LARGE-SCALE MIXED INTEGER OPTIMIZATION APPROACHES FOR SCHEDULING AIRLINE OPERATIONS UNDER IRREGULARITY

A Thesis
Presented to
The Academic Faculty

by

Jon D. Petersen

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
H. Milton Stewart School of Industrial and Systems Engineering

Georgia Institute of Technology
May 2012

LARGE-SCALE MIXED INTEGER OPTIMIZATION APPROACHES FOR SCHEDULING AIRLINE OPERATIONS UNDER IRREGULARITY

Approved by:

Ellis L. Johnson, *Committee co-chair*
H. Milton Stewart School of Industrial
and Systems Engineering
Georgia Institute of Technology

John-Paul B. Clarke, *Committee co-chair*
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

George L. Nemhauser
H. Milton Stewart School of Industrial
and Systems Engineering
Georgia Institute of Technology

Shabbir Ahmed
H. Milton Stewart School of Industrial
and Systems Engineering
Georgia Institute of Technology

Michael D.D. Clarke
Director, Marketing and Solutions
Management
Sabre Airline Solutions

Date Approved: March 28 2012

To my parents

ACKNOWLEDGEMENTS

As far back as I can remember, my father has advised me to always surround myself with those individuals I look up to the most. One look at the committee members on this thesis validates how seriously I have taken these words. It is a humbling experience to have such a distinguished team of researchers on board. I am forever grateful for their advice and guidance.

First and foremost I am indebted to my advisers Professors Ellis Johnson and John-Paul Clarke. Their frank and candid advice has considerably enhanced the quality of work contained herein. Both gentlemen have been very supportive in the difficult times, encouraging in the good times, and wonderful friends throughout. I could not be more appreciative for all they have done for me in my time at Georgia Tech. I also wish to express my gratitude for the other committee members whose insights have been invaluable along the way. I have thoroughly enjoyed my interactions with Professors George Nemhauser and Shabbir Ahmed. It has been a pleasure to learn from such exemplary scholars. When hearing about 'airline recovery' perhaps the first image conjured up in one's mind is Dr. Michael Clarke, who has been a pioneer in airline operations. He has spent numerous hours with me over the years teaching me about both aviation research as well as airline operations. While I cannot begin to thank him enough for all his time and effort, I hope to accrue a sufficiently large debt in lieu of a bar tab in the years to come that will at least marginally close the appreciation gap.

I have benefited from the mentorship of Dr. Sergey Shebalov who has provided strong guidance, and has had a strong influence on my work. Ross Darrow has also been a superb mentor. I am thankful for his assistance in putting up with a boarder during my Sabre internships. I also thank my friend and colleague Gustaf Sölveling with whom I have enjoyed a fruitful collaboration with. I also wish to acknowledge his invaluable assistance with some of the implementation in Chapter 3.

One of the biggest rewards of my Georgia Tech years has been meaningful friendships that have been developed. I could not have been more fortunate to have had Aly Megahed as an officemate who has transcended into a close friend that has made the journey a memorable one. Gallivanting around the globe with my travel buddies Aixa Cintrón, Thomas Drtil, Peter Gierlich, Annika Lechner, Stefan Lier, Uttam Malani, Matthias Pauli, and Claus Reeker has given me experiences that I could not have imagined growing up in a small town in Minnesota. I already look forward to our next trip. I would also like to sincerely thank the friends I have made, particularly So Yeon Chun, Alaa Elwany, Anand Gopal, Harjot Grewal, Kapil Gupta, Akshay Gupte, Helder Inacio, Fatma Kilinç-Karzan, Yaxian Li, Dexin Luo, Ahmed Nazeem, Dimitri Papageorgiou, Naragain (Jan) Phumchusri, Feng Qiu, Jeffrey Rieske, Norbert Remenyi, Claudio Prata Santiago, Sangho Shim, Dan Steffy, Monica Villarreal, Fangfang (Fiona) Xiao, Huizhu Wang, and Kai Wittek.

The appreciation I have for my family and all they have provided to me over the years is far too great to even attempt to put in words. Suffice it to say they are the greatest influence on my life, and I am who I am today because of their love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Perhaps no single industry has benefited more from advancements in computation, analytics, and optimization than the airline industry. Operations Research (OR) is now ubiquitous in the way airlines develop their schedules, price their itineraries, manage their fleet, route their aircraft, and schedule their crew. These problems, among others, are well-known to industry practitioners and academics alike and arise within the context of the *planning environment* which takes place well in advance of the date of departure. One salient feature of the planning environment is that decisions are made in a frictionless environment that do not consider perturbations to an existing schedule. Airline operations are rife with disruptions caused by factors such as convective weather, aircraft failure, air traffic control restrictions, network effects, among other irregularities. Substantially less work in the OR community has been examined within the context of the real-time *operational environment*.

While problems in the planning and operational environments are similar from a mathematical perspective, the complexity of the operational environment is exacerbated by two factors. First, decisions need to be made in as close to real-time as possible. Unlike the planning phase, decision-makers do not have hours of time to return a decision. Secondly, there are a host of operational considerations in which complex rules mandated by regulatory agencies like the Federal Administration Association (FAA), airline requirements, or union rules. Such restrictions often make finding even a feasible set of re-scheduling decisions an arduous task, let alone the global optimum.

The goals and objectives of this thesis are found in Chapter 1. Chapter 2 provides an overview airline operations and the current practices of disruption management employed at most airlines. Both the causes and the costs associated with irregular operations are surveyed. The role of airline Operations Control Center (OCC) is discussed in which serves as the real-time decision making environment that is important to understand for the body of this work.

Chapter 3 introduces an optimization-based approach to solve the Airline Integrated Recovery (AIR) problem that simultaneously solves re-scheduling decisions for the operating schedule, aircraft routings, crew assignments, and passenger itineraries. The methodology is validated by using real-world industrial data from a U.S. hub-and-spoke regional carrier and we show how the incumbent approach can dominate the incumbent sequential approach in way that is amenable to the operational constraints imposed by a decision-making environment.

Computational effort is central to the efficacy of any algorithm present in a real-time decision making environment such as an OCC. The latter two chapters illustrate various methods that are shown to expedite more traditional large-scale optimization methods that are applicable a wide family of optimization problems, including the AIR problem. Chapter 4 shows how delayed constraint generation and column generation may be used simultaneously through use of alternate polyhedra that verify whether or not a given cut that has been generated from a subset of variables remains globally valid.

While Benders' decomposition is a well-known algorithm to solve problems exhibiting a block structure, one possible drawback is slow convergence. Expediting Benders' decomposition has been explored in the literature through model reformulation, improving bounds, and cut selection strategies, but little has been studied how to strengthen a standard cut. Chapter 5 examines four methods for the convergence may be accelerated through an affine transformation into the interior of the feasible set, generating a split cut induced by a standard Benders' inequality, sequential lifting, and superadditive lifting over a relaxation of a multi-row system. It is shown that the first two methods yield the most promising results within the context of an AIR model.

CHAPTER I

INTRODUCTION

Operations Research (OR) has played a critical role in the complex process airlines use for scheduling various resources for their operations such as flights, aircraft, crew members, airport operations, and passengers. In the era that followed airline deregulation in the United States in 1978, the industry has been hypercompetitive as airlines had considerably more autonomy in determining where, how often, and at which price to charge for flights comprising their networks. In an industry whose margins are often low, operational efficiency is of paramount importance to the success of airlines as the gap between profitability and bankruptcy is narrow. Achieving operational efficiency, however, is nontrivial. The airline industry exhibits some of the most large and difficult to solve problems arising in transportation, and has received considerable attention from industry practitioners and academics alike in the OR literature since the inception of the field in the 1950s.

While methodology has always been ahead of computing, the gap is considerably smaller to date with modern innovations in computing. Models that were previously thought to be intractable are becoming solvable. This dissertation exploits one such class of problems. A brief overview of the airline planning process is given before a discussion of the operational environment that the core of this body of research lies within.

1.1 *Research Contributions*

There is a rich body of research dedicated to solve large-scale problems arising in aviation applications. The airline industry relies heavily upon innovations in both analytics and computation within the OR community. Early work applying mathematical programming to air transportation have focused on airline scheduling problems within the context of the *planning environment* in which scheduling decisions made for each flight are done well in advance of the day of operations. Resources for the flight schedule, fleets, aircraft, and crew members (all discussed below) are usually scheduled at a minimum of two months prior to

departure. Consequently these models are conducted in a vacuum which do not account for a myriad of disruptions that cause delays or cancellations.

The *operational environment* is very different than that of its planning counterpart. In spite of the advancements OR has brought forth with problems in the planning environment, much less has been done in the context of the day of operations. Disruption management plays an integral role of the day-to-day operations for all airlines. The majority of disruption management schemes still rely heavily on the manual construction of re-scheduling decisions made at the flight, aircraft, crew, and passenger levels. The desire to incorporate more of an optimization-based approach to disruption management is obvious, yet extremely complex. Already large and difficult-to-solve problems requiring substantial computing time and resources are exacerbated by the need to make decisions in as close to real-time as possible. Finding a feasible solution alone may be difficult due to maintenance restrictions on aircraft, crew legality restrictions on revised crew schedules, and air traffic control restrictions on scheduling decisions.

A *recovery operation* occurs in the presence of irregularity in which an airline makes re-scheduling decisions to its planned schedule, aircraft rotations, crew schedule, and passenger itineraries. Some work on recovery has been conducted in the literature, but for a number of reasons discussed in the subsequent chapters, airlines still rely heavily on manual or heuristic procedures. One reason is that traditional work on recovery seeks to solve a proper subset of these four classes of re-scheduling decisions. Relying on a sequential procedure to return a complete solution may be untenable. For example, a feasible flight schedule may be infeasible for crew schedules. Moreover, conflicts are likely to exist within operations. An attractive set of re-scheduling decisions that preserves all maintenance requirements for aircraft may induce excessive delays for passengers leading to a substantial loss in passenger goodwill. The desire to have an optimization engine that returns re-scheduling decisions for the four class of problems is naturally of interest, but has not been introduced to date.

There are two fundamental objectives to this thesis. The first is to introduce an optimization-based approach to find re-scheduling decisions in the presence of irregularity that is able to find a set of re-scheduling decisions for the flight schedule, aircraft rotations,

crew schedules, and passenger itineraries in a manner that:

- integrates all four classes of re-scheduling decisions

- solves in a suitable runtime that is amenable to the constraints imposed by the operational environment

- preserves fundamental legality requirements on flights, aircraft, and crew

- is passenger-centric that seeks to minimize aggregate passenger delay.

The second objective is to investigate methodological approaches to achieve the prior goal and study their uses in an abstract environment that may be of use to problems exhibiting a similar structure. These advancements are likely to be particularly useful in solving various applications that seek to integrate various problems within the context of airline scheduling, which seek to:

- incorporate column generation with Benders decomposition in a simultaneous manner

- accelerate Benders decomposition through finding strengthened cutting planes.

In order to study the underlying mathematics of real-time scheduling, we first review the airline planning process.

1.2   Overview of the Airline Planning Process

The airline planning process is the complex phase in which resource scheduling decisions are to be made on fleet allocation, aircraft assignments, crew schedules, pricing and revenue management decisions, among other paradigms. While the planning process varies across airlines, a representation of a typical timeline is illustrated in Figure 1.

The four classes described in the figure are by no means a comprehensive list of all decisions that are to be made, but summarize four of the most critical tasks, each of which are now discussed.
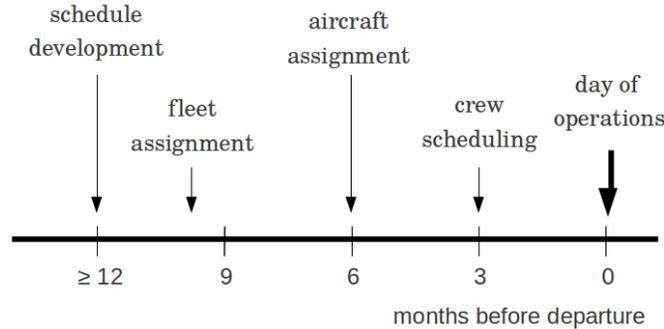
Figure 1: A Typical Airline Planning Process

1.2.1 Resource Scheduling Decisions

1.2.1.1 Schedule Development

Schedule development is the process of scheduling the stations served by a carrier and the frequency with which flights are to operate between each city pair. Once frequency is determined, then flight schedules are built that assign each flight to a scheduled departure and arrival time that define the eligible itineraries that are given by the passenger connection times.

1.2.1.2 Fleet Assignment

The fleet assignment problem assigns a fleet (or equipment) type to each flight in the schedule. The principal objective is to maximize profit by most appropriately equating supply with demand. Fleet assignment has received considerable attention in the academic literature for over two decades.

Abara [3] was the first to publish significant results of applying fleet assignment at American Airlines in which annual savings in excess of $100 million were observed.

Hane et al. in [63] introduce a basic Fleet Assignment Model (FAM) which has been widely accepted. Their problem seeks to maximize profit (which models stochastic demand) subject to the following classes of constraints:

1. assignment: every flight must be assigned to precisely one fleet

4

2. flow balance: every node in the time-space network must preserve flow conservation

3. fleet count: the total number of fleets being utilized must not exceed the number of available aircraft within each fleet.

Several variants of the basic FAM framework have been explored enhancing various features. For example, Clarke et al. in [36] study incorporating maintenance and crew considerations. Other studies have incorporated network effects of passengers in order to better estimate revenue. Such models, referred to as origin and destination FAM (ODFAM) or itinerary FAM (IFAM) are studied in Lohatepanont [79], Barnhart et al. [23], and Jacobs et al. [68].

1.2.1.3   Aircraft Routing and Maintenance Planning

Given the FAM solution, the aircraft routing problem assigns individual aircraft to operate each flight leg. The problem is usually modeled as a feasibility problem that ensures each aircraft respects the FAA-mandated requirements on aircraft maintenance.

Work on aircraft routing is seen in Soumis et al. [109], Desaulniers et al. [47], and Talluri [114].

1.2.1.4   Crew Scheduling

Given the flight schedule, crew members are to be assigned to cover flights in this phase of the planning process. From the schedule a set of *duties* are generated that give all sequences of flights that may be operated within a given time interval for crew members. In a domestic carrier, a duty typically represents a single day of flying. For long-haul flights, a duty may be comprised of two days. Most crew assignments span multiple days that concatenate duties into a *pairing* that typically consists of two to four duties which are to begin and end at each crew member's assigned base. Both duties and pairings are subject to a set of rigid legality requirements that are imposed by regulations mandated by the FAA, idiosyncratic airline rules, and union rules. Connections between duties must allow for sufficient rest.

The Crew Pairing Problem (CPP) is a set partitioning problem that seeks to assign the set of pairings to cover all flights at minimum cost. The cost of a pairing is generally

defined to be the maximum of three components: the sum of operating costs of the duties that comprise the pairing, the total time away from base, and a minimum guaranteed value that depends on the cost per duty multiplied the number of duties. The CPP is typically solved by column generation as the number of pairings may be in the billions (see Hoffman and Padberg [64], Shaw [104], and Barnhart et al. [22]).

Once the set of pairings to be flown are given, they are ordered together in a feasible sequence allowing for sufficient rest in between consecutive pairings, and allow for additional tasks such as training or vacation over a period of time, usually spanning close to one month. This is known as the *rostering* problem. Individual crew members then are assigned to specific rosters usually by a bidding process in which each crew member states his or her rostering preference to be assigned. Among North American carriers, roster assignments are made on a seniority basis, while most European carriers assign rosters on more of a fairness criterion in which employees rotate through the set of rosters. In the operational environment, the rostering problem becomes substantially more complex as re-scheduling decisions within a crew's pairing affect the given roster, and there is a chance that the original schedule is infeasible with the modified roster.

1.2.1.5   Dynamic Pricing and Revenue Management

A large body of research has been devoted to studying revenue management within the airline industry. Revenue management attempts to maximize revenue by achieving an optimal allocation of seats to fare classes. Fares are priced by taking both aggregate and idiosyncratic factors known as the dynamic pricing problem. Distinguishing lower-yield leisure passengers who make their bookings relatively early from higher-yield business passengers who make their bookings much closer to the date of departure is one fundamental characteristic that every airline will have devoted substantial resources to. The problem has become more granular recently distinguishing each individual fare class. Overviews of dynamic pricing and revenue management are given in Belobaba [25], Weatherford [125], McGill and Van Ryzin [84], and Talluri and Van Ryzin [115].

1.2.1.6   Other Relevant Paradigms

The preceding components of the airline planning process are just a few of the many problems that airlines must plan for during the day of operations. These modules will have a strong connection to the work in this thesis. The following include other relevant planning problems that are prevalent throughout the industry.

Manpower planning: Crew scheduling, as discussed above, typically refers to in-flight crew (i.e., pilots and flight attendants). Scheduling ground crew staff including gate agents, baggage handlers, and reservations agents in a manner that maximizes their utilization is an important component to managing efficient operations.

Gate assignment: Assigning aircraft to gates has important implications for connecting traffic and hence revenue opportunities. Moreover, gating solutions impact runway operations, and have a fundamental impact on operations.

Flight planning: While traditional airline scheduling is concerned only with the times at which a flight departs and arrives, four-dimensional flight planning is playing an increasingly important role in operations, particularly as next generation air traffic control systems mature. Flight planning has a particular importance in fuel conservation.

1.2.2   Integration of Resource Planning

While airline planning models have been studied for decades, their use in practice is largely constrained by computing capabilities. Because of the size and complexities associated with these models, exact methods to solve such problems are often impractical. Therefore airlines often rely on heuristic (or even manual) methods in their resource planning.

Another important observation of OR models in airline scheduling is how problems are usually solved independently, often said to be solved in 'silos'. For instance, the fleet assignment problem is solved independently of the crew pairing problem. However as computing has become more powerful, much of the contemporary research within aviation applications of OR has focused on integrating various components of the scheduling process. Integrating

7

FAM with maintenance has been studied in Clarke et al. [36] and Barnhart et al. [20]. Integrating crew and aircraft routings is of interest to airlines considering less turn time is required as oppose to crews who have to change aircraft at a given station. Cordeau et al. [45], Cohn and Barnhart [41], Mercier et al. [85], and Gao et al. [56] are four studies exploring this class of integration. Incorporating FAM with crew scheduling has is found in Barnhart et al. [24] and Sandhu and Klabjan [100].

