scheduled node in the reference set, but may be rerouted to serve at least one of the included nodes. Set of these alternative entities, $\varepsilon_2$, is obtained in line 6. If $|\varepsilon_1| + |\varepsilon_2|$ does not exceed the limit on the number of entities to include at this step, $n$, $\varepsilon_1 \cup \varepsilon_2$ is returned. In highly connected networks, size of the union set may be greater, providing many alternatives to cover affected nodes. In these cases, we only include $n$ entities to control the problem size and stability. Criterion to select the best $n$ among $|\varepsilon_1| + |\varepsilon_2|$ entities depends on the value of $\gamma$. If $\gamma$ =*Idleness*, we select the most idle entities. On the other hand, when $\gamma$ =*Relevance*, we select entities having the greatest number of nodes in common with the reference node set, and hence, can serve the greatest number nodes included in the solution space.

**Algorithm 9** Select Entities

**procedure** SELECTENTITIES($t, B, \beta, \theta, n$)
    $\varepsilon_1 = \left\{e \in E_t \setminus IE_t : \overline{SN}_e \cap B \neq \emptyset\right\}$
    **if** $\beta =$ *Affected* or $|\varepsilon_1| \geq n$ **then**
        **return** $\varepsilon_1$
    **else** ▷ if $\beta =$ *Alternative* and $|\varepsilon_1| < n$
        $\varepsilon_2 = \left\{e \in E_t \setminus IE_t : \left(\overline{N}_e \setminus \overline{SN}_e\right) \cap B \neq \emptyset\right\}$
        **if** $|\varepsilon_1| + |\varepsilon_2| \leq n$ **then**
            **return** $\varepsilon_1 \cup \varepsilon_2$
        **else**
            Sort entities in $\varepsilon_2$ with respect to the sorting criterion, $\theta$
            $\overline{\varepsilon_2}$ = first $n - |\varepsilon_1|$ entities in $\varepsilon_2$
            **return** $\varepsilon_1 \cup \overline{\varepsilon_2}$
        **end if**
    **end if**
**end procedure**

Stabilization step is carried out after the execution of all expansion steps, and it is crucial to construct feasibility of the isolated solution space. As we stop expanding, entities scheduled to an included node may be absent in the set of included entities. For instance, consider an activity node $i$ that is included in the final expansion step of crew members. It is very likely that scheduled aircraft of this flight is not included in the solution space. If the algorithm is terminated at this state, there is a possibility that this flight needs to be cancelled due to insufficient number of aircraft in the solution space resulting in an unnecessary disturbance. In order to include these missing entities, Stabilization procedure, which is presented in Algorithm 10, is carried out before finalizing the solution space. Set of missing entities, $\varepsilon$, is obtained by the operation in line 2. It is crucial not to include any additional activity nodes since we do not want to expand any more. Therefore, missing entities are not included with their entire scheduled networks. For instance, suppose that a missing entity has the scheduled path 1-2-3-4 and only node 3 is present in $IN$. In this case, we insert entity $\overline{e}$ with $IG_{\overline{e}} = (IN_{\overline{e}}, IA_{\overline{e}})$, where $IN_{\overline{e}} = \{s_{\overline{e}}, 3, t_{\overline{e}}\}$ and $IA_{\overline{e}} = \{(s_{\overline{e}}, 3), (3, t_{\overline{e}})\}$. Entity inclusion process is carried out by the subprocedure presented in Algorithm 11. We also set, $RT_{\overline{e}} = S_2 + D_2 + D_{2,3}$ to guarantee that isolated node 2 will be operated as scheduled. Similarly, we set $LAT_{\overline{e}} = S_4 - D_{3,4}$ to provide that the entity will be ready to serve node 4 on time. In this manner we guarantee the stability of the nodes that are not included in the solution space and they will be operated as planned in the recovery horizon.

**Algorithm 10** Stabilization

**procedure** STABILIZATION
    $\varepsilon = \left\{ e \in E \setminus \left( \bigcup_{t \in T} IE_t \right) : \overline{SN}_e \cap IN \neq \emptyset \right\}$
    **for all** $e \in \varepsilon$ **do**
        Let $\overline{SN}_e = \{v_1, v_2, ..., v_{n_e}\} \ni (v_i, v_i + 1) \in \overline{SA}_e$ for $i = 1, ..., n_e - 1$
        $N^{curr} = \emptyset$
        **for** $i = 1$ to $n$ **do**
            **if** $v_i \in IN$ **then**
                $N^{curr} \leftarrow N^{curr} \cup \{v_i\}$
            **else**
                IncludeEntity($e, N^{curr}$)
            **end if**
        **end for**
        **if** $N^{curr} \neq \emptyset$ **then**
            IncludeEntity($e, N^{curr}$)
        **end if**
    **end for**
**end procedure**

A more complicated case may be observed if nodes 1,2 and 4 are present in $IN$. In order not to include node 3 in the solution space and preserve its stability, we split the entity into two entities, $\overline{e}_1$ and $\overline{e}_2$, with: $IN_{\overline{e}_1} = \{s_{\overline{e}_1}, 1, 2, t_{\overline{e}_1}\}$ and $IA_{\overline{e}_1} = \{(s_{\overline{e}_1}, 1), (1, 2), (2, t_{\overline{e}_1})\}$; and $IN_{\overline{e}_2} = \{s_{\overline{e}_2}, 4, t_{\overline{e}_2}\}$ and $IA_{\overline{e}_2} = \{(s_{\overline{e}_2}, 4), (4, t_{\overline{e}_2})\}$.

### 5.4 Experimentation

We have experimented the proposed approach with integrated airline recovery problem. Three types of entities are integrated: aircraft, passengers and crew teams. Flight data and aircraft routings are extracted from data provided by Bureau of Transportation Statistics (http://www.transtats.bts.gov/DataIndex.asp). Crew and passenger routings are generated randomly while maintaining feasibility of the schedules.

**Algorithm 11** Include Entity

**procedure** INCLUDEENTITY($e, N^{curr}$)
    Let $N^{curr} = \{v_l, v_{l+1}, ..., v_m\}$
    Create entity $\overline{e}$
    **if** $l = 1$ **then**
        $RT_{\overline{e}} = RT_e$
    **else**
        $RT_{\overline{e}} = S_{v_{l-1}} + D_{v_{l-1}} + D_{v_{l-1},v_l}$
    **end if**
    **if** $m = n_e$ **then**
        $LAT_{\overline{e}} = LAT_e$
    **else**
        $LAT_{\overline{e}} = S_{v_{m+1}} - D_{v_m,v_{m+1}}$
    **end if**
    $IN_{\overline{e}} = \{s_{\overline{e}}, t_{\overline{e}}\} \cup N^{curr}$
    $IA_{\overline{e}} = \{(s_{\overline{e}}, v_l), (v_m, y_{\overline{e}})\}$
    **for** $j = 1$ to $m - 1$ **do**
        $IA_{\overline{e}} \leftarrow IA_{\overline{e}} \cup \{(v_j, v_{j+1})\}$
    **end for**
**end procedure**

### 5.4.1 Instance Creation and Settings

Data of five days of operations is used. Disruption scenarios are generated by inserting three types of disruptions on each day: (1) minor delay (departure time of a flight is delayed by 60 minutes), (2) major delay (departure time of a flight is delayed by 120 minutes), (3) cancellation (a flight is cancelled). Therefore, 15 instances are created. Disrupted flights are selected randomly. Number of entities of each type, and average number of nodes and arcs in the CNs of these entities are tabulated in Table 5.2.

Table 5.2: Network statistics

| Entity type | Nb. of entities | Average nb. of nodes | Average nb. of arcs |
|---|---|---|---|
| Aircraft | 341 | 22.56 | 40.24 |
| Itinerary | 5243 | 4.15 | 4.41 |
| Crew Team | 482 | 22.15 | 35.68 |

Recall that proposed approach is independent of the solution methodology. In order to test the performance of generated solution spaces with different methodologies, we have tried to solve instances with four variants of the mathematical model that is presented in Chapter 4. Models used in the experimentation are listed in Table 5.3. In models with ✓in cruise speed control option column, speeding up flights is allowed. However, mixed integer programming (MIP) models needs to be reformulated as conic quadratic mixed integer programming (CQMIP) models to deal with the nonlinearity in fuel cost function. Therefore, FDA- and AD- are MIP models while FDA+ and AD+ are CQMIP models. Calculating the realized passenger delay cost is challenging in integrated recovery problems. Most of the studies in the literature use an approximation approach, which will be called *flight delay approximation*. On the other hand, with *actual delay* method, realized delays are calculated. If the approximation is used, each passenger itinerary is modeled as a single entity (passengers in the same itinerary are aggregated). Otherwise, each passenger is explicitly modeled as an entity, and hence, the problem size is much greater. All models use a common objective function of minimizing total disruption and recovery costs which consist of passenger delay cost, cancellation cost, ferrying cost, deadheading cost and spill cost. In addition to these, FDA+ and AD+ have an additional cost term related with the additional fuel cost of speeding up flights.