1.3   Irregular Operations

Because decisions conducted in the planning stage take place months in advance of the day of operations, decisions made are done in a frictionless environment. Of course, operations are rife with frictions caused by a myriad of reasons. Most causes of irregularities stem from at least one of the following sources:

- weather disturbances that reduce the flow rate into or out of a sector

- mechanical failure that precludes the use of an aircraft

- Air Traffic Control (ATC) restrictions that reduce the flow of traffic

- propagation delays caused by a single flight that has cascading effects to subsequent flights

- network effects caused by disrupted aircraft or crew members

The next chapter explores the various causes of irregularity in greater depth. Introducing sophisticated optimization techniques within the operational environment has been studied, but much less extensively than problems within the planning environment. While some research has been conducted to various facets of the overall problem, airlines have been slow to adopt them in practice. The most likely reason is because these solutions that focus on a sole component do not consider important constraints from other inputs. For example, a crew recovery module may exist, but solutions from the model may not be compatible with the aircraft. Thus, manual procedures are still relied heavily upon in practice in spite of the advancements in computing that enable more advanced procedures. This thesis introduces

an approach that considers all aspects of recovery in an integrated fashion. It is shown this solution can be delivered in a reasonable time even in the presence of rigid constraints imposed by the operational environment.

1.4 Structure of this Thesis

There are 6 chapters that comprise this thesis. Chapter 2 provides an overview of the causes of disruption, airline disruption management, and contemporary issues involving irregular operations at airlines. A model of airline recovery is studied in Chapter 3, that is formulated and solved in an integrated manner combining flight, aircraft, crew, and passenger re-scheduling decisions. Solving such a large and complex model within a reasonable runtime is aided by methodological advancements that can expedite the solution. Chapter 4 explores one such idea. Namely, it is examined how row and column generation can be solved in a simultaneous manner. The core of the integrated model relies on Benders' decomposition. While this algorithm is well-known and ubiquitous throughout airline planning, it can exhibit slow convergence. One possible procedure for expediting the algorithm by to strengthen a standard Benders' cut which is discussed in Chapter 5. Conclusions and suggested areas for further research are presented in Chapter 6.

CHAPTER II

AN OVERVIEW OF OPERATIONS

The purpose of this chapter is threefold. First, trends and causes of irregular operations are studied in Section 2.1. Secondly, the costs associated with irregular operations are studied in Section 2.2. In light of these findings, an important consequence of legislation in the enactment of the Passenger Bill of Rights is discussed in Section 2.3. In order to model airline recovery, it is necessary to understand the processes governing disruption management employed at airlines. This third goal is presented in Section 2.4.

2.1 Analysis of Delays and Cancellations

Here some stylized facts and observations concerning the behavior of delays and cancellations are presented. These facts seek to only summarize publicly available data. For a more comprehensive analysis, the reader is referred to studies by Zhu [134], Bratu and Barnhart [31], or Ball et al. [18]. Unless noted otherwise, all data is from the Airline On-Time Data published by the U.S. Department of Transportation Bureau of Transportation Statistics [120], and includes only domestic flights.

2.1.1 Traffic and Disruption Behavior

Figure 2 shows the total number of U.S. domestic passenger-revenue flight operations from 1996 to 2010. While air traffic is in general noisy and highly seasonal, there has clearly been an upward trend in the total number of flight operations. The effects of the September 11 attacks are clearly seen in the immediate decline, but an upward trend before and after the shock are clear. This upward trend is anticipated to continue by industry experts. In their forecasts for the future growth in global air transportation traffic, Boeing [2] and Airbus [1] expect air traffic to grow both within North American and globally. From 2009 through 2029, Boeing and Airbus estimate annual traffic within North America to grow by 2.8% and 3.3%, respectively (whilst growing 5.3% and 4.8% globally). This rise is in the presence

of a global air transportation system that is already operating at or near capacity at most major stations. This leads to the first important observation.

Observation 2.1.1. *As the growth in air transportation is anticipated to outpace that of capacity, the National Airspace System (NAS) is expected to be placed under increasing strain. The likelihood of delays and cancellations, therefore, is likely to increase, cetris paribus.*
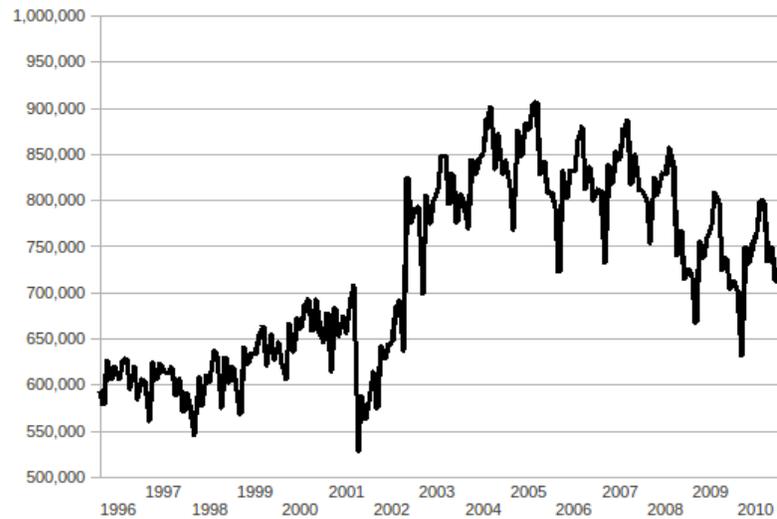
Figure 2: Domestic Passenger-Revenue Flight Operations, 1996 – 2010

Figure 3 summarizes two important metrics for the behavior of U.S. domestic flights from 2003 through 2010. Panel 3a shows the share of all flights that were on-time (defined as all flights arriving within 15 minutes of the original scheduled arrival time). Panel 3b shows the cancellation rate over the same period. In both figures, the shaded regions correspond to the first and fourth quarters of each year, when inclimate weather is more likely to occur from snowstorms.
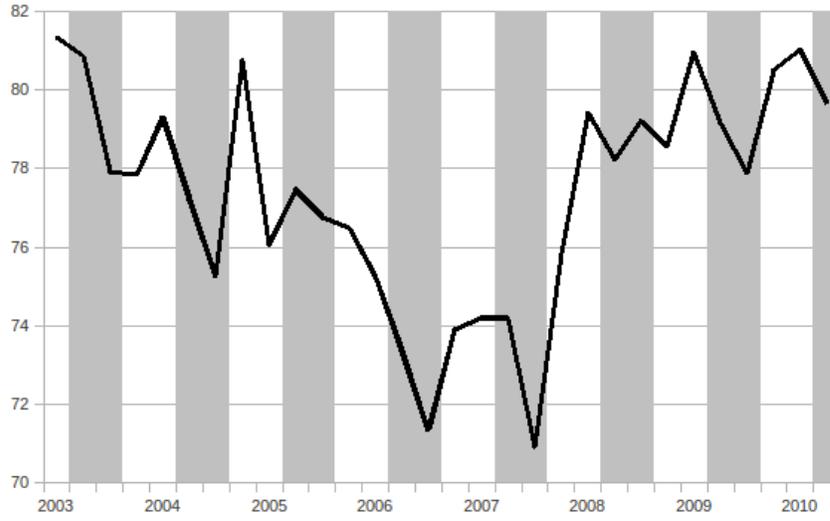
(a) On-Time Rate
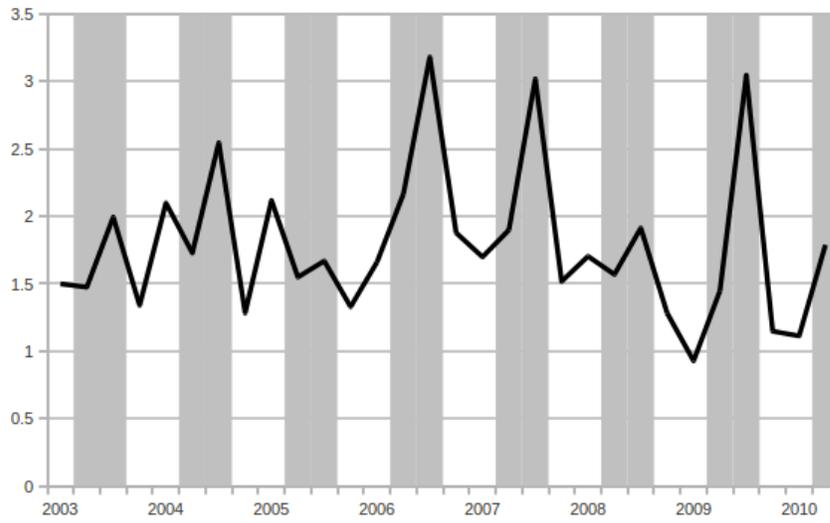
(b) Cancellation Rate

Figure 3: U.S. Flight Performance, 2003 – 2010

2.1.2 Causes of Irregularity

Irregularity is caused by a myriad of possible events. The FAA Bureau of Transportation Statistics has discretized delays and cancellation into five classes shown in Table 1.

Table 1: BTS Disruption Classification

Delay Class	Description	Examples
Air Carrier Delay	Caused by factors that are within the control for an airline	maintenance disruptions, fueling, baggage loading
Extreme Weather Delay	Disruptions caused by actual or forecasted severe meteorological disturbances	blizzards, hurricanes, tornadoes, severe thunderstorms
NAS Delay	Frictions resulting from the NAS	non-extreme weather disruptions, ATC, airports, traffic congestion
Security Delay	Delays that stem from irregularities in the security of a flight or airport	large terminal area disruptions, security breaches, excessive queues at security areas
Aircraft Arriving Late	Events attributable when a flight is disrupted as a result of a delay to the preceding flight utilized by the same aircraft	inbound flight of 101 affects causes a delay in flight 102 where the same aircraft operates both flights

Figure 4 illustrates the share of each of these five classes of delay from 2003–2010. It is readily seen that NAS delays, air carrier delays and late arriving aircraft account for around 95% of all delays and cancellations.

These classes give a very coarse way to identify causes of delay. In particular, weather delays are found within NAS delays, extreme weather delays, and as late arriving aircraft due to weather. By summing the explicit weather-related components within these three classes, the share of delays attributable to all weather events can be estimated. Figure 5 shows the share of U.S. weather-related delays for the total number of operations as well as delay minutes from 2003–2010.

Given the impact weather has on flight performance in the U.S., the second important observation is given as follows.
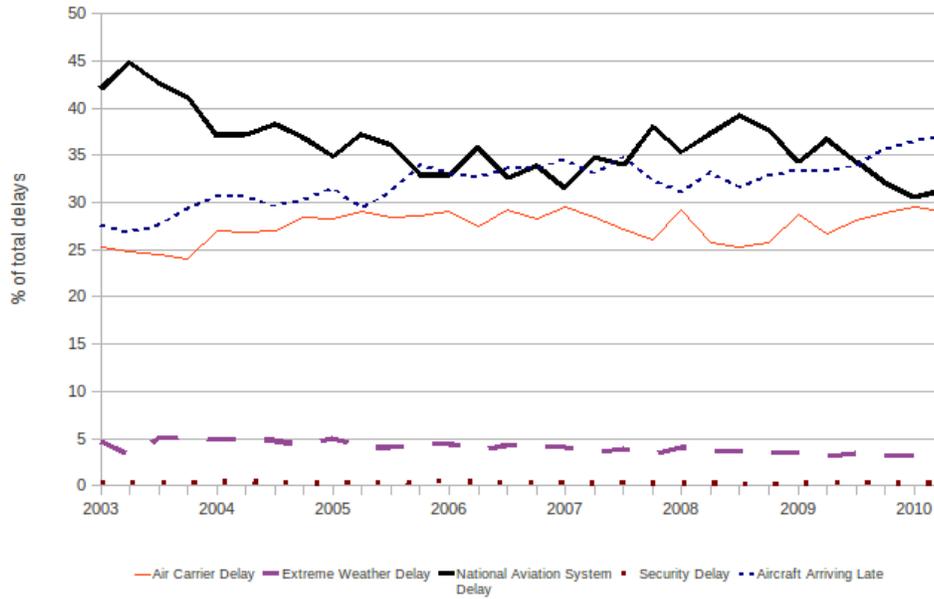
Figure 4: Causes of Delay, 2003 – 2010

Observation 2.1.2. *Delays and cancellations are strongly influenced by weather in the U.S., accounting for over 40% of both the number of disruptions as well as the length of delays.*

The preceding figures come from data aggregated across all U.S. domestic carriers. Naturally the share of each cause of disruption is subject to variability across airlines as differences in geographic concentration, fleet type, and flight schedules are idiosyncratic properties unique to each carrier. For example, Kenya Airways, who operate mostly within Africa from their Nairobi hub, reported that explicit weather delays account for only 3% of all disruptions (Schellekens [101]). Figure 6 shows idiosyncratic differences in the causes of delay from two U.S. carriers within 2010. Most notably late arriving aircraft and NAS delays account for significant variations of delay.

14
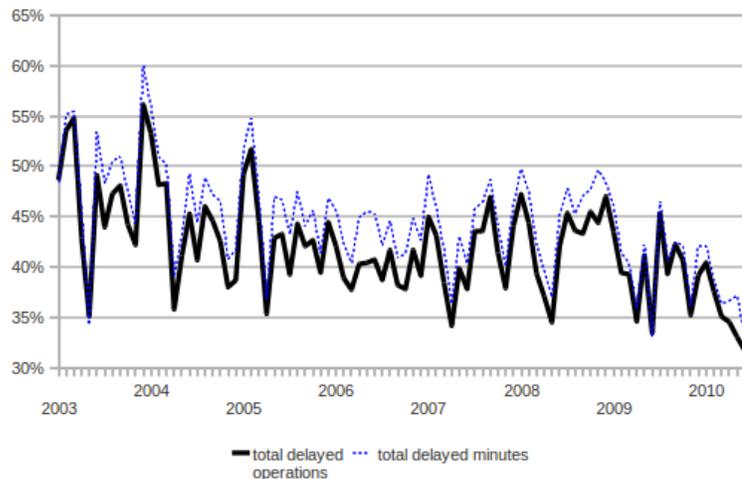
Figure 5: Share of Weather-Related Disruptions, 2003 – 2010

Observation 2.1.3. *The causes of delays are subject to variability across airlines. Factors such as geographic presence, schedule density, and air traffic control are fundamental drivers to the causes of airline delays.*

2.2 Costs of Irregularity

This section focuses on identifying the costs of irregularity incurred by airlines. The costs of an operational disruption extend far beyond than just the airline itself. There are also significant costs incurred by passengers, and macroeconomic effects that are reflected in opportunity cost of lost productivity.

2.2.1 Airline Costs

Given that the growth in the demand of air transportation is anticipated to outpace that of capacity in a system already operating at or near capacity, disruption management is expected to play a crucially important role in the future procurement of air traffic systems. While airlines have a natural desire to schedule their resources at a high utilization, introducing some slack into the system is often done so as to absorb delays in the presence of irregularities. The question of how much slack to add at which times is a nontrivial problem,

15

3.86%

28.02%

12.89%

0.27%

54.95%

Air Carrier Delay   Aircraft Arriving Late   Security Delay

National Aviation System Delay   Extreme Weather

(a) Airline 'A'

4.65%

35.51%

28.28%

0.04%

31.52%

Air Carrier Delay   Aircraft Arriving Late   Security Delay

National Aviation System Delay   Extreme Weather
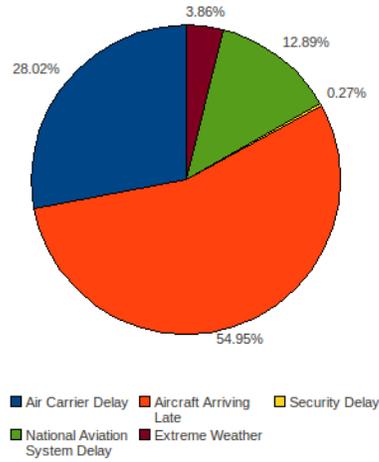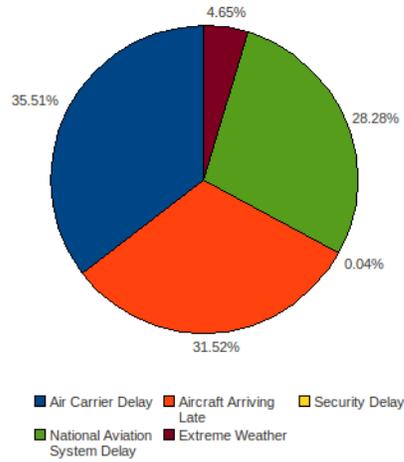
(b) Airline 'B'

Figure 6: An Intra-Airline Comparison of Causes of Delay, 2010

16

also known as robust scheduling, is strongly related to this work in which is addressed later in this thesis.

2.2.1.1 Direct Costs

To ameliorate the likelihood of disruptions airlines typically add extra buffer times, particularly at stations or flight segments in which the airline has a priori knowledge that are prone to delay. Associated with every flight connection is a minimum turn time that is represents the shortest ground time needed between the two segments. Consider a routing with consecutive flights i and j. The *planned turn time* PTT_{ij} represents the difference between the scheduled departure of flight j, SDT_j and scheduled arrival time of flight i, SAT_i, so

$$PTT_{ij} \equiv PDT_j - PAT_i. \tag{1}$$

The scheduled buffer SB_{ij} is then defined as the excess turn time relative to the minimum turn time MTT_{ij} (which depends on the timing of flights, ground resources, and other factors), or

$$SB_{ij} = PTT_{ij} - MTT_{ij}. \tag{2}$$

An illustration of this concept is shown in Figure 7 where flights 101 and 149 appear in consecutive segments within a scheduled routing. While the minimum turn time for this connection is 30 minutes, the planned turn time is 70 minutes giving a buffer time of 40 minutes. The relatively large buffer time may be attributable to congestion at LHR that increases the likelihood of delay. Buffer times are likely to be shorter at stations with less activity.

Clearly there is an inherent tradeoff through adding scheduled buffers. A longer buffer abates the risk associated with disruptions, and recovers at least as fast as a schedule without buffers. However, there is an opportunity cost associated with buffers, namely, lost revenue that could have been gained by scheduling only minimum turn times. The first cost is therefore the net effect of opportunity costs (if little or no disruption) lost by robustness and total recovery costs associated with irregularity. Ball et al. [18] study both the cost
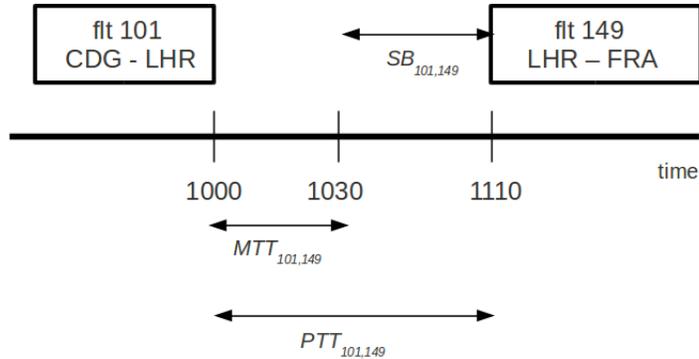
Figure 7: Incorporating Scheduling Buffers to Increase Robustness

of flight delay against the schedule as well as the costs induced by buffers. They estimate that in the U.S. alone in 2007, the industry-wide costs are at least (depending upon two approaches) \$4.6 billion in delays against schedule and \$3.7 billion from buffers.

In the event of an irregularity a number of critical costs are incurred. Crew costs are of particular concern to large disruptions. As mentioned in Section 1.2.1.4, crew members are generally paid in accordance to the maximum of three values over their pairing: the sum of operating costs of the duties that comprise the pairing , the total time spent away from their crew base, and a minimum guaranteed value. As a disruption prolongs the actual length of the pairing, the first two of these three components become more costly thereby driving up the expected cost of the pairing. Moreover additional hotel rooms may be needed, as well as the use of reserve crew, which can carry significant costs depending on the specified cost structure of the airline. Moreover excess fuel costs and environmental impacts also of importance to total airline costs as argued by Cook et al. [43].

2.2.1.2 Indirect Costs

As mentioned the introduction of scheduling buffers is designed to absorb delays throughout the flight network. *Delay propagation* occurs when the delay or cancellation of one flight

cause a subsequent flight leg to be disrupted. These indirect costs incurred by the airline can be extraordinarily high, particularly if the airline employs an aggressive scheduling procedure allocating insufficient buffer times to their scheduled operations. By letting AAT_i and SAT_i denote the actual and scheduled arrival times of flight i, respectively, the arrival delay for flight i as may be computed by $Delay_i^{\mathrm{arr}} \equiv AAT_i - SAT_i$. This, coupled with the notation from above, allows one to estimate the propagation delay for flight j, or PD_j as

$$PD_{ij} = \max\left\{Delay_i^{\mathrm{arr}} - SB_{ij}, 0\right\}. \tag{3}$$

Figure 8 illustrates an example of propagation delay on a time-space network from Figure 7 where flight 101 is delayed one hour. The direct impact on flight 101 is trivially one hour. Using the notation above, $SB_{101,149} = 40$ minutes, $Delay_{101}^{\mathrm{arr}} = 60$ minutes, and thus $PD_{101,149} = \max\left\{60 - 40, 0\right\} = 20$ minutes.
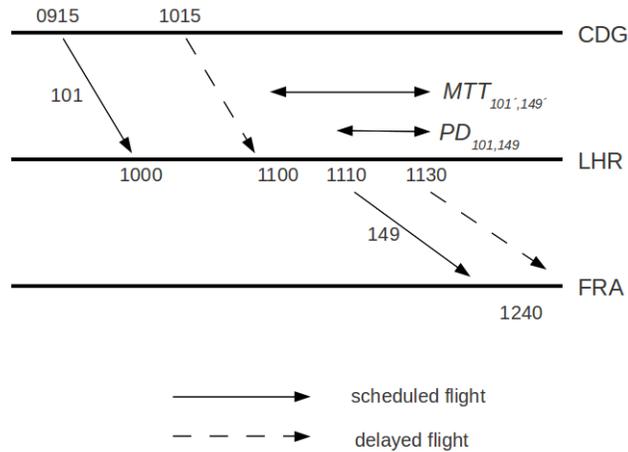
Figure 8: Propagation Delay from a One Hour Delay (Flight 101)

Delay propagation may also result in crew members that are late on an inbound flight. One common practice to reduce delay propagation is to add more slack to flight connections in which the inbound flights have a higher likelihood of delays. Managing scheduling buffers has generated some interest in the literature, as seen in papers by Lan et al. [75] and Ahmad Beygi [6].

2.2.2 Passenger Costs

Much like direct operating costs that are incurred by airlines to delays and cancellations, passengers also have costs associated with delays from both direct impacts as well as delay propagation. Scheduling buffers are also costly for passengers as they pose the same opportunity cost by reduced frequency. There are also important other costs paid by passengers that are less tangible. One is the costs borne by missed connections of those passengers whose itineraries contain multiple flights. Another is the loss of passenger goodwill or disutility that may cause an overall loss of market share. A third includes costs internalized by idiosyncratic passengers that stem from the loss of productivity or leisure events. Quantifying these subcomponents to passenger costs is an arduous task as they rely on estimations of non-observable data. Cook et al. [43] suggest one way to use dynamic cost indexing to estimate aggregate passenger costs. They estimate that a 90 minute delay for one long-haul flight for a large European carrier typically incurs an aggregate passenger cost of €12,077 (equivalent to approximately $17,100). Ball et al. [18] use an empirical method to analyze the total cost of passenger delays of the preceding components. They estimate that total passenger costs amounted to $16.7 billion in the U.S. alone in 2007.

2.2.3 Macroeconomic Costs

Costs emanating from both airline disruptions and scheduling buffers have significant macroeconomic costs associated with them. As supply chains have increasingly proliferated to a global scale, delays to aircraft carrying freight are prone to disrupt supply chains and other means of production. There are also economics costs of the loss of productivity both in direct costs (missed meetings, etc.) and indirect opportunity costs. With respect to scheduling buffers, the increased buffer times reduces frequency to markets. The loss of frequency induces higher airfare for passengers and cargo alike, both of which may result in higher prices passed through to consumers.

The Joint Economic Committee (JEC) reported (in [71]) that the impact on U.S. Gross Domestic Product amounted to $9.6 billion in 2007.

2.2.4 Overall Costs

Table 2 summarize estimates of the total cost of irregular operations in the U.S. in 2007 from two often cited sources.

Table 2: Delay Cost Studies for U.S. Carriers, 2007

Component	Cost (billions $)	
	Ball et al. [18]	*JEC* [71]
Total costs to airlines	10.5	19.1
Total costs to passengers	16.7	12.0
Total loss of GDP	4.0	9.6
Total Costs	**31.2**	**40.7**

2.3 *Passenger Bill of Rights*

Effective April 29, 2010 the U.S. Department of Transportation (DOT) enacted a resolution (H.R.624/S.213) aimed at reducing excessive tarmac delays. The so-called 'Passengers Bill of Rights' fines airlines up to $27,500 *per passenger* whom experience a tarmac delay in excess of three hours on all U.S. domestic flights whose (a) aircraft contains at least 30 seats and (b) airport of origin process at least 1.8 million enplanements per year. This poses a potentially substantial cost for airlines - with 84% load factors this can range from $1.1 million for a 50-seat regional jet to $5.4 million for a wide body Boeing 767. Similar versions of the bill have been proposed as far back as 1999 (according to Marks [82]). Enactment of the bill is likely attributable to several widely publicized isolated events of operations that have caused considerable, most notably the following events:

1. **American Airlines Flight Diversion (Austin, Texas, December 2006)** Late December 2006 the Dallas-Ft. Worth International airport (DFW) was experiencing large lightening storms and tornado warnings forcing a closure for nearly 8 hours. As a result American Airlines (which operates a hub out of DFW) diverted over 100 inbound flights, many to Austin International airport (AUS). One aircraft awaited on the tarmac for close to 9 hours (others were held for four hours) as the ground resources at AUS were insufficient to service the diverted flights. Multiple passengers

filed a lawsuit against the carrier, and motivated some passengers on board to begin a passengers rights group which has been lobbying Congress since the incident in late 2006.

2. **JetBlue Valentine's Day Disruption (New York, February 2007)** A winter nor'easter made its way through the New York City on February 14, 2007 resulting in a mix of snow and rain. Forecasters in the operations control center estimated the temperatures would be sufficiently warm for the precipitation to convert purely to rainfall thereby having a relatively small impact on its operations. Most other airlines operating out of JFK had decided to cancel their flights. The forecasts, however, were not accurate and freezing rain continued to fall prohibiting takeoff from JFK. Several planes that pushed back from their gates no longer had gates to return to as inbound aircraft had arrived after push back. Meanwhile the vehicles that were to tow the stranded aircraft were inoperable as they were frozen to the ground. Nine JetBlue aircraft were stranded on the tarmac in excess of six hours; one aircraft spent nine hours before passengers were able to deplane and were shuttled back to the terminal. The problem was exacerbated as problems took place at JFK, the main JetBlue hub. The airline ended up cancelling 47% of all scheduled operations February 14–15 in addition to the lengthy tarmac delays. The incident became widespread throughout various media outlets in which JetBlue suffered considerable passenger goodwill, which the airline enjoyed high customer satisfaction hitherto.

3. **RST Airport Tarmac Delay (Rochester, Minnesota, August 2009)** An August 8, 2009 ExpressJet flight from Houston (IAH) to Minneapolis (MSP) was placed in a holding pattern for 30 minutes when heavy rain forced a diversion to Rochester International Airport (RST). While the aircraft was awaiting clearance for takeoff to MSP, the aircraft needed refueling during which the heavy rains moved south affecting RST operations. By the time the storm had cleared, the crew members had reached their maximum allowable duty time thereby rendering the crew illegal even for the short remaining flight leg. Because RST served only two airlines with ExpressJet

not being one, the aircraft could not unload passengers directly from a jet bridge. The security staff had left the airport which was initially cited as one reason why the passengers were not allowed to immediately exit the aircraft (although it has been debated if this was a legitimate reason for keeping the 47 passengers on board). Despite the claim by ExpressJet that their crew members tried to allow the passengers to exit the aircraft, the passengers awaited on the ground for six hours overnight until TSA staff had begun their duty the next morning. Passengers then boarded after the new crew had arrived from Milwaukee. The incident garnered attention worldwide in which passenger rights groups used the incident to lobby for a bill of rights.

In addition to the airline fines, the bill also requires that food, water, lavatories, and medical assistance are provided for the passengers during the ground delay. A revision of the initial bill added other provisions effective August 23, 2011 that extend excessive tarmac delays to all airports, and all flights including international flights with a four-hour threshold. Moreover, minimum fines were introduced for all passengers who are denied boarding, as well as lost baggage.

2.3.1 An Analysis of the Three-Hour Tarmac Rule

The introduction of the bill drew much commentary in the press, generally receiving favorable feedback from passenger rights groups and other consumer advocacy groups. Aviation experts were generally less sanguine about the efficacy of the rule believing it would lead to preemptive cancellations in order to avoid paying the high costs associated with delays.

Figure 9 shows the total share of flight delays experiencing tarmac delays in excess of three hours from January 2009 through June 2011. The vertical line represents the date the three-hour tarmac rule went into effect. As seen by the figure, three-plus hour tarmac delays have declined since the implementation of the legislation. As of June 2011, no airline has been fined for violating the rule (while there have been 20 incidents of exceeding tarmac delays through the end of April 2011, no incident qualified under the terms in the bill for the fine). However, the figure also shows that these delays have occurred infrequently - never exceeding 0.05% of all flights. Long taxi-out times (the primary driver to excessive

23

delays) have also dropped considerably, falling 47 % from May-November 2010 compared with the average over 2005 - April 2010 (see Marks [82]). In light of these findings, the DOT has fervently defended the ruling.
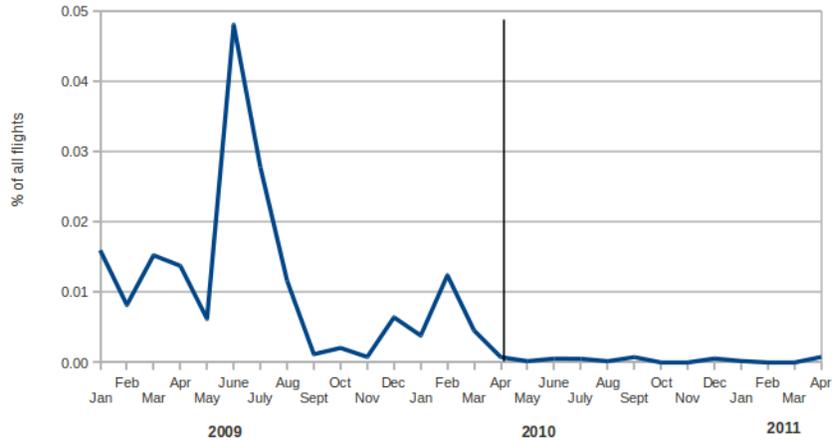
Figure 9: Share of Tarmac Delays Exceeding Three Hours

Figure 10 addresses the concerns raised by skeptics of the rule who feared that cancellations would increase to avoid the risk associated with long tarmac delays. Indeed, the cancellation rate has risen from 1.67% to 1.95% before and after the rule, respectively, from January 2009 through April 2011. Significant snowstorms in the northeast in December caused a number of cancellations, and many experts have argued that the legislation has alone caused this number to be higher than it would have been otherwise.

Given that the rule is still relatively new, making a strong inference is possibly spurious. However, some have conducted thorough cost-benefit analyses of the net effect of the rule. Marks [82] reports that even excluding the large snowstorms from December 2011, there were over 480,000 impacted passengers from the rule year-over-year from May-November 2009 to 2010. He argues that the number of fewer passengers who are stranded on the tarmac in excess of three hours is more than offset by the number of passengers affected by cancellations, and aircraft returning to the gate in order to avoid the three-hour limit. A similar exposition is given in Jenkins and Marks [70]. These are just two studies that seek to understand the net effects of the rule; there will no doubt be more studies as more data
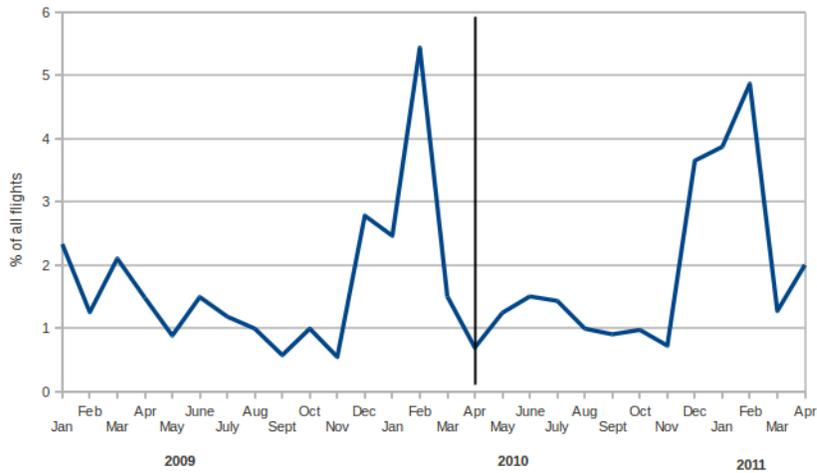
Figure 10: Cancellation Rates Before and After the Tarmac Rule

becomes available.

In addition to an increase in the cancellation rate, critics of the rule also argue the following:

1. In the event an aircraft nears the three hour threshold, it may return to the gate whereby the process renews itself as continuous - and not cumulative delay is tracked for passengers. Moreover, the aircraft loses its position in the departure queue exacerbating the total delay if a departure occurs. Because gate returns have a higher likelihood to be cancelled upon return to the terminal, the passenger recovery process becomes more difficult as potentially hundreds of new passengers have to be reaccommodated on new itineraries that are likely to already contain high load factors.

2. The rule does not apply to several other realistic scenarios that may cause excessive tarmac delays. For instance, in lightening storms where ground crews are prohibited from being on the tarmac due to the risk of exposure, the flight does not qualify for paying the penalty. The same is true for events in which the captain declares there is a safety or security issue that prevents movement of the aircraft from the tarmac.

25

2.4 Airline Disruption Management

The current practice of disruption management employed at airlines is now examined. This section begins with an overview of the centralized decision-making enviornment, and how decisions are made in practice. Even in spite of all the advancements made in both theory and computation, operational decisions made in the presence of a disruption are largely conducted manually. The reason for this is discussed which will serve as additional motivation for the integrated recovery approach presented in Chapter 3.

2.4.1 Airline Operations Control Centers

Most every airline has a centeralized environment that oversees the daily operations involving the flight schedule, aircraft, maintenance events, crew members, airport resources, and air traffic management. While this environment has different names, one that is often used that will be used throughout this thesis, is that of an Operations Control Center (henceforth referred to as an OCC; although there are several variants whose acronyms include SOC, AOCC, and AIOC). Grandeau et al. [61] give an overview of the general processes found at an OCC, including a detailed analysis of a specific carrier's OCC. Other works discussing the role of OCCs are found in Pujet and Feron [93] and Clarke [38].

OCCs have played an increasingly important role after the September 11 terrorist attacks in which there was a closure of the NAS. Prior to that point, control centers were more fragmented with passenger handling often being conducted at the station level as oppose to a centralized environment. The following are some of the groups that play an integral role in an OCC:

- **Operations Control** is responsible for maintaining the flight schedule and managing the delay and cancellation of flights.

- **Flight Dispatch** is responsible for generating flight plans, and en route tracking of flights.

- **Aircraft Dispatch (or Maintenance Control)** manage maintenance events for all aircraft, and possibly reschedule maintenance activities for aircraft whose schedules

are disrupted. They also manage the utilization of spare aircraft that may be used in a recovery operation.

- **Meteorology** is responsible for creating weather forecasts in relevant areas within the airline's network in order for the OCC to plan their operations accordingly. They may recommend proactive flight operations strategies in order to avoid the cascading effects of delays.

- **Crew Operations** track individual crew members and ensure all rigid legality requirements are preserved over the course of each crew's pairing. In the presence of an irregularity, they are responsible for re-scheduling crew members subject to legality requirements, assigning deadheads to crews to be used in recovery, manage standby and reserve crew members, and hotel arrangements for stranded crew members.

- **Passenger Reaccommodation** manages passenger handling and generates new itineraries for those passengers whose original itinerary is broken by a disruption.