Table 5.3: Mathematical models

| Name | Cruise speed control option | Passenger delay cost |
|---|---|---|
| FDA- | | Flight delay approximation |
| AD- | | Actual delay |
| FDA+ | ✓ | Flight delay approximation |
| AD+ | ✓ | Actual delay |

Effect of control parameters are also tested. Both of the main expansion strategies ($\alpha$) are experimented: *Balanced* and *Independent*. Similarly, two of the entity sorting strategies ($\theta_{tk}$) are involved in the experimentation: *Idleness* and *Relevance*. In order to limit the number of combinations, we have used a common sorting strategy in all expansion steps and for all entity types, i.e., $\theta = \theta_{tk}, \forall t, k$. Similarly, two limits on the number of selected entities are applied: $n = n_{tk}, \forall t, k$ is set to 10 or 50. Network inclusion strategy ($\gamma_{tk}$) is set to *Complete* for all expansion steps of aircraft and crew teams. On the other hand, we have experimented *Scheduled* and *Complete* versions for passengers. Recall that, only the scheduled network is included in the prior one, while passenger reallocation is allowed in the latter case. Combinations of number of expansion steps ($K_t$) and entity selection strategies ($\beta_{tk}$) to be tested are generated together depending on our preliminary runs. Same number of expansion steps are applied for each entity type, i.e., $K = K_t, \forall t$. In all expansion steps, $\beta_{tk}$ is set to *Affected* for passengers, since they are not operating entities. Moreover, same parameters are used for aircraft and crew teams. Generated five combinations are listed in Table 5.4, where $a$ is an abbreviation for aircraft. In total, 80 parameter combinations are involved in the computational study. Each combination is tested with four different mathematical models on each of the 15 instances, which adds up to 4800 instances.

Table 5.4: Experimented $K_t$ and $\beta_{tk}$ combinations

| Combination | $K$ | $\beta_{a1}$ | $\beta_{a2}$ | $\beta_{a3}$ |
|---|---|---|---|---|
| Alt | 1 | Alternative | | |
| Alt-Aff | 2 | Alternative | Affected | |
| Alt-Alt | 2 | Alternative | Alternative | |
| Alt-Alt-Aff | 3 | Alternative | Alternative | Affected |
| Alt-Alt-Alt | 3 | Alternative | Alternative | Alternative |

### 5.4.2 Problem Size

Effects of control parameters of IH on problem size are analyzed in this section. In Figure 5.7, effects of expansion strategy, entity selection strategy and number of expansion steps on the number of entities included in the solution space are illustrated. An initial analysis is the insignificance of entity selection parameter ($\beta_{tk}$) with *Balanced* strategy. This is explained with the huge number of affected entities from included nodes due to high connectivity of airline networks. Recall that if number of affected entities exceeds the limit ($n_{tk}$), no alternative entities will be included in the model. On the other hand, since *Independent* strategy expands based on a smaller reference set, we can observe the effect of this parameter. With *Balanced* strategy, effect of the number of expansion steps is dramatic between 1-step and 2-steps, while it is minor between 2-steps and 3-steps. On the other hand, number of entities and number of expansion steps have almost a linear relationship with *Independent* strategy.

Another important factor on problem size is size of the included CNs. Main factor

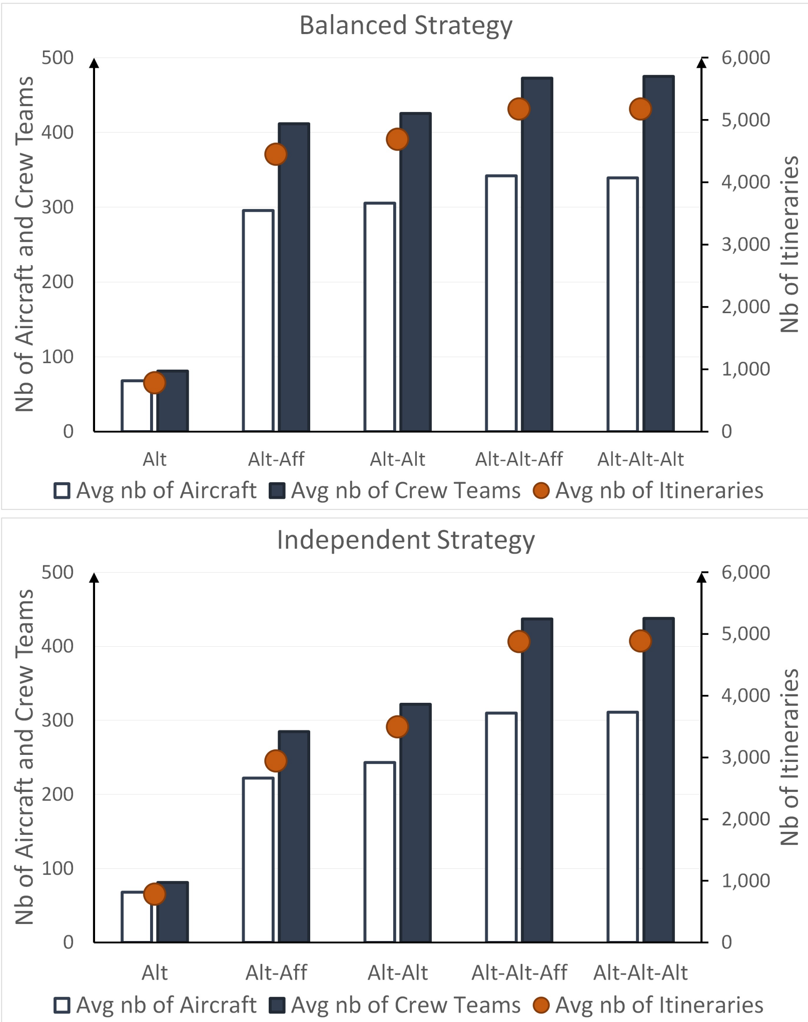


Figure 5.7: Statistics of the included entities in the solution space

affecting the size of generated CNs is the entity inclusion strategy, $\gamma_{tk}$. In this experimentation, we have only experimented $\gamma_{tk}$ on passengers. Average number of arcs in the CN of a passenger is about 3.6895 with *Scheduled* strategy, while it increases to 3.9145 with *Complete* strategy. Difference may seem negligible. However, number of passenger itineraries is much greater than the number of aircraft and crew teams, and hence, the effect on problem size is significant. Moreover, since each passenger is modeled explicitly in *AD-* and *AD+* models, the effect becomes even more important.

On the other hand, number of included entities also affects the average size of CNs, since only scheduled networks are included during the *Stabilization* step. Therefore, we observe similar effects of parameters on network sizes as displayed in Figure 5.8. Networks statistics of aircraft is illustrated only, but crew members have similar statistics as well.

Effects of parameters on CPU time of the heuristic are illustrated on Figure 5.9. As expected, number of expansion steps significantly increases the run time. Change in entity selection strategy at the final expansion step ($\beta_{tk}$) has minor effect on network generation times. On the other hand, entity inclusion strategy of passengers ($\gamma_{tk}$) is significant with *Balanced* strategy: *Scheduled* strategy results in smaller running times than *Complete*. Finally, except for 1-step settings, *Independent* strategy generates the networks faster than *Balanced* strategy.

### 5.4.3 Solutions

After isolated networks for each of the 15 instances are generated by Isolation Heuristic, problems are solved with four of the mathematical models. Since quick decisions are required in airline disruption management problems, solution times are limited with five minutes. In order to compare the quality of generated solutions, instances are optimized over its entire solution space (using CNs revised by CNRA) within five minutes again. Moreover, entire solution spaces are solved without time limits in order to achieve optimal disruption and recovery costs. Analyses are carried out for each mathematical model separately. Since effect of the limit on the number of entities included at each step ($n_{tk}$) is insignificant, this parameter has been excluded from the analyses. Moreover, only settings with good solution qualities are reported