OCC resources have to coordinate with two other groups that play a fundamental role in disruption management. The first is Air Traffic Control Coordinators (ATCC) who interface between the OCC and FAA which communicate information about FAA ground delay programs to the airline. ATCCs may or may not be located within the OCC. The second group that strongly interacts with the OCC is Station Operations Control Centers (SOCC) who manage individual airport operations as discussed in Grandeau et al. [61]. A set of re-scheduling decisions may affect runway operations, gate planning, and passenger handling which are managed under the jurisdiction of the individual SOCCs.

The input-output process of an OCC is summarized at a high-level in Figure 11. U.S. carriers have multiple calls daily (typically three) with the FAA's Air Traffic Control System Command Center (ATCSCC) to discuss current-day operations in order to effectively plan for the NAS.

The internal structure within an OCC is generally highly fragmented, with groups responsible for the operating schedule, aircraft, crews, and passengers acting mostly in isolation of one another. Figure 12 shows an internal view of a typical OCC and how the
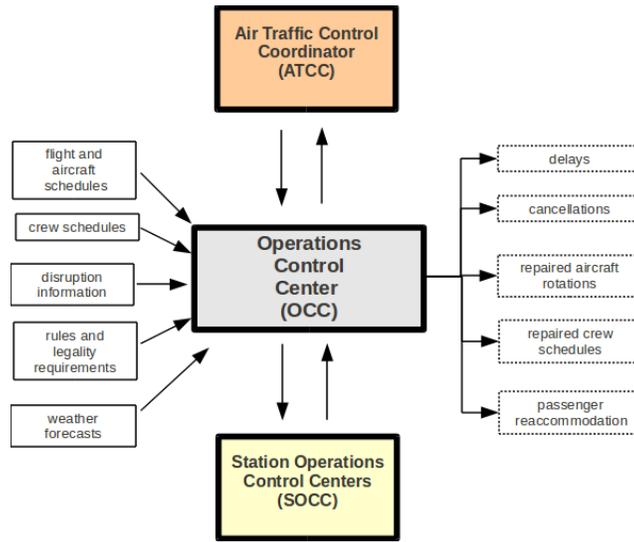
Figure 11: An External View of an OCC

decision from one group affect that of another.

In general, the core of the re-scheduling process is contained within the flight operations group who makes delays and cancellation decisions from the existing aircraft routings. This group must coordinate with the maintenance planning department to ensure any deviations from the aircraft routings meet all maintenance requirements at a minimum before the tentative schedule is passed to other groups. Once the original flight schedule has been augmented to reflect aircraft changes, delays, and cancellations, the candidate schedule is passed to the crew scheduling group who is responsible for creating any re-scheduling decisions for crew members. The schedule is also passed to the group responsible for the passenger reaccommodation process which is conducted by assigning each individual passenger to a new itinerary. Passengers are heterogeneous to the airline as higher-valued passengers are given priority to more attractive options over others to minimize the impact of the loss of passenger goodwill. If the re-scheduling decisions from the crew scheduling and passenger reaccommodation groups is deemed appropriate, the scheduling changes are made and transmitted throughout the OCC and other critical stafff. However, either group may deny the re-scheduling decisions from flight operations and the process begins anew.

28
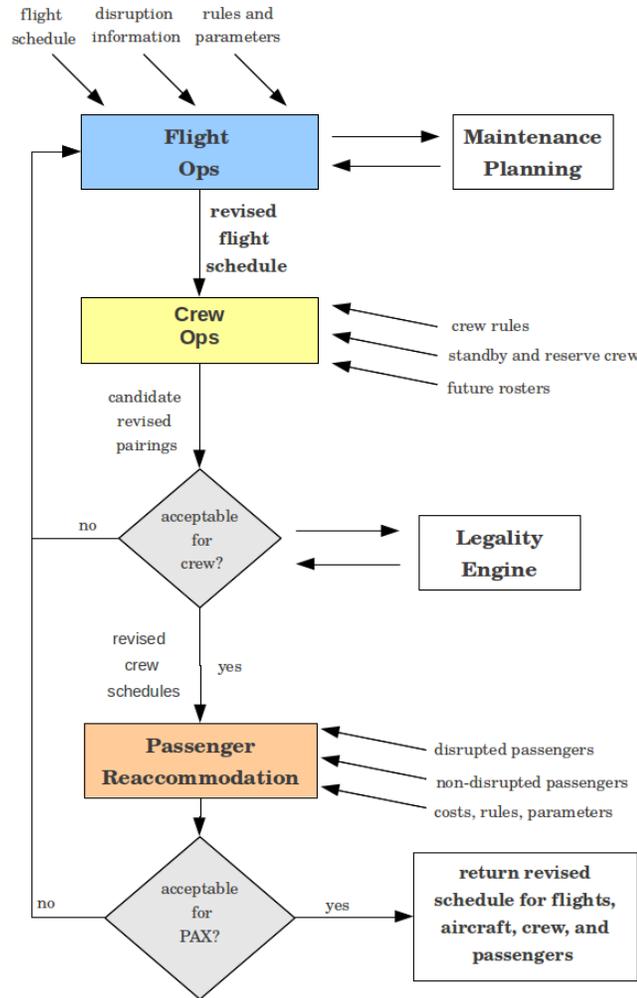
Figure 12: An Internal View of an OCC

2.4.1.1 Decision Support Tools

Few decision support tools used in operations have sophisticated optimization as many processes are still manual. OCC controllers typically rely on the use of a tracking tool that will monitor flights or crew members in real-time. These systems are usually in the form of a Gantt chart similar to that illustrated in Figure 13 for flight and maintenance operations. The user is usually allowed to manually drag and drop different activities scheduled for each resource. Some advanced systems may have OR support built within the tool that will allow for an automated solution to optimize the set of re-scheduling decisions subject to user-defined specifications concerning the parameters, objectives, and constraints.

29
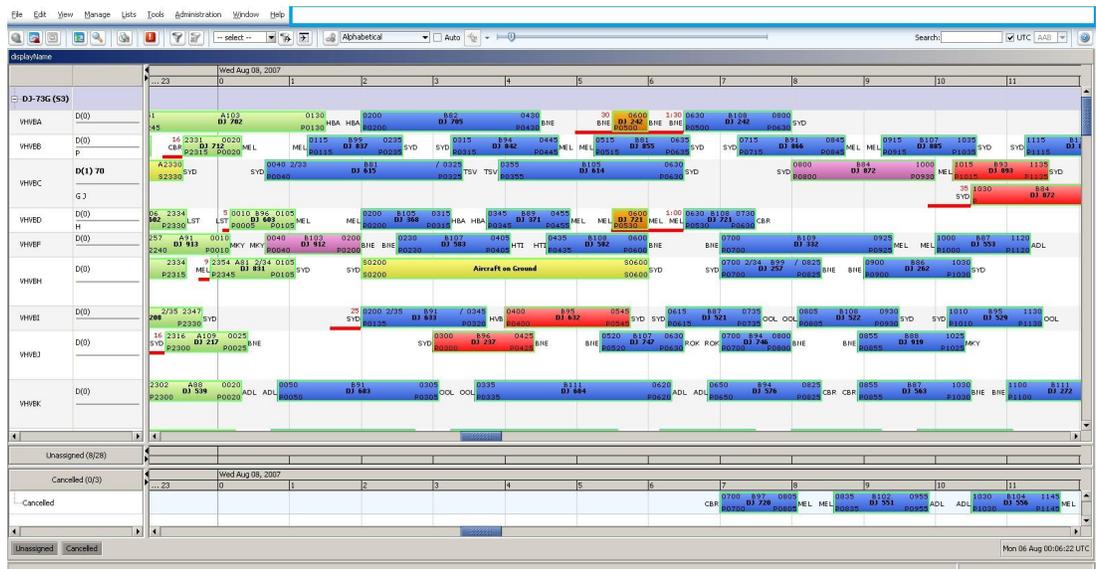
Figure 13: An Example of a Flight Tracking Tool

A number of decision-support systems employing mathematical programming techniques have been developed both by airlines and vendors. Such systems exist for re-scheduling aircraft rotations, crew schedules, and passenger itineraries in an isolated environment from the other components of the process. Due to the size and complexities of obtaining an integrated solution to all components, early work on employing mathematical programming techniques to solve the airline operational problem under irregularity has been sequential. Thus, at one extreme an OCC could employ such decision-making technology for each phase of the recovery process as seen in Figure 12. While this is plausible on a hypothetical level, this is not executed in practice. While a module may be used in production, it is likely at most only one of the three possible modules while the remaining components are constructed manually. The reasons that manual-based methods continue to play a central role in the disruption management process within the operational environment are attributable to at least one of the following two concerns:

1. The groups responsible for flight schedules, aircraft rotations, crew schedules, and passenger reaccommodations have their own set of objectives that are unlikely to be considered by other groups given the process is sequential. For example, an attractive solution for the flight operations group that minimizes the number of tail assignment

changes may be unattractive for the crew operations group due to the need to use an excessive number of reserve crews that drive the crew recovery costs above a reasonable threshold.

2. As the case with several industries, there is often times a reluctance to aggressively implement a decision-support system that reduces a decision-maker's autonomy. Many individuals whom possess in-depth knowledge of an airline's operations view any kind of black box software with skepticism.

Three following are three commonly used tactics often used by flight operations controllers in managing disruptions:

Knock-On Delays Given an aircraft whose routing contains a disrupted flight, one approach would be to delay all flights within the routing until the delay is absorbed by the scheduling buffer within the same tail. Such a strategy is known as *knock-on delays*, and is often employed for relatively small disruptions affecting a small number of aircraft, or when the objective of the flight operations controller is to keep as many aircraft routings preserved as possible.

Cancellation Cycles In hub-and-spoke networks aircraft rotations often exhibit cyclic behavior. For flights departing from a hub to a spoke, the subsequent flight in the rotation is usually returned back to the hub. Another common tactic in disruption management is to cancel all flights within at least one cycle. This strategy may become attractive when the operations coordinator wishes to return to the original schedule as soon as possible, or to ensure an aircraft is present at a given station to operate a strategic flight.

Tail Swaps It is common to temporarily switch, or *swap* aircraft to mitigate the total length of a disruption. For instance, if one aircraft exhibits an arrival delay, it may temporarily switch flying segments with another aircraft that is currently grounded at the arrival station. This is a commonly used recovery mechanism to minimize the total number of affected resources, although it is likely that the elapsed time taken to return to normal

operations will be greater.

Figure 14 illustrates these principles in a simple three station, two aircraft, nine flight example seen in Figure 14a. The thick black segment at CDG represents a closure forcing (at a minimum) the disruption to flight 103. Figure 14b shows a knock-on delay management strategy. Note that tail XYZ operates as scheduled since no flights are affected by the closure. The late arrival of flight 103 forces the disruption to flight 107. While flight 107 is late inbound to FRA, the slack absorbs the delay and flight 109 operates as scheduled. Figure 14c shows a cancellation cycle in which flights 103,107, and 109 are all cancelled. The aircraft would then remain on the ground to operate flight 103 the next day. Figure 14d shows the concept of a tail swap (in conjunction with a cancellation cycle) between aircraft ABC and XYZ. Because the late arrival of flight 103 forces a disruption to flight 107 using aircraft ABC, a tail swap occurs and flight 107 then undisrupted when operated by tail XYZ. To return XYZ to LHR, the tail also covers flight 109. In exchange, tail ABC covers flight 215 which was originally scheduled to be operated by XYZ.

The preceding discussion illustrated three commonly used approaches by airlines to manage their operations under irregularity. It is common to use a hybrid approach utilizing multiple such methods in practice. Consequently the problem of finding the optimal strategy meeting an objective is combinatorial in nature for only the problem of flight rescheduling. Considering similar strategies with crew and passenger considerations makes an already difficult problem even more complex, which also explains why manual processes for recovery are used in operations.

2.4.1.2   Disruption Management Objectives

As previously mentioned, one unattractive feature of sequential recovery methods is the reliance on coordination with different scheduling groups whose objectives may conflict with one another. An integrated approach addresses some of these shortcomings if a well-defined and mutually agreeable objective is agreed upon which solutions are to be evaluated and optimized. The following comprises a list of goals that are typically used by an airline in a recovery process. Attractive solutions are generally sought that seek to:

(a) Original Schedule

(b) Knock-On Delays

(c) Cancellation Cycle

(d) Tail Swaps
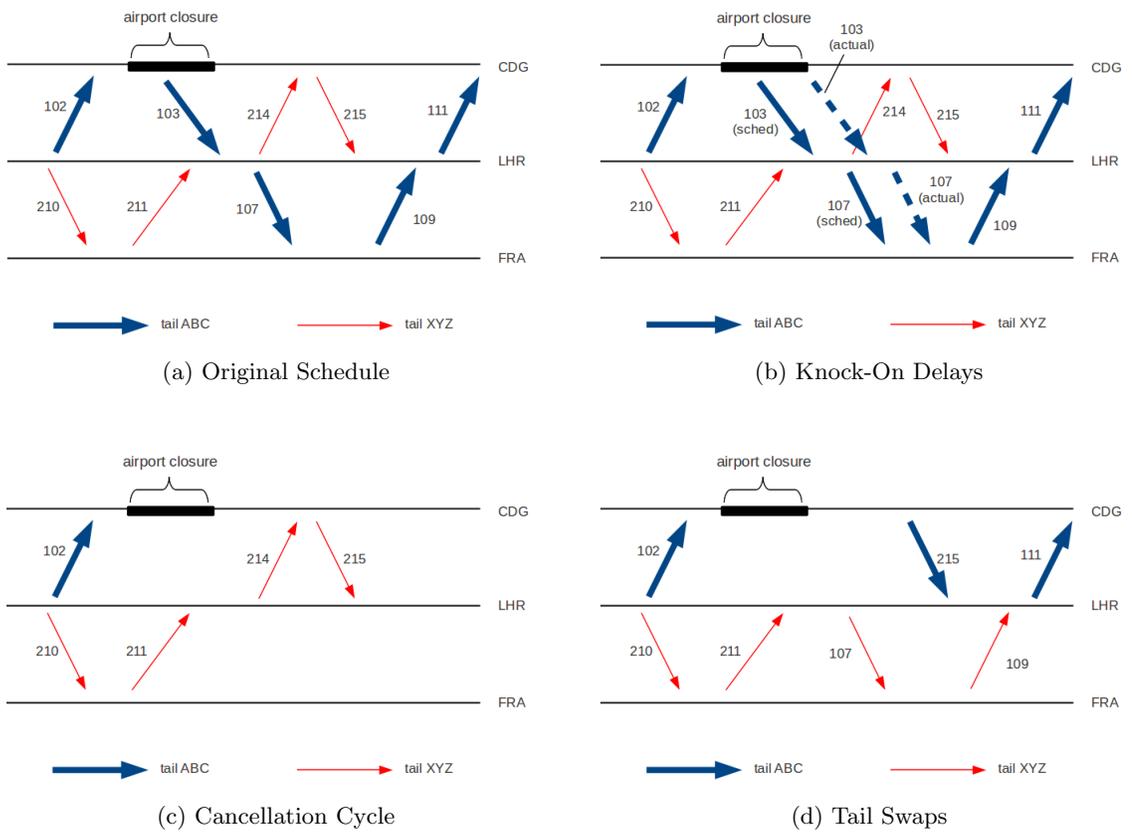
Figure 14: Commonly Employed Disruption Management Strategies

- minimize the time required to return to the undisrupted schedule

- minimize total aggregate operational costs (considering aircraft, crews, and passengers)

- minimize total passenger delay, or loss of passenger goodwill

- minimize aggregate flight delay

- minimize total deviations from the published schedule

- maximize the total number of passengers who can be accommodated to the new schedule

Finding a solution meeting all of these objectives is unlikely for most reasonable size disruptions. Moreover, tradeoffs are likely to exist making evaluating the quality of a proposed recovery plan ambiguous. For instance, a given delay and cancellation plan may be attractive as all resources are back on plan quickly, but may induce a high operational cost. Thus weighing various criteria may be necessary to evaluate such tradeoffs.

CHAPTER III

AN OPTIMIZATION APPROACH TO AIRLINE INTEGRATED RECOVERY

3.1 Introduction

The airline industry has been one of the biggest beneficiaries of advancements made in the application of advanced optimization methodologies. Fleet assignment, aircraft scheduling, crew scheduling, dynamic pricing and revenue management, and other paradigms have received considerable attention in both industry and academia throughout the past few decades. Such decisions are made well in advance of the day of operations in an environment ignoring disruptions. However in practice, operations are rife with frictions caused by disturbances such as inclimate weather or mechanical failure. In spite of all the advances made at the planning level, there has been relatively little work done at the operational level.

Even though problems at the operational phase are similar to that of the planning phase, the former's problems are exacerbated by two things. The first are additional operational complexities that arise. For example, suppose an aircraft is approaching its destination but is unable to land due to convective weather. The aircraft may be placed into a holding pattern requiring additional flying time for the cockpit crew. By the time the aircraft lands, the crew may not be legal to fly their subsequent leg due to exceeding their allowed flying time within a 24-hour period rendering a disruption to the subsequent legs. The second problem is that of timing. Most airlines utilize an operations control center (OCC) which provide a centralized decision making environment. Unlike the planning phase in which problems are sometimes made over a year in advance of operations, OCC coordinators are constrained to making decisions in as close to real-time as possible. Because decisions involving repairing the schedule, aircraft, crew, and passengers are combinatorial in nature, using an optimization-based approach may not be tractable due to the complexity of solving

each of these operational problems. As a result, airlines do not generally rely on the use of mathematical programming in the presence of a disruption to their operations.

Given a disruption to the existing schedule, the airline is said to be in a *recovery opera-tion.* Developing an optimization model is naturally of interest to the Operations Research (OR) practitioner given the challenges posed. The immense nominal costs also make it of interest to an airline. While estimates vary, these are generally considered to be tens of billions of dollars annually in the U.S. alone (see [28]). Airline passengers also have a vested interest in the problem as passenger delays have become more problematic as the growth in air transportation has outpaced that of capacity at major airports. In some instances passenger delays have drawn global attention as passengers have been subjected to exces-sively long tarmac delays. These occurrences have, in part, prompted the U.S. Congress to draft a passengers' bill of rights. Effective April 2010 the U.S. Department of Trans-portation implemented a fine of up to $27,500 per passenger who exceed a tarmac delay of three hours. While there have been some advancements made in applying mathematical programming to the operational phase of airline scheduling, little advancements have been implemented in practice. One possible explanation is that the literature has considered only a proper subset of decisions required during a recovery period in order to deliver a solution in a timely manner. Such a solution scheme may not be of use to an OCC - for example, the recovered flight schedule may not be feasible for existing crew schedules.

The principle goal of this paper is to define, formulate, solve, and analyze a fully inte-grated recovery problem in a manner that is amenable to the constraints imposed by an OCC. By heuristically reducing the set of disruptable resources that are to be rescheduled, we propose an optimization module that is to reassign the schedule, aircraft, crews, and passengers within some time horizon. We validate our method by providing computational results using data from a real U.S.-based airline. To the best of our knowledge, we are the first to provide such results to the fully integrated problem. In the context of solving this problem we also introduce some results that can extend to other related problems within the industry.

The remainder of the chapter is organized as follows. Section 3.2 provides a review of

relevant work done within irregular airline operations. The problem and model are formally defined in Section 3.3. Section 3.4 discusses how the scope of the recovery operation is limited to make the problem solvable. Our decomposition scheme is outlined in Section 3.5. Computational results are shown in Section 3.6 that validates our approach. Here we observe the improvement the integrated approach yields relative to several key performance metrics.

3.2 Literature Review

While there has been relatively little work previously done for studying and solving the airline integrated recovery problem, various components within the problem have been studied. We review some of the seminar earlier work done. This is by no means a complete survey of irregular operations. Filar et al. [50] provides an exceptional survey of previous work. Clausen et al. [39] give a recent state-of-the-art overview of disruption management of schedule, aircraft, crew, passenger, and integrated recovery.

3.2.1 Schedule Recovery

Teodorovic and Guberinic [116] consider the problem of reassigning aircraft rotation when one or more aircraft are taken out of operation that minimizes total passenger delay. A flight network is formed and the schedule is repaired with the reduced set of aircraft. The solution is obtained by the branch-and-bound method for which an efficient two-step branching rule is implemented.

Using a lexicographic dynamic programming heuristic, Teodorovic and Stojkovic [117] introduce a model that seeks to minimize total flight cancellations while minimizing passenger delay. This is the first model that considers restoring the schedule and aircraft rotations in tandem.

The first work to integrate crew rotations with aircraft rotations was studied in Teodorovic and Stojkovic [118]. A heuristic model is introduced in which both aircraft and crew rotations are repaired through a first-in, first-out (FIFO) rule and a dynamic programming algorithm that incorporates re-timing decisions.

Jarrah et al. [69] introduce two network models that form the basis for irregular operations control at United Airlines. They allow the possibility of equipment swapping and allow the use of spare aircraft. The first model seeks to output a flight delay plan until the shortage of aircraft is resolved by minimizing total disutility. The second model achieves the same objective but considers flight cancellations instead of delays. Computational results are presented for each model showing considerable improvement relative to an unoptimimized schedule.

Yan and Yang [131] provide the first study that allows for delays and cancellations simultaneously. A network flow model with side constraints was introduced that are solved by Lagrangian relaxation with the subgradient method. By obtaining efficient bounds on the optimal objective, computations were tractable and their model was readily seen to deliver efficient solutions.

Yan and Tu [130] consider schedule re-optimization in the presence of multiple fleets. A multicommodity network flow model is introduce that is efficiently solved by a modified Lagrangian relaxation scheme using the subgradient method. A case study is presented in which their framework improved profits in each scenario. See Yan and Lin [129] for a similar study.

Clarke [38] introduces the Airline Schedule Recovery Problem (ASRP) that is strongly related to our model below. The comprehensive framework that is proposed considers flight delays and cancellations in tandem, as well the management of air traffic control (ATC). He also imposes constraints on crew availability so as to make the schedule compatible with respect to the initial positions of each crew. Two greedy heuristic procedures and an optimization-based solution procedure are considered and the results are evaluated under different scenarios.

Argüello et al. [10] use metaheuristic approach by presenting a greedy randomized adaptive search procedure (GRASP) to restore aircraft routings in the presence of a ground delay program. Their algorithm is polynomial in the number of flights and aircraft that has found near-optimal solutions to minimize delay and cancellation costs under a wide range of scenarios. Over 90% of the GRASP solutions were within 10% of optimality that were

generally obtained within a few seconds.

A binary quadratic programming approach is introduced by Cao and Kanafani [33] and [34] that integrates delays and cancellations. Their model maximizes profit while penalizing undesirable outcomes.

An overview of the decision-making environment at OCCs is given in Clarke [37]. This paper discusses the primary causes of irregularities, reviews the information systems and decision-support systems utilized, and proposes a new decision framework. A more recent, but similar exposition is given by Kohl et al. [74].

Three multicommodity network flow models are presented in Thengvall et al. [119] for schedule recovery that follows a hub closure. Each model considers flight cancellations, delays, ferrying, and swaps. The first two models - a pure network with side constraints and a generalized network - seek to maximize profit that attempts to keep as much of the original schedule preserved as possible. The third model, which is a pure network with side constraints with a discretized time horizon, seeks to minimize the cost incurred from flight cancellations and delays. Their results show that swapping opportunities have a substantial impact in the solution quality.

Stojković et al. [111] proposed a model that allows for not only the delaying of flights, but altering the duration of service as well to preserve maintenance schedules, ground service, crew connections, and passenger connections. The dual to their proposed model is a network model which allows for computation in near real-time.

Rosenberger et al. [97] develop a set packing model that seeks to assign routes to aircraft by minimizing an objective that is comprised of both the assignment cost as well as cancellation cost. Maintenance feasibility is preserved by enumerating all routings involving a maintenance activity a priori. Their model is considered in the presence of both aircraft disruptions as well as station disruptions in a ground delay program. They present an efficient heuristic that is used to identify the subset of aircraft that are to be rerouted, and their model is validated by simulation. They also extend their model to consider crew and passenger connections.

Eggenberg et al. [48] repair the schedule through an efficient column generation scheme

in which new columns are quickly generated through solving a resource constrained shortest path problem.

3.2.2 Crew Recovery

To our knowledge, the first to study crew recovery were Wei et al. [126]. The authors propose a comprehensive multicommodity network flow network. A heuristic-based search algorithm is used within the context of a depth-first search branch-and-bound algorithm that seeks to repair the original crew pairings. Song et al [108] consider a similar structure.

Stojković et al. [113] propose a model that, given a fixed flight schedule, seeks to output a set of modified crew pairings at minimum cost through a set partitioning problem that uses column generation throughout the branch-and-bound tree in a suitable runtime between a few seconds an about 20 minutes.

Our work is strongly related to Lettovsky et al. [78]. Given the set of canceled flights they also assign crew to modified pairings at minimum cost. They allow crews to deadhead either within the modified pairing or back to their crew base. They present efficient pre-processing techniques to identify the subset of the schedule to be disruptable. The model is solved by the primal-dual method on the LP relaxation of the model. Three branching techniques are studied, and they show that branching on follow-ons (where consecutive flight legs either are or are not present in a pairing) tends to be an efficient procedure for obtaining integer solutions.

Stojković and Soumis [112] consider a one day crew recovery model that allows for scheduling changes that keep aircraft routings fixed. Their problem is formulated as an integer nonlinear multicommodity network flow problem that is solved by Dantzig-Wolfe decomposition with branch-and-bound. Three problem instances are run showing that even in the largest instance, quality solutions were obtained in under 15 seconds.

Nissan and Haase [89] present a new methodology that is particularly appropriate to European carriers as their model assumes a fixed-cost structure of crew as oppose to pay-and-credit that is prevalent among North American carriers. Their objective is therefore to adhere as close as possible to the old schedule. By not explicitly repairing broken crew

pairings, the problem size is diminished considerably in that they solve a disruption for every duty period. A set-covering model is solved using branch-and-price with new columns being added from a residual network by solving a shortest path problem. Their approach is shown to solve in a runtime that is acceptable in operations.

3.2.3 Passenger Recovery

For the most part, airlines abstract passenger disruption within the context of their decision-making process. Finding an optimal tradeoff in the disruption of the schedule and its passengers, Bratu and Barnhart [30] suggest a framework that can reduce passenger disruptions while holding down other scheduling costs in irregular operations. Their model allows flight delays and cancellations that assigns reserve crew and spare aircraft to accommodate the new schedule. Two models are presented: the disrupted passenger metric (DPM) model and the passenger delay metric (PDM) model. The former model assigns only disrupted passengers and is only a proxy of actual delay costs, whereas the latter model assigns all passengers and provides a more accurate description of the true costs of delay. Their model is validated by a simulated OCC. While the DPM model is shown to not solve in sufficient time so as to implement in an actual OCC, the PDM model suggests that it might be amenable to a real-time decision making environment.

Zhang and Hansen [133] propose integrating other means of transportation to accommodate disrupted passengers. Such intermodal connections are often preferred particularly when the destination is relatively nearby the disrupted station within a hub-and-spoke network. By incorporating ground transportation into passenger recovery, they propose a mixed integer nonlinear programming model that is solved heuristically by first relaxing integrality and then fixing variables. Runtimes were shown to be under 20 minutes. Moreover their experiments show that the number of disrupted passengers may be greatly reduced by allowing intermodal substitution; one experiment showed this number was reduced by more than 84%.

3.2.4 Robust Scheduling

An area closely related to recovery is schedule robustness. The central idea is to design a schedule that is able to be recovered from more efficiently in the presence of irregularity. Robust scheduling was studied extensively in Ageeva[5], Smith [106], Rosenberger et al. [98], Smith and Johnson [107], and Burke et al. [32]. Crew robustness was studied in Klabjan et al. [73], Yen and Birge [132], Shebalov and Klabjan [105], Ball et. al. [17], Gao et al. [56], and Weide et al. [127]. The impact of schedule robustness to passenger recovery can be seen in Lan et al. [75].

3.2.5 Partially and Fully Integrated Recovery

There have been a number of studies whose aim is to partially integrate operations under irregularity. Abdelghany et al. [72] presented a decision support tool in which combines a schedule simulation with a resource optimization model that minimizes cancellations and disruptions while incorporating important crew considerations of both pilots and flight attendants. Given the anticipated severity of disruption the flight simulation model predicts a list of disrupted flights. Given this disruption the resource assignment optimization model assigns an efficient plan that is to delay and cancel flights that consider crew and aircraft swaps and utilization of reserve resources. A drawback of their approach is they do not allow crews to deadhead. After 177 potential flight disruptions are simulated, their iterative process saves 661 minutes of delay; 8.7% of the observed delay in the actual scenario which is found in just over 3 seconds.

The 2009 ROADEF challenge [91] introduced a competition that sought to deliver a recovery solution that was to integrate the schedule, aircraft, and passengers. Gabteni [54] presents an overview of the different proposed methodologies. The winning team, seen in Bisaillon et al. [27] employ a large-scale neighborhood search heuristic that iteratively constructs, repairs, and improves solutions that incorporates randomness to diversify the search procedure. Feasibility was quickly achieved in the first phase, while the third phase was shown to be significant as cost reductions we shown to be apparent in several instances.

The third place entry is shown in Acuna-Agost et al. [4]. They define a MIP model to

achieve their objective by first solving the problem on a very limited set with many variables fixed a priori. The novel feature of their framework is the introduction of a Statistical Analysis of Propagation of Incidents (SAPI). Using a logistic regression, the probability of each flight being disrupted are estimated. If these probabilities exceed a certain threshold flight cancellation variables are fixed, and if the probabilities are sufficiently low, the previous MIP solution is fixed. Neighboring solutions are then explored by local branching and fed back into the MIP. Because the search space is limited, the MIP computation is tangible.

3.2.6 Fully-Integrated Recovery

Handling aircraft and crew in concert is an arduous ask which explains why previous computational studies have ignored crew considerations. There have been some studies that include a fully integrated airline recovery framework, although these tend to be only formulations.

Two such proposals for integrated recovery are seen in Ph.D. dissertations by Lettovsky [77] and Gao [55]. The formulation given by the former is closely related to our work. He presented a fully integrated model that decomposes into a structure suitable for Benders decomposition. The linking variables are fleeting decisions to flight legs in which are passed to subproblems represented by repairing aircraft rotations, crew pairings, and passenger itineraries. While a formulation was provided, no computations were preformed.

3.3 The Airline Integrated Recovery Problem

We formally define the *airline recovery problem* to be comprised of the following four problems:

- The *schedule recovery problem* seeks to fly, delay, cancel, or divert flights from their original schedule. We call the solution to this problem the *repaired schedule*.

- The *aircraft recovery problem* assigns individual aircraft routings to accommodate the repaired schedule that are feasible for the constraints imposed by maintenance requirements.

- The *crew recovery problem* assigns individual crew members to flights according to the repaired schedule that satisfy the complex legality requirements.

- The *passenger recovery problem* re-assigns disrupted passengers to new itineraries that delivers them to their destination.

Given a disruption, we define the *time window* to be an exogenous interval $\mathcal{T} := \left[\underline{t}, \bar{t} \right]$ in which flights, aircraft rotations, crew schedules, and passenger itineraries are allowed to be modified. Each component may have a different interval, although we restrict our analysis to the same horizon. The requirement is that all components be back on their original (undisrupted) schedule by the end of the time window $\bar{t}$.

3.3.1 Schedule Recovery

The Schedule Recovery Model (SRM) returns re-timing and flight cancellation decisions. Our model is closely related to Clarke [38] in that we consider additional constraints imposed by air traffic control systems.

Instead of a leg-based model, we utilize flight strings which was introduced by [19]. A *flight string* (which we refer to as *string*) is a sequence of flights, with timing decision, to be operated by the same aircraft. The same sequence of flights might be present in multiple strings, although each sequence must have a unique set of re-timing decisions. A string-based model has a number of advantages. While the number of strings naturally grows significantly with respect to the number of flights, efficient column generations techniques can be employed. Strings are also able to capture network effects that individual flight decisions do not. Also, ground arcs need not formally be defined in the underlying time-space network. The biggest advantage is that integer solutions to the aircraft recovery problem (discussed in Section 3.3.2) are immediately obtained from the LP-relaxation.

3.3.1.1 Sets

F: set of all flight legs

E: set of equipment types (fleets)

S: set of flight strings

A: set of all airports

$A^{\mathbf{arr}}$: set of arrival slot capacities specified by an inbound station, arrival limit, and time interval

$A^{\mathbf{dep}}$: set of departure slot capacities specified by an outbound station, departure limit, and time interval

G: set of gate restrictions specified by a station, gate limit, and time interval

$I\left(a, \underline{t}^a, \bar{t}^a\right)$: set of strings that are inbound to station a between $\underline{t}^a$ and $\bar{t}^a$

$O\left(a, \underline{t}^a, \bar{t}^a\right)$: set of strings that are outbound from station a between $\underline{t}^a$ and $\bar{t}^a$

$W\left(a, \underline{t}^a, \bar{t}^a\right)$: set of strings that occupy a gate at station a between $\underline{t}^a$ and $\bar{t}^a$

$F^{\mathbf{strategic}}$: set of strategic flights that are prohibited from cancellation

$F^{\mathbf{market}}$: set of flights that have exogenous market requirements set by the airline that require a minimum number of flights or seats to be offered in a given segment

3.3.1.2 Data

$c_{e,s}^{\mathbf{assign}}$: cost of assigning equipment type $e \in E$ to string $s \in S$

$c_f^{\mathbf{cancel}}$: cost of cancelling flight $f \in F$

CAP_e: capacity of equipment type $e \in E$

$n_f^{\mathbf{seats}}$: minimum number of seats required by flight $f \in F^{\mathrm{market}}$

3.3.1.3 Decision Variables

$$
x_{e,s} = \begin{cases} 1 & \text{if equipment type } e \in E \text{ is assigned to string } s \in S \\ 0 & \text{otherwise} \end{cases}
$$

$$
\kappa_f = \begin{cases} 1 & \text{if flight } f \in F \text{ is cancelled} \\ 0 & \text{otherwise} \end{cases}
$$

The SRM formulation is given as follows:

$$\min \quad \sum_{e \in E} \sum_{s \in S} c_{e,s}^{\text{assign}} x_{e,s} + \sum_{f \in F} c_f^{\text{cancel}} \kappa_f \tag{4}$$

$$\text{s.t.} \quad \sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f}} x_{e,s} + \kappa_f = 1 \qquad \forall f \in F \tag{5}$$

$$\sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f}} x_{e,s} = 1 \qquad \forall f \in F^{\text{strategic}} \tag{6}$$

$$\sum_{e \in E} \sum_{s \in I\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \le n_a^{\text{arr}} \quad \forall \left(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a\right) \in A^{\text{arr}} \tag{7}$$

$$\sum_{e \in E} \sum_{s \in O\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \le n_a^{\text{dep}} \quad \forall \left(a, n^{\text{dep}}, \underline{t}^a, \bar{t}^a\right) \in A^{\text{dep}} \tag{8}$$

$$\sum_{e \in E} \sum_{s \in W\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \le n_a^{\text{gates}} \quad \forall \left(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a\right) \in G \tag{9}$$

$$\sum_{e \in E} \sum_{s: s \ni f} CAP_e x_{e,s} \ge n_f^{\text{seats}} \quad \forall f \in F^{\text{market}} \tag{10}$$

$$x_{e,s} \in \{0,1\} \quad \forall e \in E, \forall s \in S$$

$$\kappa_f \in \{0,1\} \quad \forall f \in F$$

The objective (4) is to minimize the aggregate cost comprised of string assignment (including re-timing decisions) and flight cancellations. Flight assignment constraints, as seen in (5), either require a flight to be contained in exactly one string or cancelled. To prohibit strategic flights from being cancelled constraints of the form (6) are added. Arrival and departure capacities at certain airports at given time intervals are not to be exceeded as captured in (7) and (8), respectively. (9) ensures the number of aircraft on the ground does not exceed the number of gates available at certain station and times. Market requirements are captured in (10); they ensure that a minimum number of seats are operated on certain flights. There are also other constraints that prohibit certain resources from being assigned to certain flights that we do not explicitly include for brevity. For instance, a curfew constraint ensures no flight arrives or departs within a curfew period. Other such constraints include weather restrictions, and constraints prohibiting certain fleet types from operating

at specific stations that cannot accommodate that type of aircraft.