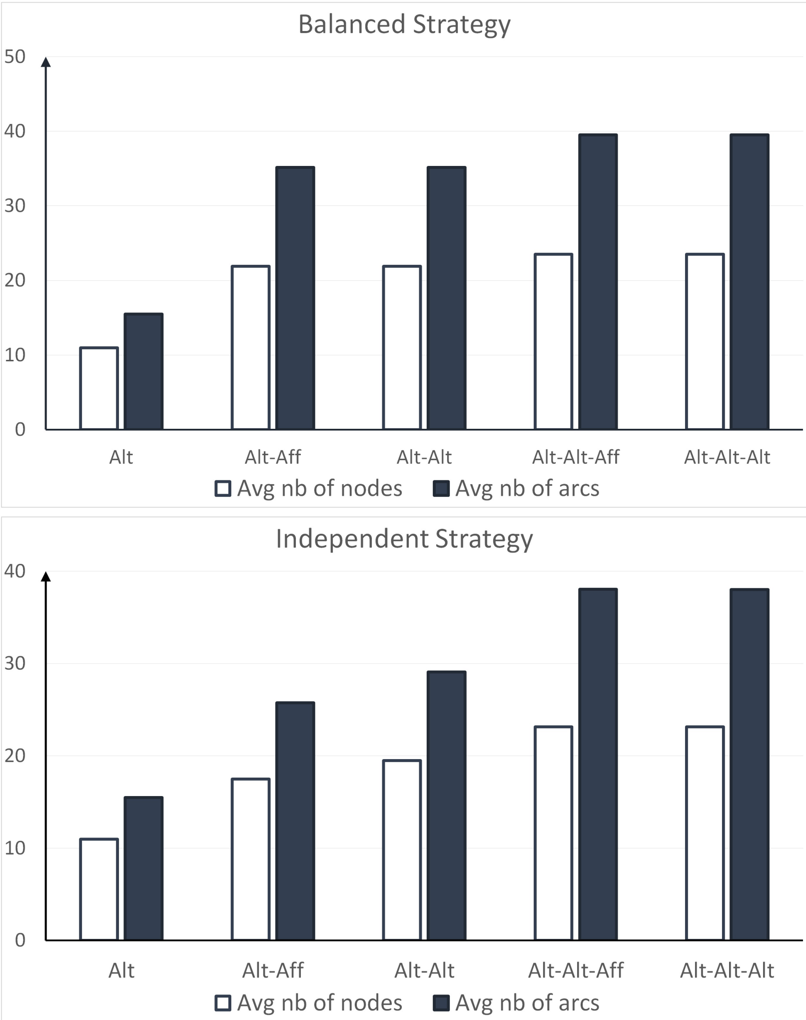


Figure 5.8: Statistics of the isolated aircraft networks

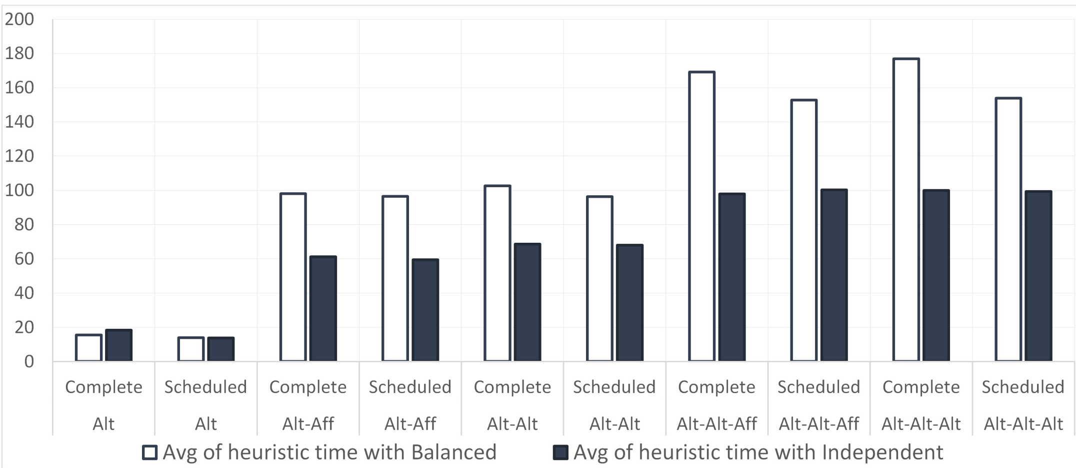


Figure 5.9: CPU time of Isolation Heuristic

while the complete tables of solution qualities are given in Appendix A.

#### 5.4.3.1 Solutions with FDA-

FDA- is a MIP formulation with passenger aggregation, and hence, is the least complex one. All instances are solved to optimality within five minutes (average solution time is 145.30 seconds) over the entire solution space. Effects of IH parameters are tabulated in Table 5.5. Columns nb ins., nb feas., nb 10% and nb opt. represents the number of instances, number of feasible solutions obtained, nb of solutions within 10% relative gap with respect to the optimal solution, and number of optimal solutions obtained, respectively. Relative gap is calculated by dividing the difference between the objective value obtained over the reduced solution space and optimal cost by the optimal cost. Final two columns of the table represents average and maximum of the observed relative gaps.

Table 5.5: FDA- Solutions

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt-Aff | Com. | Idl. | 30 | **30** | 16 | 26 | 3.13% | 12.04% |
| | | | Rel. | 30 | **30** | 18 | **30** | 1.97% | 7.97% |
| | | Sch. | Rel. | 30 | **30** | 12 | 23 | 4.49% | 14.41% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 16 | **30** | 2.35% | 9.10% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.39% | 5.04% |
| | | Sch. | Rel. | 30 | **30** | 12 | **30** | 3.24% | 8.00% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | **30** | 22 | **30** | 1.26% | 6.18% |
| | | | Rel. | 30 | **30** | 22 | 29 | 1.27% | 11.24% |
| | | Sch. | Rel. | 30 | **30** | 28 | **30** | 0.23% | 3.51% |
| | Alt-Alt-Alt | Com. | Rel. | 30 | **30** | 20 | **30** | 1.38% | 8.23% |
| | | Sch. | Rel. | 30 | **30** | 28 | **30** | 0.23% | 3.51% |
| Ind. | Alt-Alt | Com. | Rel. | 30 | **30** | 11 | **30** | 4.14% | 9.87% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 16 | 22 | 6.15% | 39.87% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.40% | 6.91% |
| | | Sch. | Idl. | 30 | **30** | 15 | 27 | 3.08% | 13.22% |
| | | | Rel. | 30 | **30** | 16 | 29 | 2.49% | 11.12% |

All 1-step settings resulted in inferior solutions. Therefore, we can state that single expansion step results in insufficient number of alternative actions, i.e., too small solution space. On the other hand, some of the 3-step settings end up with positive gaps within five minutes. Both 2-step settings result in very small gaps with *Balanced* strategy. We can observe a significant improvement in solution quality when the passengers are included with their *Complete* networks (instead of *Scheduled*). In other words, when reallocation of passengers to alternative flights is allowed, reduction in the number of spilled passengers significantly affects total costs. Since, 3-step settings work result in greater solution spaces, solution quality will be better provided that optimal solution is obtained within the time limit.

Among *Independent* settings, including the *Affected* entities at the final expansion step results in inferior solutions in almost all instances. Moreover, 2-step settings seem to generate too small solution spaces. All 3-step settings provide good solutions. Finally, we observe that using *Relevance* criterion for sorting the entities outperforms *Idleness* criterion. In other words, generating more connections with the expanded activity nodes is more important than the idle time of the entity.

#### 5.4.3.2 Solutions with AD-

This formulation is again a MIP model, but this time each passenger is modeled explicitly resulting in a much greater problem size. Only few of the instances could be optimized over the entire space within five minutes. Optimal solutions are obtained with an average solution time of 1916.47 seconds (about half an hour). Parameter settings that perform well with this model are tabulated in Table 5.6. 3-step settings of *Balanced* strategy failed to achieve good solutions within five minutes due to large size of the solution space. Moreover, difference between *Complete* and *Scheduled* strategies become more significant as more emphasis is placed on passenger delay costs. *Relevance* criterion for sorting entities again performs slightly better than the *Idleness* criterion. Finally, we observe similar outcomes with *Independent* strategy.

Table 5.6: Quality of solutions with different parameter settings in AD- instances

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt-Aff | Com. | Idl. | 30 | **30** | 15 | 24 | 3.21% | 11.63% |
| | | | Rel. | 30 | **30** | 15 | 28 | 2.61% | 10.46% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 17 | 26 | 2.19% | 12.24% |
| | | | Rel. | 30 | **30** | 21 | **30** | 1.22% | 7.72% |
| Ind. | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 13 | 28 | 3.70% | 12.87% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.66% | 7.70% |
| | | Sch. | Idl. | 30 | **30** | 16 | 21 | 5.55% | 17.99% |
| | | | Rel. | 30 | **30** | 14 | 29 | 3.73% | 14.35% |

#### 5.4.3.3 Solutions with FDA+

This formulation is a CQMIP model with passenger integration. None of the complete versions of instances are optimized within five minutes. Average solution time required to optimize over the entire solution spaces is 4542.77 seconds (about 75 minutes). Settings with best solution qualities are listed in Table 5.7. Similar conclusions with the previous models can be made for *Balanced* strategy. However, as the complexity of the problem increases 2-step settings perform better than 3-step settings with *Independent* strategy, and become more important.

Table 5.7: Quality of solutions with different parameter settings in FDA+ instances

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt-Aff | Com. | Idl. | 30 | **30** | 3 | 15 | 9.48% | 17.44% |
| | | | Rel. | 30 | **30** | 10 | **30** | 4.69% | 9.72% |
| | | Sch. | Idl. | 30 | **30** | 1 | 11 | 13.89% | 26.69% |
| | | | Rel. | 30 | **30** | 1 | 13 | 10.70% | 23.83% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 8 | 28 | 3.38% | 10.33% |
| | | | Rel. | 30 | **30** | 17 | **30** | 2.10% | 8.82% |
| | | Sch. | Idl. | 30 | **30** | 6 | 26 | 5.43% | 14.65% |
| | | | Rel. | 30 | **30** | 7 | 27 | 5.82% | 14.65% |
| Ind | Alt-Aff | Com. | Rel. | 30 | **30** | 12 | 17 | 9.38% | 31.10% |
| | Alt-Alt | Com. | Rel. | 30 | **30** | 12 | 28 | 3.41% | 10.60% |
| | | Sch. | Idl. | 30 | **30** | 1 | 14 | 11.02% | 20.52% |
| | | | Rel. | 30 | **30** | 2 | 22 | 6.98% | 12.44% |

#### 5.4.3.4 Solutions with AD+

This is the most difficult formulation which is complex due to the existence of conic quadratic constraints and has huge number of constraints since each passenger is modeled explicitly. Again instances could not be optimized over the entire solution space within five minutes, while the optimal solutions are achieved with an average solution time of 11503.83 seconds (about 3 hours). Best settings for this formulation are tabulated in Table 5.8. First important conclusion is that including *Scheduled* networks of passengers performs better than including the *Complete* networks with *Balanced* strategy. The reason of this result is that increased complexity prevents *Complete* strategy to reach optimal solutions within five minutes. On the other hand, *Independent* strategy manages to optimize over the *Complete* networks of passengers.