3.3.2 Aircraft Recovery

The Aircraft Recovery Model (ARM) assigns individual tail numbers to strings while meeting maintenance and other aircraft requirements. The ARM is solved for each equipment type $e \in E$.

3.3.2.1 Sets

$AC(e)$**:** set of aircraft of equipment type $e \in E$

$A^{\mathbf{maint}}(e)$: set of maintenance stations capable of maintenance of equipment type $e \in E$

$H(e)$**:** set of aircraft of type $e \in E$ that requires maintenance activity within the time window $\mathcal{T}$

$S_n(a, t_{\min}, T)$**:** set of eligible strings to be flown by aircraft $n \in AC(e)$ that visit station $a \in A^{\mathrm{maint}}(e)$ for at least $t_{\min}$ units of time within subinterval $T \subset \mathcal{T}$

3.3.2.2 Data

$c_{e,s}^n$**:** cost of assigning tail $n \in AC(e)$ to string $s \in S$

3.3.2.3 Decision Variables

$$
x_{e,s}^n = \begin{cases} 1 & \text{if aircraft } n \in AC(e) \text{ is assigned to string } s \\ 0 & \text{otherwise} \end{cases}
$$

Given equipment type $e \in E$, the Aircraft Recovery Model, or ARM(e) is

$$\min \quad \sum_{n \in AC(e)} \sum_{s \in S} c_{e,s}^n x_{e,s}^n \tag{11}$$

$$\text{s.t.} \sum_{n \in AC(e)} x_{e,s}^n = x_{e,s} \quad \forall s \in S \tag{12}$$

$$\sum_{s \in S} x_{e,s}^n = 1 \quad \forall n \in AC(e) \tag{13}$$

$$\sum_{s \in S_n(a,t_{\min},T)} x_{e,s}^n \geq 1 \quad \forall (n,a,t_{\min},T) \in H(e) \tag{14}$$

$$x_{e,s}^n \in \{0,1\} \quad \forall s \in S, \forall n \in AC(e). \tag{15}$$

The objective (11) minimizes the cost associated with aircraft assignment. The cost can be thought of penalties or bonuses. For instance, a penalty may be imposed for any deviation from the original routing. The string cover constraints (12) assure each string that is chosen from the SRM is assigned to some eligible aircraft. (13) ensure each aircraft is assigned to precisely one string. In the event that the required initial and end stations coincide for a particular aircraft, we define a *null* string to be one with no flights so the aircraft stays on the ground. Maintenance cover constraints are seen in (14). This simply ensures that at least one maintenance opportunity is built in for all tail numbers requiring maintenance. The inputs to this class of constraints includes the eligible station(s), latest possible time for service, and minimum time duration necessary to perform the maintenance event. Different types of maintenance checks can be incorporated into these constraints with the given parameters required. The specific maintenance planning of choosing which event opportunities that are to be utilized can be done post-optimization. Other constraints we include but do not explicitly formulate are user-dependent constraints prohibiting certain aircraft from operating at some airports, and similar operational restrictions.

3.3.3 Crew Recovery

Crew members are assigned to *pairings* which are comprised of *duties* that contain specific flight assignments over a period of time. Each consecutive duty assignment must observe a

rigid set of legality rules as mandated by the FAA and possible additional airline and union requirements. A duty typically represents a single day of flying, and the pairing usually spans between 2 and 4 duties. A *roster* period consists of a number of pairings over a period of time, typically about one month. If a specific crew has a pairing that becomes disrupted, the pairing is said to be *broken*. A broken pairing may be augmented during the period overlapping with the time window $\mathcal{T}$ so as to deliver the crew to the station they are required to be at immediately outside of $\mathcal{T}$. All other components within the crew schedule outside of $\mathcal{T}$ are to be preserved. We ensure the repaired pairing is legal for the entire duration of the original pairing for the crew, although it may be not be the case for the roster period in which this would have to be fixed between the end of the pairing and end of the roster.

The Crew Recovery Model (CRM) seeks to repair disruptable crew pairings at minimum cost. Like the ARM, the CRM is solved for each equipment type corresponding to crew rating. For brevity within the context of CRM, a *pairing* is really meant by 'the broken component of the original crew pairing'.

Crew deadheading is an important component to the crew recovery process. Formally a *deadhead* occurs when a crew member is transported on a flight but does not operate the aircraft. Deadheading occurs during the recovery process when a schedule imbalance creates a shortage or surplus of crew members at a given station. There are two classes of deadheads. The first is deadheading within a pairing, i.e. when a crew member deadheads to some station to then operate a subsequent flight. The second class of deadheading is when crew members deadhead home to their crew base ending their current pairing. This is common when stringent legality requirements are nearly exhausted for a crew and no pairing can be assigned during the time window. Airlines typically have vastly different policies on deadheading crew members. Our module requires penalties for each class of deadheads that occur.

3.3.3.1   Sets

K: set of all available crew members

P_k: set of eligible pairings for crew $k \in K$

P: set of all pairings, i.e. $P = \bigcup_{k \in K} P_k$

A pairing $p \in P_k$ is *eligible for crew $k \in K$* if:

(i) p begins at the station where crew k is at the beginning $\underline{t}$ of the time window $\mathcal{T}$

(ii) p ends at the station where crew k is required to be at by the end $\bar{t}$ of the time window

(iii) all flight, duty, and pairing legality requirements are satisfied

3.3.3.2 Data

$c_{k,p}^{\text{assign}}$: cost of assigning crew $k \in K$ to pairing $p \in P_k$

d_f^{pairing}: cost of deadheading a crew on flight $f \in F$

d_k^{base}: cost of deadheading crew $k \in K$ back to base

3.3.3.3 Decision Variables

$$
y_{k,p} = \begin{cases} 1 & \text{if crew } k \in K \text{ is assigned to pairing } p \in P_k \\ 0 & \text{otherwise} \end{cases}
$$

$$
\nu_k = \begin{cases} 1 & \text{if crew } k \in K \text{ is to deadhead back to base} \\ 0 & \text{otherwise} \end{cases}
$$

s_f = the number of surplus crew on flight $f \in F$ (deadheads within pairing)

The CRM model we consider for equipment type $e \in E$, CRM(e), is as follows:

$$\min \quad \sum_{k \in K} \sum_{p \in P_k} c_{k,p}^{\text{assign}} y_{k,p} + \sum_{f \in F} d_f^{\text{pairing}} s_f + \sum_{k \in K} d_k^{\text{base}} \nu_k \tag{16}$$

$$\text{s.t.} \quad \sum_{\substack{k \in K \\ p \ni f}} \sum_{\substack{p \in P_k: \\ p \ni f}} y_{k,p} - s_f = 1 - \kappa_f \quad \forall f \in F \tag{17}$$

$$\sum_{p \in P_k} y_{k,p} + \nu_k = 1 \qquad \forall k \in K \tag{18}$$

$$y_{k,p} \in \{0,1\} \quad \forall k \in K, \forall p \in P_k$$

$$\nu_k \in \{0,1\} \quad \forall k \in K$$

$$s_f \in \mathbb{Z}_+ \qquad \forall f \in F$$

The objective (16) seeks to minimize total crew cost. (17) ensures that some crew operates each flight that is not cancelled. If $s_f > 0$, the flight is to contain at least one crew that is to deadheading on a pairing. (18) assigns each crew to either some eligible pairing or they are to deadhead to their home crew base.

3.3.4   Passenger Recovery

There are two components to the passenger recovery process. The first is an iterative module by which the costs from aggregate itinerary delays are minimized by integration with the SRM, ARM, and CRM. The second problem takes the eligible set of itineraries from the first problem and assigns itineraries to passenger groups to minimize the actual cost associated with passenger delay.

Each passenger is defined as a 4-tuple consisting of origin, departure time at origin, destination, and scheduled time of arrival at destination. All possible eligible itineraries are generated a priori from the original flight schedule. Some itineraries are constructed even though they may be infeasible from the initial schedule, but may become feasible with delays. For example, consider a passenger scheduled to depart at 8:00. If there is a flight between the same origin and destination scheduled to depart at 7:00, then that flight might be able to be used in the recovery solution if it experiences a delay of at least

one hour. If it does not, then constraints will prohibit the use of that itinerary. Our model reassigns disrupted passengers to new itineraries assuming homogeneous passengers. In practice a more granular version of this is employed that distinguishes each individual passenger based on certain attributes like fare class or frequent flier status. Our framework chooses the specific itineraries that are to be used determining the flow of passengers to be assigned to each itinerary only, and not which specific passengers are to be assigned (this could be done post-processing).

3.3.4.1 Sets

OD: set of disrupted passengers classified by an origin-destination (OD) pair

Γ: set of all passenger itineraries

$\Gamma_i \subseteq \Gamma$: set of all itineraries eligible to assign passenger $i \in OD$

$\Gamma_i^{\textbf{multi-flt}} \subseteq \Gamma_i$: set of multi-flight itineraries available to passenger $i \in OD$

3.3.4.2 Decision Variables

$z_{i,\gamma}$: number of passengers from $i \in OD$ to assign to itinerary $\gamma \in \Gamma_i$

s_i: number of passengers from $i \in OD$ that are not assigned to an itinerary

$\delta_{i,\gamma}$: hourly delay if passenger $i \in OD$ is assigned to itinerary $\gamma \in \Gamma_i$

3.3.4.3 Data

$c_{i,\gamma}^{\textbf{delay}}$: hourly cost of passenger delay associated with assigning $i \in OD$ to itinerary $\gamma \in \Gamma_i$

$c_i^{\textbf{unassign}}$: cost of being unable to a assign passenger to an itinerary

$\omega_{i,\gamma}$: weight of assigning $i \in OD$ to itinerary $\gamma \in \Gamma_i$ in the aggregate delay cost

$n_i^{\textbf{PAX}}$: number of passengers for $i \in OD$

CAP_e: capacity of equipment type $e \in E$

$\underline{f}(\gamma)$: initial flight in itinerary $\gamma \in \Gamma$

$\overline{f}(\gamma)$: final flight in itinerary $\gamma \in \Gamma$

$t_f^{\mathbf{arr}}$: actual time of arrival for flight $f \in F$

$t_f^{\mathbf{dep}}$: actual time of departure for flight $f \in F$

$t_i^{\mathbf{STD}}$: scheduled time of departure at origin for $i \in OD$

$t_i^{\mathbf{STA}}$: scheduled time of arrival at destination for $i \in OD$

$t_{\min}^{\mathbf{connect}}$: minimum passenger connection time for multi-flight itineraries

3.3.4.4 Itinerary Recovery Model

As previously discussed, all eligible itineraries are initially constructed given the original flight schedule. Several of the itineraries will be ineligible with different solutions provided by the SRM. The itinerary recovery model (IRM) seeks to output the set of eligible itineraries for each OD such that the *aggregate delay costs* are minimized subject to ensuring a feasible set of passenger itinerary assignments. The IRM is formulated as follows.

$$\min \quad \sum_{i \in OD} \sum_{\gamma \in \Gamma_i} c_{i,\gamma}^{\text{delay}} \omega_{i,\gamma} \delta_{i,\gamma} + \sum_{i \in OD} c_i^{\text{unassign}} s_i \tag{19}$$

$$\text{s.t.} \sum_{i \in OD} \sum_{\substack{\gamma \in \Gamma_i: \\ \gamma \ni f}} z_{i,\gamma} \leq \sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f}} x_{e,s} CAP_e \quad \forall f \in F \tag{20}$$

$$\sum_{\gamma \in \Gamma_i} z_{i,\gamma} + s_i = n_i^{OD} \qquad \forall i \in OD \tag{21}$$

$$\delta_{i,\gamma} \geq \sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni \overline{f}(\gamma)}} x_{e,s} t_{\overline{f}(\gamma)}^{\text{arr}} - t_i^{\text{STA}} \quad \forall i \in OD, \forall \gamma \in \Gamma_i \tag{22}$$

$$z_{i,\gamma} \leq n_i (1 - \kappa_f) \quad \forall f \in F, \forall (i,\gamma) \in OD \times \Gamma_i : \gamma \ni f \tag{23}$$

$$\sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni \underline{f}(\gamma)}} t_{\underline{f}(\gamma)}^{\text{dep}} x_{e,s} \geq t_i^{\text{STD}} - M_{i,\gamma} (1 - v_{i,\gamma}) \quad \forall i \in OD, \forall \gamma \in \Gamma_i \tag{24}$$

$$z_{i,\gamma} \leq M_{i,\gamma} v_{i,\gamma} \quad \forall i \in OD, \forall \gamma \in \Gamma_i \tag{25}$$

$$\sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f_j}} t_{f_j}^{\text{dep}} x_{e,s} - \sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f_i}} t_{f_i}^{\text{arr}} x_{e,s} \geq t_{\min}^{\text{connect}} - M'_{i,\gamma} (1 - w_{i,\gamma}) \quad \forall i \in OD, \forall (f_i, f_j) \in \Gamma_i^{\text{multi-flt}}$$
$$\tag{26}$$

$$z_{i,\gamma} \leq M'_{i,\gamma} w_{i,\gamma} \qquad \forall i \in OD, \forall \gamma \in \Gamma_i^{\text{multi-flt}} \tag{27}$$

$$(z_{i,\gamma}, \delta_{i,\gamma}, v_{i,\gamma}, w_{i,\gamma}) \in \mathbb{Z} \times \mathbb{R} \times \{0,1\} \times \{0,1\} \quad \forall i \in OD, \forall \gamma \in \Gamma_i$$

$$s_i \in \mathbb{Z} \qquad \forall i \in OD$$

The objective (19) seeks to minimize the total weighted nominal delay cost of all itineraries and unassigned passengers. The weights can be either unit-valued or reflect the share of OD passengers present in the disruption. (20) prohibits the spilling of passengers. For each OD (21) either assigns passengers to a feasible itinerary or strands them with no itinerary being assigned. If a passenger cannot be assigned to an itinerary, they may overnight at a connection point, be placed on another airline, or have their itinerary delayed outside of $\mathcal{T}$. (22) tracks the delay of each passenger-itinerary pair where the itinerary delay is the difference between the actual arrival time of the last flight in the itinerary and the scheduled time of arrival to the passenger's destination. (23) ensures no passenger is assigned to an itinerary that contains a cancelled flight (where n_i is an upper bound for $z_{i,\gamma}$).

Recall that all eligible itineraries are overbuilt a priori in which some itineraries are infeasible with respect to the original schedule but may become eligible through delays. The next four constraints are logical constraints that ensure only legal itineraries are considered given the solution from the SRM. Inequalities (24) and (25) prohibit assigning any itineraries to passengers in which the initial flight in the itinerary departs prior to the passenger ready time. For all $i \in OD$ and $\gamma \in \Gamma_i$, $M_{i,\gamma} = t_i^{\text{STD}}$ is chosen as a valid upper bound. Given the solution from the SRM, passenger connection times are observed. If the connection time does not exceed the minimum necessary connection time $t_{\min}^{\text{connect}}$, then no passengers can be assigned to that itinerary. This is reflected in (26) and (27) where $M'_{i,\gamma} > 0$ is appropriately chosen (for example, maximum possible connection time).

3.3.4.5 Passenger Reaccommodation Model

Once the set of flight strings have been found that induces the minimal aggregate passenger delay, the passenger reaccommodation model (PRM) is solved. The PRM allocates passengers to the given set of itineraries to minimize the total *assignment cost*.

For all $i \in OD$ let Γ_i^* denote the set of eligible itineraries for the given OD induced by the optimal SRM solution. The PRM is formulated as

$$\min \quad \sum_{i \in OD} \sum_{\gamma \in \Gamma_i^*} c_{i,\gamma}^{\text{delay}} \delta_{i,\gamma}^* z_{i,\gamma} + \sum_{i \in OD} c_i^{\text{unassign}} s_i \tag{28}$$

$$\text{s.t.} \quad \sum_{\substack{i \in OD}} \sum_{\substack{\gamma \in \Gamma_i^*: \\ \gamma \ni f}} z_{i,\gamma} \leq \sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni f}} x_{e,s} CAP_e \quad \forall f \in F \tag{29}$$

$$\sum_{\gamma \in \Gamma_i^*} z_{i,\gamma} = n_i^{OD} \quad \forall i \in OD \tag{30}$$

$$z_{i,\gamma} \in \mathbb{Z} \quad \forall i \in OD, \forall \gamma \in \Gamma_i^*$$

$$s_i \in \mathbb{Z} \quad \forall i \in OD.$$

Note the summations in (29) and (30) differ from (20) and (21) in that the former are taken over the index sets Γ_i^*. While the objective function of the IRM does not depend on $z_{i,\gamma}$, constraints (20) and (21) are included in the IRM to ensure a feasible solution in the

PRM. Moreover the cost coefficients $c_{i,\gamma}^{\text{delay}}$ are chosen to be identical for both the IRM and PRM to measure the cost associated with passenger delay.

The two-stage approach to passenger recovery can be combined into a single step in which reaccommodation is done explicitly. However our approach is advantageous in two ways. Considerable computational effort is required to model each passenger individually; the number of cut coefficients generated by the Benders cut has introduced a vast complexity to the master problem which is solved as a Mixed-Integer Programming (MIP) problem. Secondly, our approach only requires a single call to the itinerary generator a priori as opposed to building new itineraries every time a the master problem is solved.

3.4   Limiting the Scope of Recovery

The size and complexity of the integrated recovery problem outlined above most likely precludes the delivery of a globally optimal solution. In order to tractably solve the problem for reasonably large scenarios, careful consideration must be placed on how to limit the size or scope of the problem.

A flight is said to be *disrupted* if one of its resources precludes the flight from operating as scheduled. Such resources include the arrival or departure airport, aircraft, or assigned crew members. Flight disruptions may be exogenous or endogenous. An example of an exogenous disruption is the closure of an airport for a specific period of time, in which all flight activity to or from the airport within that time interval must be altered. However, system-wide disruptions can be mitigated by *endogenous* flight disruptions. An example of an endogenous flight disruption is seen in Figure 15 on a simple flight network consisting of three flights: 101 from MIA to ATL, 102 from ATL to ORD, and 114 from CLT to ATL. The thick black segment at ATL represents a closure which forces the (exogenous) disruption to flight 101. While flight 102 is unaffected by the disruption, it may be advantageous to (endogenously) delay the flight in order to accommodate connecting passengers. Of course this illustration is simplistic, but shows the combinatorial nature of the problem.