Table 5.8: Quality of solutions with different parameter settings in AD+ instances

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt-Aff | Sch. | Idl. | 30 | **30** | 0 | 8 | 14.10% | 30.15% |
| | | | Rel. | 30 | **30** | 4 | 12 | 12.74% | 27.52% |
| | Alt-Alt | Sch. | Idl. | 30 | **30** | 2 | 24 | 7.68% | 18.46% |
| | | | Rel. | 30 | **30** | 2 | 27 | 5.51% | 14.63% |
| Bal. | Alt-Alt | Com. | Idl. | 30 | **30** | 1 | 15 | 12.01% | 27.12% |
| | | | Rel. | 30 | **30** | 3 | 19 | 7.74% | 14.22% |
| | | Sch. | Idl. | 30 | **30** | 0 | 6 | 19.51% | 43.12% |
| | | | Rel. | 30 | **30** | 1 | 11 | 17.61% | 44.56% |

#### 5.4.3.5 Overall Strategy and Performance

Two dominated settings are observed during the experimentation. Firstly, 1-step settings do not manage to find good solutions. Running time of the heuristic and mathematical models are quite small with these settings, however, it is obvious that reduced solution spaces are too small. Secondly, we observe that *Relevance* strategy outperforms *Idleness* in almost all instances. In other words, entities which can recover more activities in the included nodes are more valuable than those with greater idle time. Intuitively, this may be explained by huge geographical area that flights are operated on. For instance an idle aircraft may be too far away from the disrupted nodes.

The general notion of the approach is to generate a solution space as large as possible that can provide real time solutions. Effects of control parameters on tractability of the instances are straightforward as displayed in Figure 5.10. Among these parameters, the effect of number of expansion steps is the most significant. Therefore, it is not very difficult to identify the suitable value for number of expansion steps. In this experimentation, we observe that two expansion steps perform well for *Balanced* strategy. With *Independent* settings, three expansion steps are better for FDA- and AD- instances, while two step settings perform better with more complex mathematical models. Finally, we can state that setting $\beta_{tk}$ to *Affected* to maintain tractability reduces the solution quality significantly (especially with *Independent* strategy).

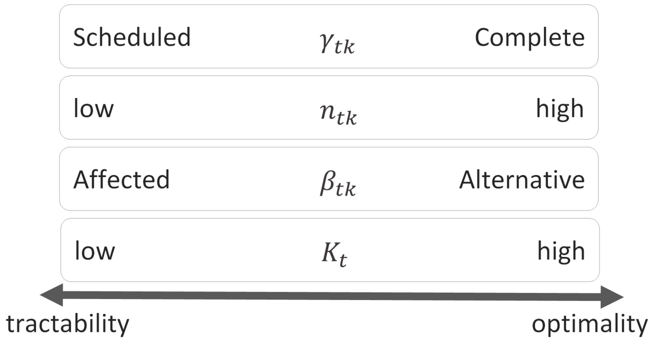


Figure 5.10: Effect of control parameters on tractability and optimality

Therefore, we propose to set this parameter to *Alternative* to provide enough rerouting and swapping alternatives, while maintaining the tractability of the problems with the remaining parameters.

Using these observations, we propose to identify the number of expansion steps initially, set entity selection strategy ($\beta_{tk}$) to *Affected*, set entity sorting strategy ($\theta_{tk}$) to *Relevance* and parallelize *Complete* and *Scheduled* options of entity inclusion strategy ($\gamma_{tk}$), for integrated airline recovery problems. Finally, we propose to parallelize on *Balanced* and *Independent* strategies, as they work in a completely different manner. Average performance of the heuristic with this approach is illustrated in Figure 5.11.

## 5.5 Conclusion

Recently, there is an increasing effort in integrating different entity types in airline disruption management problems to achieve a global optimum in contrary to sequential approaches. However, due to increased problem size and limited solution time, integrated recovery problem is challenging. Approaches in the literature fail to optimize complex mathematical models of huge transportation networks in short solution times. Moreover, to the best of our knowledge there is a lack of heuristic approaches in the literature that provides good and fast recovery actions. In this approach, we propose a practical solution procedure based on an alternative connection network representation. Different entity types can easily be integrated with connection networks. Moreover, these networks help to obtain states of entities and generate the

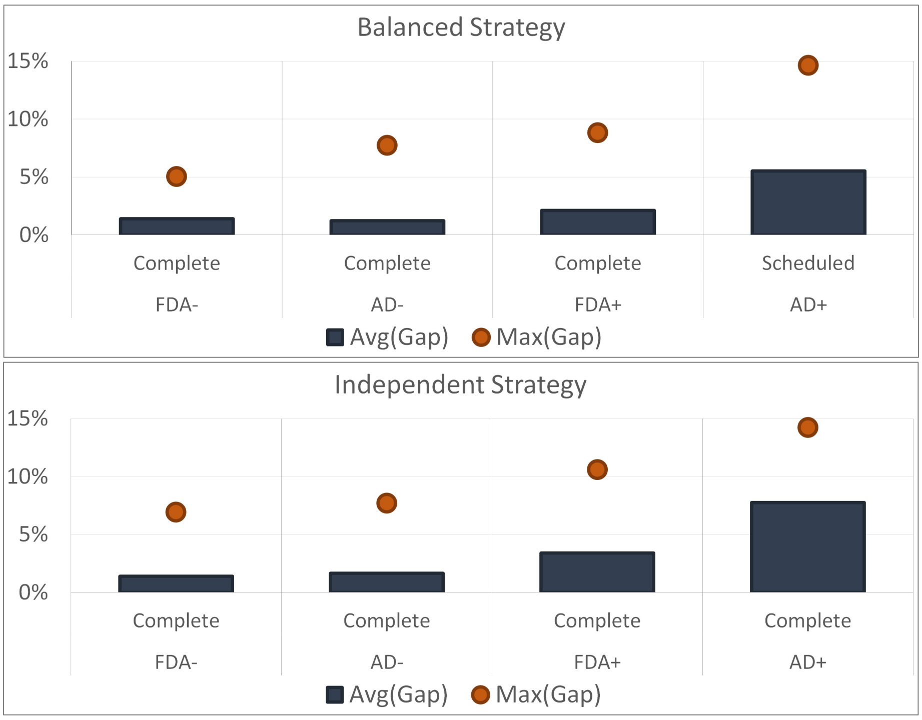


Figure 5.11: Average performance of the proposed solution procedure

entire solution space whenever a disruption occurs in a very short time.

For huge transportation networks and complex formulations, we propose Isolation Heuristic to cleverly isolate the solution space. The heuristic is based on the fact that scheduling decisions of only a subset of entities and operations will be altered in the optimal solution. Remaining activities will be operated as planned. Main challenge is to identify such activities and entities. In order to achieve this goal, proposed approach uses a relational distance levels of entities from disruption. These levels are based on the relationships of CNs of entities. The algorithm selects entities and activities that are *close to* the point of disruption. Moreover, we propose control parameters to balance the tradeoff between the tractability of the problems and quality of the solutions. An important property of the proposed approach is that it is independent from the solution methodology. The approach only reduces the problem size considering recovery actions and generates the representation of the smaller solution space.

We have tested the practicality of our approach using a large sized airline network of a major U.S. airline. Four different optimization models with different complexities are used to solve instances over the entire solution space and reduced solution spaces generated by the proposed heuristic. Most of the instances can be solved to optimality with less complex formulations within five minutes. Solutions provided by isolation heuristic have negligible gaps from the optimal objective values for these instances. On the other hand, the heuristic becomes more valuable as the complexity of the solution methodology increases. For two sets of complex instances that are optimized in average solution times of 75 minutes and three hours, proposed approach provides solutions with about 2.76% and 6.63% more costs within five minutes. Note that the complex optimization models fail to provide feasible solutions to the majority of instances over the entire solution space when solution time is limited with five minutes.

Problem representation proposed in this chapter is in a very general form which allows modeling different entity types in the same manner. Since PNRA and IH are based on this representation, different transportation problems can easily be associated. We have focused on airline disruption management in this chapter, however, we expect to apply the proposed approach on different transportation systems in near future.

# CHAPTER 6

# CONCLUSION

The work done in this thesis is summarized in Section 6.1 and possible future research directions are discussed in Section 6.2.

## 6.1 Concluding Remarks

We can summarize the main objectives of this thesis as follows:

- integrating cruise speed control option with common recovery actions, and testing the practicality and beneficialness of the solution approaches;
- developing a fully integrated recovery approach for airline disruption management problem;
- evaluating all possible passenger recovery actions in order to make passenger-friendly recovery decisions;
- proposing realistic passenger delay cost formulations in order to enhance the accuracy of the mathematical models; and
- developing a heuristic approach to deal with huge airline networks and challenging solution time limitations.

The mathematical model that we propose in Chapter 3 deals with the aircraft passenger recovery (APR) problem. In this formulation, we manage to integrate several aspects of aircraft recovery and passenger recovery. Passenger rerouting decisions are accurately evaluated in the formulation while an approximation similar to the ones proposed in the literature is used for evaluating passenger delay costs. An important contribution of this chapter is the integration of cruise speed control action. We manage to evaluate cruise speed decisions on a continuous space. Cruise speeds of flights are optimized together with the other recovery actions, and hence, tradeoff between the increased cruise speed and the network effect of the reduction in the arrival delay can be evaluated. Due to the nonlinear increase in fuel consumption when we speed up a flight, the resulting formulation is a mixed integer nonlinear programming model (MINLP). Proposed MINLP has linear constraints and a nonlinear cost term in the objective function. We first linearize the objective function by introducing new constraints. Then, we show that the resulting model can be reformulated as a second-order cone programming (SOCP) model. This enables us to solve APR problem with commercial CQMIP solvers such as CPLEX. In our experimentations, we have been able to solve ARP instances on a four-hub network of a major U.S. airline in less than a minute on the average. Therefore, we state that the proposed reformulation scheme is an efficient method to integrate cruise speed control in integrated airline recovery problems.

After dealing with a partially integrated recovery problem and integration of cruise speed control action in Chapter 3, we focus on full integration (integration of aircraft, crew and passenger recovery together with all recovery actions) in Chapter 4. Major concern of fully integrated recovery approaches proposed in the literature is the increased problem size and complexity. Therefore, problem representation plays a crucial role on the performance of the solution approaches. We propose an alternative connection network (CN) representation in which activities (flights) are represented by nodes. Proposed CN representation has several advantages on problem size. It is very appropriate for our formulations since we focus on integration of different entity types and interdependencies among entities can easily be represented on the common flight node set. Furthermore, all recovery actions, not only for passengers but for all entity types, can be generated through the CN. Since the activities are kept on nodes, variable activity times (cruise speed control) can easily be associated. We propose a network based formulation based on the alternative CN representation, which is again a MINLP model due to the additional fuel cost term in the objective function. With a similar reformulation scheme that is proposed in Chapter 3, we show that the MINLP model is equivalent to a CQMIP model. In order to understand the tradeoff between the burden of cruise speed control action and quality of solutions, we test our formulation with and without cruise speed controllability. In most flight departure time delay and aircraft ready time delay scenarios, we observe that this action is utilized in the optimal solutions. Speeding up several flights may mitigate delays in the downstream flights and have a significant impact on total realized delays. In flight cancellation and hub closure scenarios, on the other hand, we observe another advantage of this action. Being able to work with variable flight times, we can generate new swap and rerouting opportunities that may be very beneficial against disruptions. On the average, we observe a significant reduction in total disruption and recovery costs, and hence, state that it is a very beneficial recovery action to reduce the disturbances of disruptions. This part of our study is accepted for publication in Annals of Operations Research ([7]).

We work with two passenger delay cost functions in Chapter 4. The first one assumes a linear relation between the amount of arrival delay and passenger delay cost. This formulation can be considered as the common practice in the literature. On the other hand, some authors propose a nonlinear relationship. For instance, an arrival delay less than 5 minutes may not cost to a passenger while it may increase more and more as the duration increases. In order to estimate such a relation, we also propose a piecewise linear delay cost function. Another important consideration in calculating the passenger delay cost is the increased complexity while calculating the realized delays of passengers in the recovery. The complexity arises from passenger-related recovery actions. Most studies that integrate passenger recovery assumes that passengers follow their original paths while calculating passenger delay cost. However, they may be reallocated or even be spilled in the recovered schedules, and hence, this assumption may underestimate or overestimate actual passenger delay cost of the solution. We call this method as flight delay approximation and use in our formulations. Furthermore, we propose to passenger delay cost calculation method that is based on the actual delays realized by passengers. This requires modeling individual passengers rather than aggregating passengers in the same fare class of an itinerary. To the best of our knowledge, this formulation is the first to model each passenger explicitly and provide solutions to practical sized problems within the required time limitations.

In our experimentations, we have been able to optimize the majority of the instances with our integrated approach that uses flight delay approximation. On the other hand, we have observed tractability issues with actual delay calculations while dealing with the largest instances. In order to enable find quick solutions to huge instances with our most complex formulations, we propose a heuristic approach in Chapter 5. Flight delay approximation and sequential recovery approach are the common practices in the literature to limit the problem size. Our heuristic approach (Isolation Heuristic), on the other hand, limits the problem size in a different manner. Based on our alternative problem representation, the algorithm starts with a minimal solution space including only the disrupted flight nodes. At several expansion steps, the solution space is expanded by including new entities and flight nodes. Entities to be included are selected with respect to their proximities to the isolated solution space. Therefore, at the end of the expansion process, the entities and flight nodes that are included in the solution space are somehow *close to* the disruption. The procedure can be considered as systematic mimic of the decision making process of the dispatchers in the AOCCs. Furthermore, control parameters of the heuristic help to balance the tradeoff between the problem size and solution quality. We have tested the performance of our approach with four different formulations proposed in Chapter 4 each having a different level of complexity. Our computational study has shown that, the approach is more beneficial with more complex formulations and larger instances. We have been able to reach good quality solutions to large instances for which we are unable to find feasible solutions within the limited solution times.

## 6.2 Future Research Directions

In this thesis, we have integrated cruise speed control action with common recovery actions by the reformulation schemes given in Chapter 3 and Chapter 4. Our computational studies have shown that the enhancements in conic quadratic programming can result in efficient formulations. In our reformulation approaches, we deal with similar constraints representing the relationship between a binary assignment variable ($y$), constant power of a continuous activity time variable ($t^K$) and another continuous variable required to evaluate the additional fuel cost incurred due to the cruise time compression ($q$):

$$\frac{y}{t^K} \leq q$$

The third term can be generalized as a continuous variable that we want to minimize. Binary assignment variables and variable activity times are common in many scheduling and rescheduling problems. Proposed reformulation scheme can be applied in scheduling and rescheduling problems that have such a nonlinear relationship between variable activity times and some term in the objective function.

We have been able to optimize large integrated airline recovery instances within very short solution times with formulations based on our alternative CN representation. Proposed representation has several advantages, one of which is its simplicity, and hence, ease of integrating different types of entities. We have easily modeled restrictions of different entity types in our integrated airline recovery approaches. This property encourages us to apply this representation in different airline problems. As discussed in Section 1.1, schedule planning problem in airlines is a more complex problem with additional considerations such as maximizing the captured passenger demand. On the other hand, greater solution times are allowed. Airline schedule planning problem is relatively more studied than airline recovery problems. However, integrated airline schedule planning problem is still challenging. We believe that we can develop an efficient formulation based on our alternative problem representation in this research area.

Isolation Heuristic that we present in Chapter 5 differs from the other approximation or sequential approaches proposed in the literature. It provides an infrastructure for real time disruption management problems, such as integrated airline recovery problems and disruption management in dial-a-ride problem. In our experimentations, we have observed that the approach is more beneficial when more than one entity type is important to the decision maker. Note that in most transportation systems at least two types of entities are important (vehicles and passengers / commodities, etc.). Using the good structure of the underlying CN representation, the algorithm quickly isolates the solution space that has reduced size to enable real time solutions, and also that can still provide good quality solutions. Underlying CN representation allows to easily model transportation problems. Furthermore, since the heuristic approach is independent of the optimization methodology, it can be easily applied and experimented with different disruption management problems.

# REFERENCES


[1] K. Abdelghany, A. Abdelghany, and G. Ekollu. An integrated decision support tool for airlines schedule recovery during irregular operations. *European Journal of Operational Research*, 185(2):825–848, 2008.

[2] Airbus. Airbus flight operations support and line assistance, getting to grips with the cost index. Technical Report Issue 2, Airbus Customer Services, France, 1998.

[3] Airbus. Airbus flight operations support and line assistance, getting to grips with fuel economy. Technical Report Issue 4, Airbus Customer Services, France, 2004.

[4] M. S. Aktürk, A. Atamtürk, and S. Gürel. Aircraft rescheduling with cruise speed control. *Operations Research*, 62(4):829–845, 2014.

[5] T. Andersson and P. Varbrand. The flight perturbation problem. *Transportation Planning and Technology*, 27(2):91–117, 2004.

[6] M.F. Argüello, J.F. Bard, and G. Yu. A grasp for aircraft routing in response to groundings and delays. *Journal of Combinatorial Optimization*, 1(3):211–228, 1997.

[7] U. Arıkan, S. Gürel, and M. S. Aktürk. Integrated aircraft and passenger recovery with cruise time controllability. *Annals of Operations Research*, doi:10.1007/s10479-013-1424-2, 2013.

[8] J. F. Bard, G. Yu, and M. F. Argüello. Optimizing aircraft routings in response to groundings and delays. *IIE Transactions*, 33(10):931–947, 2001.

[9] C. Barnhart, P. Belobaba, and A. Odoni. Applications of operations research in the airline industry. *Transportation Science*, 37(4):368–391, 2003.

[10] C. Barnhart, N. Boland, L. Clarke, E. L. Johnson, G. L. Nemhauser, and R. Shenoi. Flight string models for aircraft fleeting and routing. *Transportation Science*, 32(3):208–220, 1998.

[11] C. Barnhart and A. M. Cohn. Airline schedule planning: Accomplishments and opportunities. *Manufacturing & Service Operations Management*, 6(1):3–22, 2004.

[12] C. Barnhart, D. Fearing, and V. Vaze. Modeling passenger travel and delays in the national air transportation system. *Operations Research*, 62(3):580–601, 2014.

[13] C. Barnhart, T. Kniker, and M. Lohatepanont. Itinerary-based airline fleet assignment. *Transportation Science*, 36(2):199–217, 2002.

[14] J. E. Beasly and B. Cao. A dynamic programming based algorithm for the crew scheduling problem. *Computers & Operations Research*, 25(7-8):567–582, 1998.

[15] N. Belanger, G. Desaulniers, F. Soumis, and J. Desrosiers. Periodic airline fleet assignment with time windows, spacing constraints and time dependent revenues. *European Journal of Operational Research*, 175(3):1754–1766, 2006.

[16] N. Belanger, G. Desaulniers, F. Soumis, J. Desrosiers, and J. Lavigne. Weekly airline fleet assignment with homogeneity. *Transportation Research Part B*, 40(4):306–318, 2006.

[17] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications.* BTN:ModernOptBook, 2001.

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

[19] Boeing. Fuel conservation strategies: Cost index explained. *AERO Quarterly*, Quarter 2:26–28, 2007.