Flights that are candidates for disruptions are said to be *disruptable*. For example, consider flight 114 from Figure 15 that is directly unaffected by the disruption. It would be
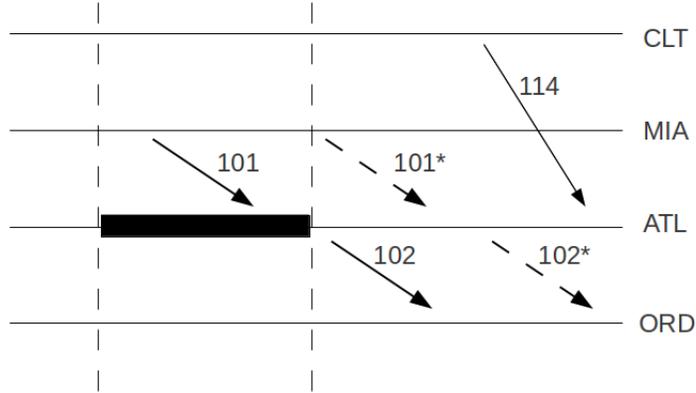
Figure 15: Exogenous vs. Endogenous Flight Disruptions

plausible to not consider that flight as a candidate for disruption. While simple to identify on a three-flight example, the process of identifying which subset of flights to be considered disruptable poses a considerable challenge.

We now discuss the procedure by which we identify all disruptable flights. Initially the disruptable flight set includes those flights that are directly affected by a resource at the airport. The set is then expanded to consider aircraft, crews, and passengers.

3.4.1 Limiting Flights

The disruptable flight set is instantiated with all exogenous flight disruptions that contain a resource that forces a delay or cancellation.

Flights from disrupted routings   A *disruptable aircraft* exists if its scheduled routing contains a disruptable flight. Suppose k_n flights are scheduled for disruptable aircraft n within the time window $\mathcal{T}$ denoted by $f_1, f_2, \ldots, f_{k_n}$. Let f_i denote the earliest flight from the disruptable routing present in the disruptable flight set. Denote $F_n \equiv \{f_i, f_{i+1}, \ldots, f_{k_n}\}$ as all subsequent flights within $\mathcal{T}$ that were scheduled to be operated by aircraft n. Because of delay propagation, a disruption to flight f_i may cause disruption to the subsequent flights from F_n. Thus the disruptable flight set is appended with *all* flights from F_n. Repeating this procedure for all disruptable routings gives the updated disruptable flight set.

Flights from disrupted crew   Similar to that of aircraft, a *disruptable crew* exists if a crew is scheduled to fly a disruptable flight within their pairing. The disruptable flight set is appended in a similar fashion to that of aircraft. A list of flights is extracted that each crew member is scheduled to fly in the disruption period. If a disruptable flight is present, then that flight and all subsequent flights within the scheduled pairing within the disruption period are added to the flight set.

The new flights that have been added from the crew schedules might be operated by aircraft not previously identified as disruptable. In this case, the new aircraft is appended to the disruptable set of aircraft.

Flights from tight passenger connections   We take a passenger-centric approach to integrated recovery, and thus minimizing passenger delay is central to our study. We further modify the disruptable flight set by considering additional candidate flights that are identified for abating passenger delay through preprocessing. Consider a passenger originating in MIA whose destination is ORD seen in Figure 16. Note that the connection between flights 101 and 102 appears to be tight. Even a moderate disruption in flight 101 is likely to break the connection for such passengers. Additional flight candidates are introduced for such tight connections through a simple rule. If a non-disruptable flight has the same origin and destination from a flight contained in a tight connecting itinerary, then that flight is introduced as disruptable if the departure times are within some tolerance threshold specified by the airline. Figure 16 illustrates this concept of augmenting the disruptable flight set to mitigate passenger delay. There are two other non-disruptable flights from ATL to ORD. Flight 100 departs from ATL relatively near that of flight 102 and is added to the disruptable flight set assuming the difference is within the threshold. Naturally all passengers on flight 100 are then considered in our model since the flight becomes disruptable. If the departure of flight 110 is too late (i.e. outside the threshold), it remains non-disruptable.

These new flights will have new aircraft and new crew members associated with them. As was done with adding new flights from crew schedules, we consider the single-flight entities only, and ensure both the aircraft and crew members are eligible to operate the

58
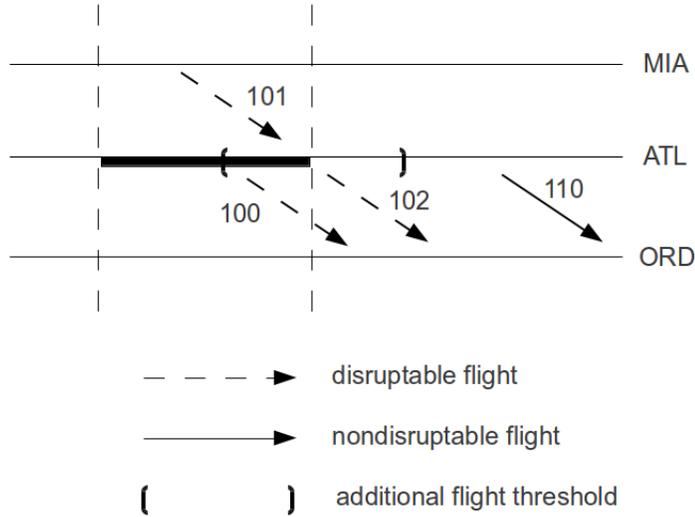
Figure 16: Identifying Passenger-Friendly Flights

next flight in their respective schedules.

3.4.2 Re-timing Flights

Initial work on airline recovery modeled flight delays by making copies of each flight arc that departed at uniform intervals (see Clarke [38] and Gao [55]). While the uniform flight copy approach is simple and intuitive, generating strings over copies of flights becomes extraordinarily large and complex. We instead model delays through an event-driven approach. The idea is that events like arrivals, and times associated with constraints from the SRM give more relevant delay decisions than arbitrary departure times from uniform flight copies.

Given a maximum allowable delay period $d_{\max}$, a timeline is created for each flight from 0 to $d_{\max}$ representing the given flight delay. Note that in the SRM some constraints are a function of time (see, for example, constraints (7) through (9)). Formally these are referred to as *time-dependent constraints*. Table 3 gives an example of a set of time-dependent constraints present in the flight network from Figure 16.

The flight departure interval is partitioned into $k \geq 1$ disjoint subintervals from the set of time-dependent constraints that give a maximum of $k + 1$ departure options. If a flight

Table 3: An Example of Time-Dependent Constraints

Event	Time	Station	Constraint	Directly Affected by Disruption?
1	0930 - 1030	ATL	flow rate reduction	Yes
2	0930 - 1000	MIA	gate restriction	No
3	1130 - 1200	ATL	slot restriction	No
4	1200 - 1245	MIA	gate restriction	No

f is present in any of the time-dependent constraints, then a new subinterval is created representing a new candidate departure time. Each string must then have no more than one departure from each subinterval. Strings are generated through the *augmented flight network*, defined to be the original flight network whose number of copies (i.e. delay options) correspond to the number of subintervals from the delay interval.

Figure 17 shows a simple two-flight example of how delay options are generated from these events using a maximum allowable delay (d_{max}) of 2 hours. The shaded regions in Figure 17a represent the time-dependent constraints as given in Table 3. Figure 17b shows how the flight network is augmented to accommodate different departure times. Both flights are partitioned into 3 subintervals giving a maximum of 4 departure options for each flight.

The idea of event-driven delays is that the strings present in the augmented flight network are likely to dominate most strings created from uniform flight copies whilst generating fewer flight strings. From Figure 17 there are a maximum of 4^2 possible strings from this approach. If uniform flight copies were instead employed at a coarse discretization of 15 minutes, 8 delay options would arise in addition to the original flight departure time. Thus, 9 copies of the same flight are represented for two flights giving a maximum of 9^2 possible strings for just this trivial two-flight illustration. Another problem with uniform flight copies is that several strings are likely to be present in the same set of time-dependent constraints and therefore exhibit duplicate columns in the SRM formulation.

3.5 Solution Methodology

Even by limiting the scope of the problem to make it computationally tractable, the problem is likely too large and complex to return a globally optimal solution for most reasonable
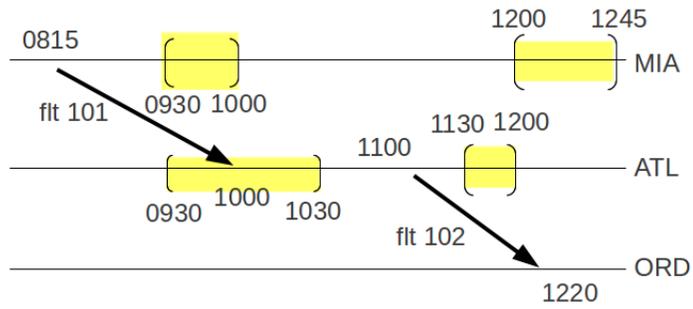
(a) Scheduled Flights and Time-Dependent Constraints
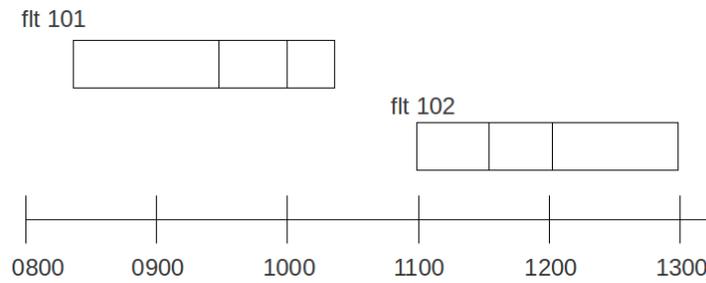
(b) Augmenting the Flight Network

Figure 17: Modeling Event-Driven Flight Delays

disruption scenarios. There is an inherent tradeoff between solution quality and runtime. A possible method might be to develop a recovery scheme in a two-phased approach that first seeks to recover the schedule, then to recover the other three components taking the repaired schedule as given. There are a number of problems associated with this scheme, however tractable as it seems. Conflicting objectives almost certainly exist between the schedule, crew costs, and passenger delays. Passing a single feasible schedule is too restrictive with respect to each of the second-stage problems. We argue that if this were a plausible recovery method in practice, virtually every airline OCC would have already implemented a variation of such a solution strategy. Instead, airlines often try to find a single feasible schedule manually. The other extreme would be to deliver a fully integrated solution that is globally optimal with respect to each of the four components. And while an integrated recovery framework is naturally desirable, the size and complexity may preclude such a mechanism to be implemented in practice. Therefore a balance between these two extremes must be reached with the goal of delivering an integrated solution.

Our approach is to return a solution that is globally optimal with respect to aggregate passenger delay meaning passenger assignment are globally optimal over all itineraries *and* all flight strings. We emphasize that optimality is in accordance to our model over the reduced problem whose scope has been limited as discussed in the preceding section. While this is clearly desirable for crew scheduling decisions as well, the crew recovery component is the bottleneck of the process and the number of repaired pairings can be so large that optimizing over all pairings *and* strings is unlikely to solve in an efficient manner. Two tactics are employed to ameliorate the large cost associated with crew recovery:

1. We do not require the delivered solution to be globally optimal over all strings and pairings. New pairings are priced out until the master solution $\left(x_{e,s}^*, \kappa_f^* \right)$ is globally optimal for the IRM and feasible for the ARM and CRM. When this termination criterion is reached no further pairings are priced out (see Figure 18). Thus our approach is considered to be passenger friendly with crew considerations.

2. Multiple cockpit crew members are required for each flight, usually two including a

captain and a first officer. Even though the crew members may have different pairings, we assume the pair of crew members assigned at the beginning of the time window stay fixed through the time window. We solve the CRM only for the captain and check the legality of the first officer in the post-processing stage. If the assigned pairing violates some legality restriction, a swap is conducted or a reserve crew is assigned if possible.

Other than being computationally tractable for a single-day horizon, returning a globally optimal passenger solution has another advantage: it is more satisfying to passengers whose aggregate delay is at a minimum. Recent news headlines have reported about excessive passenger delays inducing a 'passenger revolt' and a number of variants for a passenger bill of rights have been proposed among Congress. Effective April 2010 the U.S. Department of Transportation has enacted a rule whereby airlines would be forced to pay up to $27,500 for *each* passenger experiencing a tarmac delay in excess of three hours (U.S. Department of Transportation 49 U.S.C. 40113).

3.5.1 Decomposition

Because scheduling decisions affect repaired aircraft rotations, crew schedules, and passenger itineraries, employing a Benders' decomposition scheme would be natural to decompose the problem. The master problem is the SRM with linking variables $\{x_{e,s}\}$, $\{\kappa_f\}$ that are passed into the subsequent subproblems: ARM, CRM, and IRM.

While the three subproblems are independent of each other, they are solved sequentially. First, the SRM and IRM iterate until the aggregate passenger delay cost is minimal. The ARM is then solved. If the ARM is infeasible, a Benders feasibility cut is added to the SRM. Otherwise, the CRM is then solved. Again, a feasibility cut is added if the CRM is infeasible. Otherwise, a tentative solution is found. If the optimality gap between the current CRM iterate is within some tolerance level specified by the user, a solution to the iterative scheme is given. Otherwise, new columns are generated and returned to the CRM, or a Benders optimality cut is returned to the SRM. The problem structure is amenable to parallelization, but we employ the sequential implementation.

There are five classes of Benders cuts that are passed into the master problem. Only

the relaxation of each of the three subproblems are solved so as to obtain coefficients of the Benders cuts. The master problem is first solved as an LP-Relaxation, and new strings are generated based on the corresponding dual extreme ray if the relaxed SRM is infeasible until feasibility is attained. Obtaining integer solutions for the three subproblems is further discussed in Section 3.5.4.

The five families of Benders cuts that are included in the master problem are

$$\sum_{e \in E} \sum_{s \in S} \pi^{\text{ARM}}_{e,s} x_{e,s} \leq \pi^{\text{ARM}}_0 \tag{31}$$

$$\sum_{f \in F} (1 - \kappa_f) \pi^{\text{CRM}}_f + \sum_{k \in K} \rho^{\text{CRM}}_k \leq 0 \tag{32}$$

$$\sum_{f \in F} (1 - \kappa_f) \pi^{\text{CRM}}_f + \sum_{k \in K} \rho^{\text{CRM}}_k \leq \eta^{\text{CRM}} \tag{33}$$

$$\sum_{e \in E} \sum_{f \in F} \sum_{s \ni f} \pi^{\text{IRM}}_f CAP_e x_{e,s} + \sum_{i \in PAX} \sum_{\gamma \in \Gamma_i} \sum_{e \in E} \sum_{s \ni \overline{f}(\gamma)} \sigma^{\text{IRM}}_{i,\gamma} t^{\text{arr}}_{\overline{f}(\gamma)} x_{e,s}$$

$$+ \sum_{i \in OD} \sum_{\gamma \in \Gamma_i} \sum_{f \in \gamma} \nu^{\text{IRM}}_f n_i (1 - \kappa_f) + \sum_{i \in OD} \sum_{\gamma \in \Gamma_i} \theta^{\text{IRM}}_{i,\gamma} \left(\sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni \underline{f}(\gamma)}} t^{\text{dep}}_{\underline{f}(\gamma)} x_{e,s} \right)$$

$$+ \sum_{i \in PAX} \sum_{\gamma \in \Gamma^{\text{multi-flt}}_i} \tau^{\text{IRM}}_{i,\gamma} \left(\sum_{e \in E} \sum_{s \ni f_j} t^{\text{dep}}_{f_j} x_{e,s} - \sum_{e \in E} \sum_{s \ni f_i} t^{\text{arr}}_{f_i} x_{e,s} \right) \leq \pi^{\text{IRM}}_0 \tag{34}$$

$$\sum_{e \in E} \sum_{f \in F} \sum_{s \ni f} \pi^{\text{IRM}}_f CAP_e x_{e,s} + \sum_{i \in PAX} \sum_{\gamma \in \Gamma_i} \sum_{e \in E} \sum_{s \ni \overline{f}(\gamma)} \sigma^{\text{IRM}}_{i,\gamma} t^{\text{arr}}_{\overline{f}(\gamma)} x_{e,s}$$

$$+ \sum_{i \in OD} \sum_{\gamma \in \Gamma_i} \sum_{f \in \gamma} \nu^{\text{IRM}}_f n_i (1 - \kappa_f) + \sum_{i \in OD} \sum_{\gamma \in \Gamma_i} \theta^{\text{IRM}}_{i,\gamma} \left(\sum_{e \in E} \sum_{\substack{s \in S: \\ s \ni \underline{f}(\gamma)}} t^{\text{dep}}_{\underline{f}(\gamma)} x_{e,s} \right)$$

$$+ \sum_{i \in PAX} \sum_{\gamma \in \Gamma^{\text{multi-flt}}_i} \tau^{\text{IRM}}_{i,\gamma} \left(\sum_{e \in E} \sum_{s \ni f_j} t^{\text{dep}}_{f_j} x_{e,s} - \sum_{e \in E} \sum_{s \ni f_i} t^{\text{arr}}_{f_i} x_{e,s} \right) \leq \eta^{\text{IRM}} + \pi^{\text{IRM}}_0 \tag{35}$$

where the superscripts denote the given subproblem, π^{ARM}_0 and π^{IRM}_0 are constants that depends on the dual variables from the right-hand side of constraints that do not depend on master variables from the ARM and IRM, respectively. η^{CRM} and η^{IRM} are new decision variables in the master problem corresponding to the optimal objectives in the CRM and IRM, respectively. The cuts are ARM feasibility, CRM feasibility, CRM optimality, IRM

feasibility, and IRM optimality, respectively. We model the ARM as a feasibility problem so ARM optimality cuts are unnecessary.