[20] S. Bratu and C. Barnhart. Flight operations recovery: New approaches considering passenger recovery. *Journal of Scheduling*, 9(3):279–298, 2006.

[21] J-M. Cao and A. Kanafani. Real-time decision support for integration of airline flight cancellations and delays part i: mathematical formulation. *Transportation and Planning Technology*, 20(3):183–199, 1997.

[22] J-M. Cao and A. Kanafani. Real-time decision support for integration of airline flight cancellations and delays part ii: algorithm and computational elements. *Transportation and Planning Technology*, 20(3):201–217, 1997.

[23] H. Chu, E. Gelman, and E. L. Johnson. Solving large scale crew scheduling problems. *European Journal of Operational Research*, 97(2):260–268, 1997.

[24] L. Clarke, E. L. Johnson, G. L. Nemhauser, and Z. Zhu. The aircraft rotation problem. *Annals of Operations Research*, 69:33–46, 1997.

[25] M. Clarke. Irregular airline operations: a review of the state-of-the-practice in airline operations control centers. *Journal of Air Transportation Management*, 4(2):67–76, 1998.

[26] J. Clausen, A. Larsen, J. Larsen, and N. J. Rezanova. Disruption management in the airline industry - concepts, models and methods. *Computers & Operations Research*, 37(5):809–821, 2010.

[27] A. M. Cohn and C. Barnhart. Improving crew scheduling by incorporating key maintenance routing decisions. *Operations Research*, 51(3):387–396, 2003.

[28] A. Cook, G. Tanner, V. Williams, and G. Meise. Dynamic cost indexing - managing airline delay costs. *Journal of Air Transport Management*, 15(1):26–35, 2009.

[29] J.-F. Cordeau, G. Stojkovic, F. Soumis, and J Desrosiers. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation Science*, 35(4):375–388, 2001.

[30] M. S. Daskin and N. D. Panayotopoulos. A lagrangian relaxation approach to assigning aircraft to routes in hub and spoke networks. *Transportation Science*, 23(2):91–99, 1989.

[31] G. Desaulniers, J. Desrosiers, Y. Dumas, M. M. Solomon, and F. Soumis. Daily aircraft routing and scheduling. *Management Science*, 43(6):841–855, 1997.

[32] G. Dobson and P. J. Lederer. Airline scheduling and routing in a hub-and-spoke system. *Transportation Science*, 27(3):281–297, 1993.

[33] A. S. Duran, S. Gürel, and M. S. Aktürk. Robust airline scheduling with controllable cruise times and chance constraints. *IIE Transactions*, doi:10.1080/0740817X.2014.916457, 2014.

[34] N. Eggenberg, M. Salani, and M. Bierlaire. Constraint-specific recovery network forsolving airline recovery problems. *Computers & Operations Research*, 37(6):1014–1026, 2010.

[35] T. Feo and J. Bard. Flight scheduling and maintenance base planning. *Management Science*, 35(12):1415–1432, 1989.

[36] L. A. Garrow. *Discrete Choice Modelling and Air Travel Demand: Theory and Applications.* Aldershot: Ashgate Publishing, 2010.

[37] L. A. Garrow, S. P. Jones, and R. A. Parker. How much airline customers are willing to pay: an analysis of price sensitivity in online distribution channels. *Journal of Revenue and Pricing Management*, 6(1):1–20, 2010.

[38] D. Goldsman, E. L. Johnson, A. Kleywegt, and G. L. Nemhauser. A stochastic model of airline operations. *Transportation Science*, 36(4):357–377, 2002.

[39] R. Gopalan and K. T. Talluri. Mathematical models in airline schedule planning: A survey. *Annals of Operations Research*, 76(1):155–185, 1998.

[40] R. J. Graham, L. A. Garrow, and J. D. Leonard. Business travelers' ticketing, refund, and exchange behavior. *Journal of Air Transportation Management*, 16(4):196–201, 2010.

[41] Y. Guo, T. Mellouli, L. Suhl, and M. P. Thiel. A partially integrated airline crew scheduling approach with time-dependent crew capacities and multiple home bases. *European Journal of Operational Research*, 171(3):1169–1181, 2006.

[42] C. A. Hane, C. Barnhart, E. L. Johnson, R. E. Marsten, G. L. Nemhauser, and G. Sigismondi. The fleet assignment problem: Solving a large-scale integer program. *Mathematical Programming*, 70(1-3):211–232, 1995.

[43] M. Haouari, N. Aissaoui, and Z. Mansour. Network flow-based approaches for integrated aircraft fleeting and routing. *European Journal of Operational Research*, 2(1):591–699, 2009.

[44] K. Hoffman and M. Padberg. Solving airline crew scheduling problems by branch-and-cut. *Management Science*, 39(6):657–682, 1993.

[45] I. Ioachim, J. Desrosiers, F. Soumis, and N. Belanger. Fleet assignment and routing with schedule synchronization constraints. *European Journal of Operational Research*, 119(1):75–90, 1999.

[46] N. Jafari and S. H. Zegordi. The airline perturbation problem: considering disrupted passengers. *Transportation Planning and Technology*, 33(2):203–220, 2010.

[47] A. I. Z. Jarrah, G. Yu, N. Krishnamurthy, and A. Rakshit. A decision support framework for airline flight cancellations and delays. *Transportation Science*, 27(3):266–280, 1993.

[48] A. I. Z. Jarrah, G. Yu, N. Krishnamurthy, and A. Rakshit. A dynamic approach for aircraft assignment and maintenance scheduling by airlines. *Journal of Air Transport Management*, 6(4):233–237, 2000.

[49] D. Kim and C. Barnhart. Flight schedule design for a charter airline. *Computers & Operations Research*, 34(6):1516–1531, 2007.

[50] D. Klabjan, E. L. Johnson, and G. L. Nemhauser. Solving large airline crew scheduling problems: Random pairing generation and strong branching. *Computational Optimization Applications*, 20(1):73–91, 2001.

[51] D. Klabjan, E. L. Johnson, G. L. Nemhauser, E. Gelman, and S. Ramaswamy. Airline crew scheduling with time windows and plane-count constraints. *Transportation Science*, 36(3):337–348, 2002.

[52] N. Kohl and S. E. Karisch. Airline crew rostering: Problem types, modeling and optimization. *Annals of Operations Research*, 127(1-4):223–257, 2004.

[53] S. Lan, J. P. Clarke, and C. Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006.

[54] L. Lettovsky, E. L. Johnson, and G. L. Nemhauser. Airline crew recovery. *Transportation Science*, 34(4):337–348, 2000.

[55] M. Lohatepanont and C. Barnhart. Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science*, 38(1):19–32, 2004.

[56] M. Love, K. R. Sorensen, J. Larsen, and J. Clausen. Using heuristics to solve the dedicated aircraft recovery problem. *Central European Journal of Operations Research*, 13(2):189–207, 2005.

[57] J. A. Lovegren and R. J. Hansman. Quantification of fuel burn reduction in cruise via speed and altitude optimization strategies. Technical Report ICAT-2011-03, Massachusetts Institute of Technology, International Center for Air Transportation (ICAT), 2011.

[58] S. Luo and G. Yu. On the airline schedule perturbation problem caused by the ground delay program. *Transportation Science*, 31(4):298–311, 1997.

[59] L. Marla, B. Vaaben, and C. Barnhart. Integrated disruption management and flight planning to trade off delays and fuel burn. Technical report, Technical University of Denmark, Department of Management Engineering, 2011.

[60] D. F. X. Mathaisel. Decision support for airline system operations control and irregular operations. *Computers & Operations Research*, 23(11):1083–1098, 1996.

[61] C. P. Medard and N. Sawhney. Airline crew scheduling from planning to operations. *European Journal of Operational Research*, 183(3):1013–1027, 2007.

[62] J. D. Petersen, G. Sölveling, J. P. Clarke, E. L. Johnson, and S. Shebalov. An optimization approach to airline integrated recovery. *Transportation Science*, 46(4):482–500, 2012.

[63] B. Rexing, C. Barnhart, T. Kniker, A. Jarrah, and N. Krishnamurthy. Airline fleet assignment with time windows. *Transportation Science*, 34(1):1–20, 2000.

[64] J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. A stochastic model of airline operations. *Transportation Science*, 36(4):357–377, 2002.

[65] J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. Rerouting aircraft for airline recovery. *Transportation Science*, 37(4):408–421, 2003.

[66] J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. A robust fleet-assignment model with hub isolation and short cycles. *Transportation Science*, 38(3):357–368, 2004.

[67] R. A. Rushmeier and S. A. Kontogiorgis. Advances in the optimization of airline fleet assignment. *Transportation Science*, 31(2):159–169, 1997.

[68] A. Sarac, R. Batta, and C. M. Rump. A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research*, 175(3):1850–1869, 2006.

[69] A. Schaefer, E. L. Johnson, A. Kleywegt, and G. L. Nemhauser. Airline crew scheduling under uncertainty. *Transportation Science*, 39(3):340–348, 2005.

[70] H. D. Sherali, K. H. Bae, and M. Haouari. An integrated approach for airline flight selection and timing, fleet assignment, and aircraft routing. *Transportation Science*, 47(4):455–476, 2013.

[71] G. Stojkovic and F. Soumis. The operational airline crew scheduling problem. *Transportation Science*, 32(3):232–245, 1998.

[72] G. Stojkovic, F. Soumis, J. Desrosiers, and M. Solomon. An optimization model for a real-time flight scheduling problem. *Transportation Research Part A: Policy and Practice*, 36(9):779–788, 2002.

[73] K. T. Talluri. Swapping applications in a daily airline fleet assignment. *Transportation Science*, 30(3):237–248, 1996.

[74] D. Teodorovic. A model for designing the meteorologically most reliable airline schedule. *European Journal of Operations Research*, 21(2):156–164, 1984.

[75] D. Teodorovic and S. Guberinic. Optimal dispatching strategy on an airline network after a schedule perturbation. *European Journal of Operations Research*, 15(2):178–182, 1984.

[76] D. Teodorovic and G. Stojkovic. Model for operational daily airline scheduling. *Transportation Planning and Technology*, 14(4):273–285, 1990.

[77] D. Teodorovic and G. Stojkovic. Model to reduce airline schedule disturbances. *Journal of Transportation Engineering*, 121(4):324–331, 1995.

[78] B. G. Thengvall, J. F. Bard, and G. Yu. Balancing user preferences for aircraft schedule recovery during irregular operations. *IIE Transactions*, 32(3):181–193, 2000.

[79] B. G. Thengvall, J. F. Bard, and G. Yu. A bundle algorithm approach for the aircraft schedule recovery problem during hub closures. *Transportation Science*, 37(4):392–407, 2003.

[80] P. H. Vance, C. Barnhart, E. L. Johnson, and G. L. Nemhauser. Airline crew scheduling: A new formulation and decomposition algorithm. *Operations Research*, 45(2):188–200, 1997.

[81] G. Wei, G. Yu, and M. Song. Optimization model and algorithm for crew management during airline irregular operations. *Journal of Combinatorial Optimization*, 1(3):305–321, 1997.

[82] S. Yan and J.-C. Chang. Airline cockpit crew scheduling. *European Journal of Operational Research*, 136(3):501–511, 2002.

[83] S. Yan and Y.-P. Tu. A network model for airline cabin crew scheduling. *European Journal of Operational Research*, 140(3):531–540, 2002.

[84] S. Yan and D-H. Yang. A decision support framework for handling schedule perturbation. *Transportation Research B*, 30(6):405–419, 1996.

[85] S. Yan and H.-F. Young. A decision support framework for multi-fleet routing and multi-stop flight scheduling. *Transportation Research A*, 30(5):379–398, 1996.

# APPENDIX A

# COMPUTATIONAL RESULTS OF ISOLATION HEURISTIC

Table A.1: Quality of solutions with different parameter settings in FDA- instances (*Balanced* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt | Com. | Idl. | 30 | **30** | 10 | 13 | 14.63% | 38.19% |
| | | | Rel. | 30 | **30** | 10 | 13 | 13.60% | 38.19% |
| | | Sch. | Idl. | 30 | **30** | 4 | 8 | 23.31% | 54.44% |
| | | | Rel. | 30 | **30** | 4 | 8 | 22.23% | 52.51% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 16 | 26 | 3.13% | 12.04% |
| | | | Rel. | 30 | **30** | 18 | **30** | 1.97% | 7.97% |
| | | Sch. | Idl. | 30 | **30** | 12 | 16 | 8.48% | 29.56% |
| | | | Rel. | 30 | **30** | 12 | 23 | 4.49% | 14.41% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 16 | **30** | 2.35% | 9.10% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.39% | 5.04% |
| | | Sch. | Idl. | 30 | **30** | 12 | 29 | 4.04% | 16.99% |
| | | | Rel. | 30 | **30** | 12 | **30** | 3.24% | 8.00% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | **30** | 22 | **30** | 1.26% | 6.18% |
| | | | Rel. | 30 | **30** | 22 | 29 | 1.27% | 11.24% |
| | | Sch. | Idl. | 30 | **30** | 27 | 29 | 0.80% | 16.99% |
| | | | Rel. | 30 | **30** | 28 | **30** | 0.23% | 3.51% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 19 | 22 | 4.42% | 22.66% |
| | | | Rel. | 30 | **30** | 20 | **30** | 1.38% | 8.23% |
| | | Sch. | Idl. | 30 | **30** | 27 | 29 | 0.80% | 16.99% |
| | | | Rel. | 30 | **30** | 28 | **30** | 0.23% | 3.51% |

Table A.2: Quality of solutions with different parameter settings in FDA- instances (*Independent* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Ind. | Alt | Com. | Idl. | 30 | **30** | 14 | 17 | 13.18% | 56.81% |
| | | | Rel. | 30 | **30** | 10 | 14 | 14.71% | 52.49% |
| | | Sch. | Idl. | 30 | **30** | 4 | 10 | 21.80% | 57.35% |
| | | | Rel. | 30 | **30** | 4 | 10 | 22.20% | 55.94% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 12 | 16 | 9.62% | 26.60% |
| | | | Rel. | 30 | **30** | 12 | 16 | 9.74% | 35.99% |
| | | Sch. | Idl. | 30 | **30** | 8 | 12 | 13.22% | 31.51% |
| | | | Rel. | 30 | **30** | 8 | 12 | 13.36% | 35.99% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 10 | 22 | 6.39% | 24.84% |
| | | | Rel. | 30 | **30** | 11 | **30** | 4.14% | 9.87% |
| | | Sch. | Idl. | 30 | **30** | 8 | 16 | 11.10% | 39.50% |
| | | | Rel. | 30 | **30** | 8 | 15 | 10.63% | 29.56% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | **30** | 16 | 26 | 3.32% | 25.06% |
| | | | Rel. | 30 | **30** | 16 | 27 | 2.74% | 12.22% |
| | | Sch. | Idl. | 30 | **30** | 16 | 26 | 4.23% | 25.06% |
| | | | Rel. | 30 | **30** | 16 | 26 | 3.98% | 17.23% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 16 | 22 | 6.15% | 39.87% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.40% | 6.91% |
| | | Sch. | Idl. | 30 | **30** | 15 | 27 | 3.08% | 13.22% |
| | | | Rel. | 30 | **30** | 16 | 29 | 2.49% | 11.12% |

Table A.3: Quality of solutions with different parameter settings in AD- instances (*Balanced* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt | Com. | Idl. | 30 | **30** | 10 | 14 | 15.29% | 39.74% |
| | | | Rel. | 30 | **30** | 10 | 13 | 16.04% | 39.74% |
| | | Sch. | Idl. | 30 | **30** | 7 | 11 | 22.63% | 59.46% |
| | | | Rel. | 30 | **30** | 8 | 11 | 23.02% | 59.46% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 15 | 24 | 3.21% | 11.63% |
| | | | Rel. | 30 | **30** | 15 | 28 | 2.61% | 10.46% |
| | | Sch. | Idl. | 30 | **30** | 11 | 17 | 10.38% | 36.93% |
| | | | Rel. | 30 | **30** | 14 | 24 | 4.60% | 20.20% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 17 | 26 | 2.19% | 12.24% |
| | | | Rel. | 30 | **30** | 21 | **30** | 1.22% | 7.72% |
| | | Sch. | Idl. | 30 | **30** | 10 | 19 | 9.30% | 37.02% |
| | | | Rel. | 30 | **30** | 13 | 29 | 4.63% | 10.74% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | **30** | 18 | 18 | 11.67% | 36.93% |
| | | | Rel. | 30 | **30** | 17 | 18 | 11.75% | 37.99% |
| | | Sch. | Idl. | 30 | **30** | 13 | 16 | 13.88% | 55.94% |
| | | | Rel. | 30 | **30** | 14 | 16 | 13.32% | 57.06% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 18 | 18 | 3.72% | 36.93% |
| | | | Rel. | 30 | **30** | 17 | 18 | 2.71% | 26.27% |
| | | Sch. | Idl. | 30 | **30** | 15 | 18 | 2.36% | 36.93% |
| | | | Rel. | 30 | **30** | 14 | 18 | 2.18% | 31.88% |

Table A.4: Quality of solutions with different parameter settings in AD- instances (*Independent* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Ind. | Alt | Com. | Idl. | 30 | **30** | 10 | 15 | 15.19% | 39.74% |
| | | | Rel. | 30 | **30** | 10 | 14 | 15.63% | 39.74% |
| | | Sch. | Idl. | 30 | **30** | 7 | 12 | 19.93% | 51.10% |
| | | | Rel. | 30 | **30** | 8 | 12 | 19.57% | 51.10% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 12 | 15 | 10.43% | 36.93% |
| | | | Rel. | 30 | **30** | 12 | 16 | 9.98% | 36.93% |
| | | Sch. | Idl. | 30 | **30** | 8 | 14 | 14.83% | 36.93% |
| | | | Rel. | 30 | **30** | 8 | 13 | 14.61% | 36.93% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 10 | 20 | 8.24% | 25.94% |
| | | | Rel. | 30 | **30** | 9 | 27 | 5.36% | 18.48% |
| | | Sch. | Idl. | 30 | **30** | 9 | 13 | 12.69% | 36.93% |
| | | | Rel. | 30 | **30** | 8 | 14 | 11.44% | 36.93% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | **30** | 14 | 18 | 7.49% | 23.50% |
| | | | Rel. | 30 | **30** | 12 | 21 | 5.95% | 22.57% |
| | | Sch. | Idl. | 30 | **30** | 8 | 17 | 9.38% | 38.54% |
| | | | Rel. | 30 | **30** | 9 | 17 | 8.09% | 23.27% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | **30** | 13 | 28 | 3.70% | 12.87% |
| | | | Rel. | 30 | **30** | 19 | **30** | 1.66% | 7.70% |
| | | Sch. | Idl. | 30 | **30** | 16 | 21 | 5.55% | 17.99% |
| | | | Rel. | 30 | **30** | 14 | 29 | 3.73% | 14.35% |

Table A.5: Quality of solutions with different parameter settings in FDA+ instances (*Balanced* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt | Com. | Idl. | 30 | **30** | 5 | 8 | 16.83% | 42.27% |
| | | | Rel. | 30 | **30** | 8 | 8 | 14.48% | 37.08% |
| | | Sch. | Idl. | 30 | **30** | 0 | 9 | 20.81% | 46.53% |
| | | | Rel. | 30 | **30** | 1 | 11 | 17.25% | 37.72% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 3 | 15 | 9.48% | 17.44% |
| | | | Rel. | 30 | **30** | 10 | **30** | 4.69% | 9.72% |
| | | Sch. | Idl. | 30 | **30** | 1 | 11 | 13.89% | 26.69% |
| | | | Rel. | 30 | **30** | 1 | 13 | 10.70% | 23.83% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 8 | 28 | 3.38% | 10.33% |
| | | | Rel. | 30 | **30** | 17 | **30** | 2.10% | 8.82% |
| | | Sch. | Idl. | 30 | **30** | 6 | 26 | 5.43% | 14.65% |
| | | | Rel. | 30 | **30** | 7 | 27 | 5.82% | 14.65% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | 12 | 0 | 0 | 35.00% | 59.99% |
| | | | Rel. | 30 | 13 | 3 | 5 | 23.00% | 48.67% |
| | | Sch. | Idl. | 30 | 15 | 0 | 2 | 25.15% | 59.63% |
| | | | Rel. | 30 | 14 | 2 | 6 | 19.11% | 45.61% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | 11 | 0 | 3 | 19.07% | 38.09% |
| | | | Rel. | 30 | 10 | 2 | 3 | 18.37% | 39.48% |
| | | Sch. | Idl. | 30 | 11 | 0 | 2 | 27.16% | 59.63% |
| | | | Rel. | 30 | 11 | 0 | 1 | 23.58% | 57.41% |

Table A.6: Quality of solutions with different parameter settings in FDA+ instances (*Independent* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Ind. | Alt | Com. | Idl. | 30 | **30** | 5 | 6 | 22.93% | 56.81% |
| | | | Rel. | 30 | **30** | 4 | 6 | 22.91% | 56.81% |
| | | Sch. | Idl. | 30 | **30** | 1 | 5 | 28.99% | 56.81% |
| | | | Rel. | 30 | **30** | 1 | 7 | 28.59% | 56.81% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 11 | 13 | 13.47% | 49.60% |
| | | | Rel. | 30 | **30** | 12 | 17 | 9.38% | 31.10% |
| | | Sch. | Idl. | 30 | **30** | 8 | 10 | 14.74% | 51.70% |
| | | | Rel. | 30 | **30** | 10 | 13 | 13.16% | 37.93% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 4 | 22 | 6.67% | 17.66% |
| | | | Rel. | 30 | **30** | 12 | 28 | 3.41% | 10.60% |
| | | Sch. | Idl. | 30 | **30** | 1 | 14 | 11.02% | 20.52% |
| | | | Rel. | 30 | **30** | 2 | 22 | 6.98% | 12.44% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | 25 | 0 | 3 | 16.86% | 51.63% |
| | | | Rel. | 30 | 25 | 0 | 5 | 16.81% | 46.94% |
| | | Sch. | Idl. | 30 | 24 | 1 | 6 | 16.59% | 51.64% |
| | | | Rel. | 30 | 21 | 2 | 6 | 16.17% | 51.67% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | 22 | 0 | 3 | 18.33% | 51.64% |
| | | | Rel. | 30 | 23 | 0 | 4 | 19.23% | 51.57% |
| | | Sch. | Idl. | 30 | 23 | 1 | 7 | 16.92% | 51.65% |
| | | | Rel. | 30 | 22 | 2 | 9 | 15.65% | 46.86% |

Table A.7: Quality of solutions with different parameter settings in AD+ instances (*Balanced* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt | Com. | Idl. | 30 | **30** | 0 | 10 | 16.62% | 49.42% |
| | | | Rel. | 30 | **30** | 1 | 14 | 13.52% | 34.94% |
| | | Sch. | Idl. | 30 | **30** | 0 | 7 | 20.24% | 47.57% |
| | | | Rel. | 30 | **30** | 1 | 10 | 17.92% | 47.09% |
| | Alt-Aff | Com. | Idl. | 30 | 22 | 0 | 5 | 14.79% | 33.10% |
| | | | Rel. | 30 | 23 | 1 | 9 | 11.23% | 24.07% |
| | | Sch. | Idl. | 30 | **30** | 0 | 8 | 14.10% | 30.15% |
| | | | Rel. | 30 | **30** | 4 | 12 | 12.74% | 27.52% |
| | Alt-Alt | Com. | Idl. | 30 | 18 | 5 | 9 | 9.99% | 30.66% |
| | | | Rel. | 30 | 19 | 5 | 9 | 8.36% | 40.11% |
| | | Sch. | Idl. | 30 | **30** | 2 | 24 | 7.68% | 18.46% |
| | | | Rel. | 30 | **30** | 2 | 27 | 5.51% | 14.63% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | 8 | 1 | 3 | 15.15% | 33.96% |
| | | | Rel. | 30 | 8 | 0 | 2 | 18.45% | 35.79% |
| | | Sch. | Idl. | 30 | 9 | 0 | 2 | 19.96% | 38.29% |
| | | | Rel. | 30 | 10 | 0 | 3 | 13.20% | 37.21% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | 6 | 1 | 3 | 17.34% | 42.43% |
| | | | Rel. | 30 | 6 | 0 | 1 | 31.79% | 58.09% |
| | | Sch. | Idl. | 30 | 5 | 0 | 1 | 32.84% | 51.88% |
| | | | Rel. | 30 | 6 | 0 | 0 | 31.83% | 46.09% |

Table A.8: Quality of solutions with different parameter settings in AD+ instances (*Independent* strategy)

| $\alpha$ | $\beta$ | $\gamma$ | $\theta$ | nb. ins. | nb. feas. | nb. opt. | nb. 10%. | Avg (Gap) | Max (Gap) |
|---|---|---|---|---|---|---|---|---|---|
| Bal. | Alt | Com. | Idl. | 30 | **30** | 5 | 9 | 19.60% | 55.25% |
| | | | Rel. | 30 | **30** | 3 | 10 | 16.69% | 46.21% |
| | | Sch. | Idl. | 30 | **30** | 2 | 5 | 25.83% | 55.84% |
| | | | Rel. | 30 | **30** | 0 | 5 | 24.88% | 55.16% |
| | Alt-Aff | Com. | Idl. | 30 | **30** | 5 | 7 | 18.99% | 43.79% |
| | | | Rel. | 30 | **30** | 5 | 11 | 14.12% | 32.01% |
| | | Sch. | Idl. | 30 | **30** | 1 | 7 | 19.26% | 42.65% |
| | | | Rel. | 30 | **30** | 1 | 7 | 17.99% | 43.11% |
| | Alt-Alt | Com. | Idl. | 30 | **30** | 1 | 15 | 12.01% | 27.12% |
| | | | Rel. | 30 | **30** | 3 | 19 | 7.74% | 14.22% |
| | | Sch. | Idl. | 30 | **30** | 0 | 6 | 19.51% | 43.12% |
| | | | Rel. | 30 | **30** | 1 | 11 | 17.61% | 44.56% |
| | Alt-Alt-Aff | Com. | Idl. | 30 | 8 | 0 | 1 | 10.47% | 12.62% |
| | | | Rel. | 30 | 10 | 1 | 2 | 11.05% | 22.48% |
| | | Sch. | Idl. | 30 | 11 | 1 | 4 | 7.57% | 12.62% |
| | | | Rel. | 30 | 11 | 1 | 5 | 13.00% | 51.16% |
| | Alt-Alt-Alt | Com. | Idl. | 30 | 6 | 0 | 2 | 16.80% | 42.83% |
| | | | Rel. | 30 | 11 | 0 | 2 | 14.67% | 35.97% |
| | | Sch. | Idl. | 30 | 9 | 1 | 4 | 12.08% | 38.75% |
| | | | Rel. | 30 | 10 | 1 | 4 | 13.54% | 49.03% |

# CURRICULUM VITAE

## PERSONAL INFORMATION

**Surname, Name:** Arıkan, Uğur

**Nationality:** Turkish (TC)

**Date and Place of Birth:** 05.05.1984, Adana

**Marital Status:** Single

**Phone:** 0 312 210 4792

**Fax:** 0 312 210 4786

**Email:** ugur.arikan.ie@live.com

## EDUCATION

| Degree | Institution | Year of Graduation |
|---|---|---|
| M.S. | Graduate School of Natural and Applied Sciences / Industrial Engineering | 2009 |
| B.S. | Faculty of Engineering / Industrial Engineering | 2006 |
| High School | Adana Anatolian High School | 2002 |

## PROFESSIONAL EXPERIENCE

| Year | Place | Enrollment |
|---|---|---|
| Dec, 2010 - Present | Industrial Engineering Department / METU | Research Assistant |
| Sep, 2008 - Sep, 2009 | Pagos, Inc. | Industrial Engineer |
| Sep, 2006 - Dec, 2010 | Computer Center / METU | Industrial Engineer |

**PUBLICATIONS**

U. Arıkan, S. Gürel, and M. S. Aktürk. Integrated aircraft and passenger recovery with cruise time controllability. *Annals of Operations Research*, doi: 10.1007/s10479-013-1424-2, 2013.