3.5.2 Column Generation

Given the large number of flight strings and repaired crew pairings, only a subset of columns are generated through each of these problems. Multiple columns are generated through a residual network which is built from the flight network for flight strings and the crew duty network for repaired crew pairings. Given a directed network $G = (V, A)$, a dummy source and sink node are added in which a variable (flight string or repaired crew pairing) corresponds to an $s - t$ path. Paths are constructed by computing the reduced cost for every arc $a \in A$. Arcs with a sufficiently high reduced cost are eliminated and resulting paths (columns) are generated. In order to generate multiple columns at once, a tolerance parameter $\epsilon > 0$ is defined and all columns whose path p prices out less than ϵ are then added. This is sometimes known as path generation through an ϵ-residual network (see Ahuja et al. [7] for a general description; Shaw [104] gives an example pertinent to a traditional crew pairing problem). A summary of this method is shown in Algorithm 1.

Algorithm 1 Path Generation Through ϵ-Residual Network

    Given: Set of resources R, general resource network $G = (V, A)$, dual information $\pi_v \quad \forall v \in V$, and tolerance parameter $\epsilon > 0$
    Initialize: Newly generated variables $X = \emptyset$
    for $i = 1$ **to** $|R|$ **do**
        create augmented network for resource i, $G^i = (V, A)$
        add source node s and sink node t
        construct all arcs from s to eligible initial nodes and arcs to t from eligible end nodes
        for all $a \in A$ **do**
            compute reduced cost $\bar{c}_a$
            if $\bar{c}_a > \epsilon$ **then**
                delete arc a: $A \leftarrow A \setminus \{a\}$
            end if
        end for
        Let $X^i = \left\{ \bigcup p : \ p \text{ is an } s - t \text{ path s.t. } \sum_{a \in p} \bar{c}_a < \epsilon \right\}$
        $X \leftarrow X^i$
    end for
    return new columns X

3.5.3 Simultaneous Row and Column Generation

The preceding section illustrates how we are employing both Benders cuts as well as column generation. While these two classical large-scale optimization methods are widely known, they are in isolation of one another. Given an infeasible or suboptimal subproblem a Benders cut $f(x_{e,s}, \kappa_f) \leq f_0$ is added to the master problem. But this cut generated is valid only over the subset of strings $S' \subseteq S$ that have been generated. Moreover in the case of the CRM where repaired crew pairings are also being generated, the given cut is valid only over those subset of pairings $P' \subseteq P$ that have been generated.

We discuss two cases how these methods are used together.

3.5.3.1 Flight Strings

A general Benders cut is valid over all *generated* flight strings $S' \subseteq S$. As new strings are added, the Benders cut may be invalid for some $s \in S \setminus S'$. While to the best of our knowledge, there does not exist a way to overcome this barrier , we simply remove the Benders cuts anytime new strings are added (a related problem introduced by Van Roy [123] is that of cross decomposition). Because cycling may occur once the cuts are deleted, we do not generate new strings within every iteration. Rather, they are generated every $k > 1$ iterations from the LP-Relaxation of the master problem.

3.5.3.2 Repaired Crew Pairings

A Benders cut is valid over all generated linking variables as well as those local to the subproblem. However, if columns are being added to the subproblem, new columns may violate the previous cuts rendering them as invalid to all variables. Therefore any cut initially generated becomes a *candidate cut* since it is feasible only over all *generated* variables. In the context of the CRM, we denote $P' \subseteq P$ to be the set of all generated pairings. Simultaneity of these two procedures by first obtaining a certificate of infeasibility that proves the CRM is infeasible over all P for a given master solution. If the candidate cut meets this criterion, then the cut is added to the master. Otherwise, it is discarded. In both cases, new columns are being generated.

The complete details of how Benders' cuts and newly-generated crew pairings are discussed in Chapter 4.

3.5.4 Integrality

The iterative Benders scheme solves only the master problem (SRM) to integrality and solves the subsequent three subproblems in their respective LP relaxations. Once the iterative algorithm has terminated, then branching is done to find a nearby solution if a fractional solution is present. If no feasible integer solution is found by branching, the node returned by the algorithm is then rejected and the procedure is to continue until an integer solution is delivered. We discuss how integrality is obtained in each of the three subproblems.

SRM Integrality The SRM module is solved to integrality using branch-and-cut. One particularly useful strategy is to branch on *follow-ons*. This concept was introduced by Ryan and Falkner ([99]). A *follow-on* is a pair of flights that are contained in the same fractional-valued string. The branching dichotomy either forces or forbids the given follow-on. Anbil et al. ([9]) and Lettovsky et al. ([78]) show follow-on branching to be successful in driving integrality of crew recovery models in particular. We find this branching strategy to also be very effective in the SRM.

ARM Integrality One of the advantages of the flight string models is it makes the routing problem considerably easier to solve as shown in Theorem 3.5.1.

Theorem 3.5.1. (ARM Integrality) *The polyhedron associated with the LP-Relaxation of the ARM is integral*

Proof. This problem reduces to a maximum cardinality bipartite matching problem for node sets aircraft-string assignments $\{x_{e,s}^n\}$ and assigned strings from the master problem $\{x_{e,s}^*\}$. This class of problems is well-known to be integral (see Nemhauser and Wolsey [88]). $\square$

CRM Integrality Solving the LP-relaxation of the CRM induces integer solutions in many scenarios. However the polytope is itself not integral. Similar to the case of driving SRM integrality, we employ branching on follow-ons with respect to fractional crew pairings.

IRM Integrality Solving the PRM could be done through a multi-commodity network flow algorithm yielding integer solutions. However the associated polyhedra is highly integral and branching is done only in the presence of a fractional solution.

3.5.5 Overview

Figure 18 summarizes our approach to solving the AIR model.
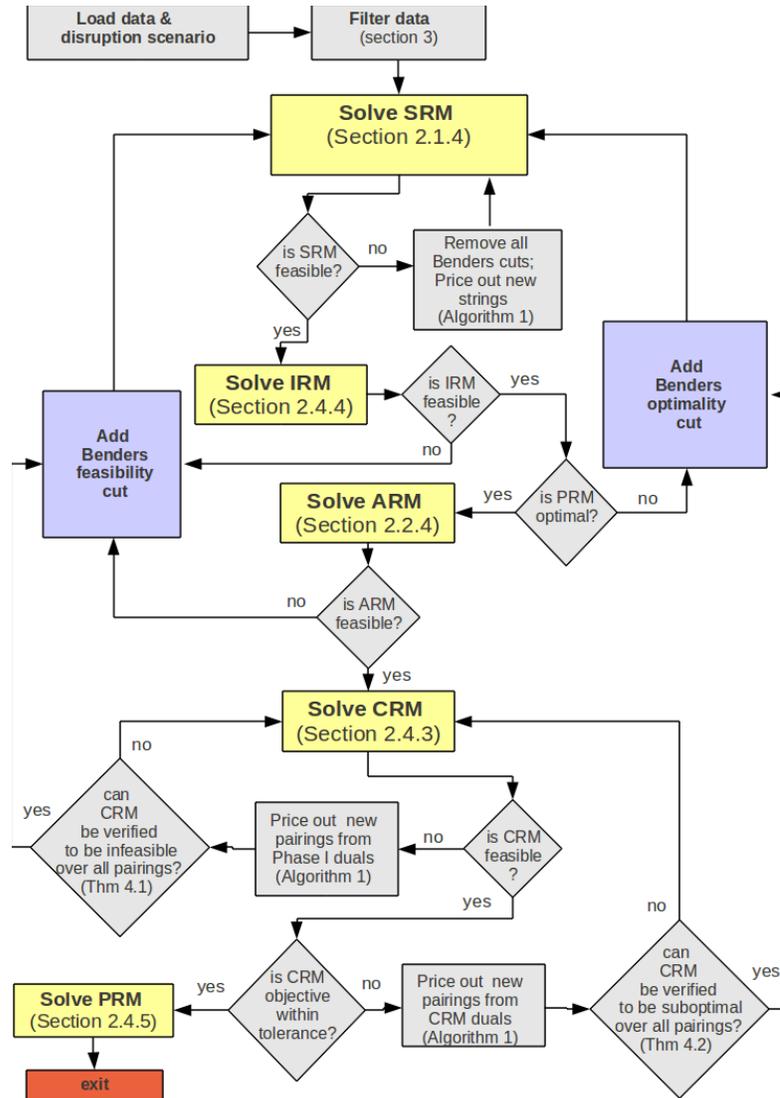
Figure 18: AIR Optimization Module

3.6 Computational Results

Our model is tested using 2007 data from a hub-and-spoke regional airline based in the U.S. with approximately 800 daily flights and two fleet types. The main disruption of interest is a flow rate reduction into and out of the hub, and possibly other stations. We consider a reduction in terms of a certain percentage of scheduled operations as well as a full hub closure for some period of time. Table 4 summarizes the benchmark parameters used in the results obtained. As shown in the table, the SRM cost objective is only to minimize the cost associated with canceling flights, whilst ignoring the cost of assigning equipment to flight strings. An obvious alternative is to penalize all flights whose equipment type deviates from the schedule. The same could be said for assigning individual tails to flight strings in the ARM. The cost of $38 per hour of passenger delay is given by Ball et al. ([18]).

Note that we consider a zero objective on individual crew pairing assignments. This is because the crew recovery problem is quite different from the well-known crew pairing problem where the objective is to minimize the sum of crew pairing assignments known as *pay-and-credit*, which is a complex objective which factors in the total time the crew is away from base, flying hours, and number of duties in a pairing. Deadhead costs are influential to the cost of the entire pairing, and therefore by minimizing deadhead costs during the broken part of a crew pairing, pay-and-credit can be reduced.

The data represented in Table 4 comes from a priori knowledge about the given network and airline under consideration. Of course, different airlines could incorporate their own set of parameters characterizing their own idiosyncratic values. We emphasize the specific values are not important per se, but rather the methodology that determines the set of rescheduling decisions as different sets of parameters could be used to reflect other carriers.

Our goal is to deliver a solution within 30 minutes as agreed upon by our industry partners. While this number is likely greater than the allowable time posed by an OCC coordinator, we emphasize the challenges posed by this particular regional carrier is among the most complex and difficult-to-solve class of problems. Moreover our implementation serves only as a prototype versus production software. A number of ways to expedite our implementation exist including utilizing parallelization and improved computational

Table 4: Benchmark Parameters Used in Computations

parameter	description	value
$c_{e,s}^{\text{assign}}$	cost of assigning equipment $e \in E$ to string $s \in S$	\$0
c_f^{cancel}	cost of canceling flight $f \in F$	\$25,000
$c_{e,s}^n$	cost of assigning tail $n \in AC(e)$ to string $s \in S$	\$0
$c_{k,p}^{\text{assign}}$	cost of assigning crew k pairing p	\$0
d_f^{pairing}	cost of deadheading on flight f within a pairing	\$1,000
d_k^{base}	cost of crew k deadheading to crew base	\$2,000
$c_{i,\gamma}^{\text{delay}}$	cost in passenger goodwill per hour of delay	\$38
c_i^{unassign}	cost of unassigned itinerary for passenger $i \in OD$	\$2,500
$\omega_{i,\gamma}$	weight of passenger itinerary cost in IRM	$\frac{n_i^{\text{PAX}}}{\sum_i n_i^{\text{PAX}}} \forall \gamma \in \Gamma_i$

infrastructure that is likely to be found at an OCC. We emphasize that our model is scalable. For small disruptions that airlines have to deal to every day much less time is needed, while being able to provide an answer for larger scenarios. Even by sacrificing on optimality, our module is likely able to provide an improvement over incumbent methods which often rely on the manual construction of rescheduling decisions.

Our model has been implemented in C++ using Concert/CPLEX 12.2 on a quad-core computing cluster whose head node is a 2.66 GHz Xeon X5355 processor.

Problem Size and Length of Disruption   Section 3.4 discussed how the scope of the recovery operation was limited. Figure 19 shows how the number of disruptable flights grows with respect to the duration of closure at the hub beginning at 8:00 local time. While a one-hour disruption affects nearly half the flights, every flight is disruptable when the length of the disruption reaches 105 minutes. This is partially due to the fact that the

data set comes from a regional carrier whose flight legs are typically short relative to major carriers whose networks span a larger geographical region. This is readily seen as that every tail number has some activity at the hub between 8:00 and 9:15 AM local time.
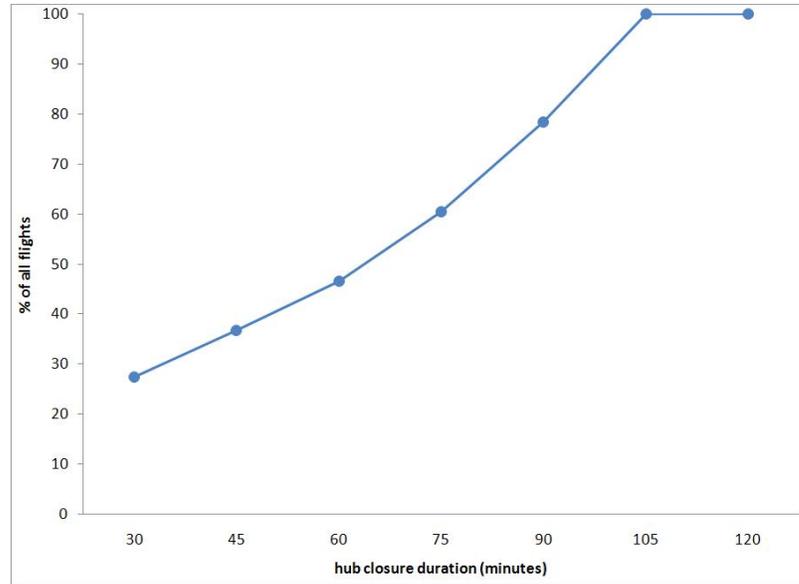
Figure 19: Disruptable Flights and Length of Hub Closure

Build versus Repair of Crew Duty Network   One of the major bottlenecks in the solution process outlined above is the construction of, and generating paths through the crew duty network. Because this network is apt to change for each new scheduling decision made in the master problem, there are two approaches how to manage the crew duty network. The first is to build it once before the iterative process begins, then heuristically repair broken duties and missed connections and repair the original network based on the current scheduling decisions. The second is to construct a new network entirely after each master solution. The obvious tradeoff is computational resources spent constructing the crew duty network and information about the true network. If the time window includes more than one day, the number of connecting duties increases substantially thereby making the CRM even more complex, and the former approach is more plausible. As a first attempt to study the AIR problem, we begin by restricting our analysis to a one-day time window so that the crew duty network can be rebuilt within each iteration. It may be naturally of interest

to take the other approach for larger problems. The multi-day problem would require a different set of algorithms.

3.6.1 Disruption Scenarios

We model three classes of disruption scenarios:

1. 50 % reduction in flow rate (arrivals & departures)

2. 75 % reduction in flow rate (arrivals & departures)

3. 100 % reduction in flow rate (arrivals & departures)

Each scenario will examine four different disruption events characterized by a disruption time, disruption location, and time window shown in Table 5. Scenario 4 considers two disruptions: one at the hub and the other at one of the largest spokes used in the network. Given the growth of problem size on the length of hub closure (see Figure 19), we consider a maximum hub disruption to be 75 minutes, which our heuristic search procedure includes every flight after the disruption. The final column represents the maximum delay considered which has a profound effect on the number of strings being generated. For a two hour hub disruption, the total number of flight strings (that contain no more than 7 flights) increase from under 200,000 using a one hour maximum delay period to more than 2.6 million using a three hour maximum delay period. If a set of passenger itineraries is suboptimal after the 30 minute threshold, the best incumbent solution is given and passed to the ARM and CRM subproblems. The algorithm has timed out only for the largest scenarios in our study.

Table 5: Simulated Disruption Events

event	disruption time	disruption location	time window $\mathcal{T}$	max delay time (minutes)
1	08:00 - 08:30	hub	08:00 - 23:59	90
2	08:00 - 09:00	hub	08:00 - 23:59	120
3	08:00 - 09:15	hub	08:00 - 23:59	150
4	08:00 - 09:00 / 09:00 - 14:00	hub / spoke	08:00 - 23:59	120

3.6.2 Integrated versus Sequential Recovery

We report costs for all subproblems and important metrics that determine quality of solution. We do not report costs for the ARM in the integrated model since it amounts to a feasibility problem, and is always feasible in the sequential module. All times are reported in MM:SS format.

Disruption Scenario 1: 50% Flow Rate Capacity Reduction   Tables 6 and 7 show the first set of results for a 50% flow rate reduction into and out of the hub for the sequential process and integrated process, respectively.

Table 6: Sequential Recovery Summary (50% flow rate reduction)

		Event			
		1	2	3	4
subproblem costs ($)	SRM	0	0	0	150,000
	ARM	0	0	0	INFEAS
	CRM	0	0	0	INFEAS
	PRM	11,653	28,257	55,665	116,471
solution metrics	mean flt delay	20:05	23:34	42:21	41:36
	cancelled flts (%)	0	0	0	4.6
	delayed flts (%)	12.8	59.4	56.2	52.0
	total deadheads	0	0	0	INFEAS
	mean PAX delay	23:09	24:28	45:39	39:15
	unassigned PAX	0	4	5	31
	CPU time	0:58	07:28	17:20	12:02

Table 7: Integrated Recovery Summary (50% flow rate reduction)

		Event			
		1	2	3	4
subproblem costs ($)	SRM	0	0	0	50,000
	CRM	0	0	0	4,000
	PRM	11,653	22,942	46,057	54,820
solution metrics	mean flt delay	20:05	20:34	39:50	33:41
	cancelled flts (%)	0	0	0	1.6
	delayed flts (%)	12.8	35.1	38.0	49.3
	total deadheads	0	0	0	2
	mean PAX delay	23:09	21:47	39:22	33:37
	unassigned PAX	0	3	3	5
	CPU time	1:02	24:41	32:28	36:34

Disruption Scenario 2: 75% Flow Rate Capacity Reduction   Tables 8 and 9 show
the results from reducing capacity by 75%.

Table 8: Sequential Recovery Summary (75% flow rate reduction)

		Event			
		1	**2**	**3**	**4**
subproblem costs ($)	SRM	0	0	0	150,000
	ARM	0	0	INFEAS	INFEAS
	CRM	0	0	INFEAS	INFEAS
	PRM	15,316	29,440	62,316	85,039
solution metrics	mean flt delay	17:57	28:24	46:58	44:01
	cancelled flts (%)	0	0	0	4.6
	delayed flts (%)	28.7	37.7	52.3	54.4
	total deadheads	0	0	INFEAS	INFEAS
	mean PAX delay	22:19	28:52	50:23	44:41
	unassigned PAX	2	4	8	24
	CPU time	1:01	10:02	14:11	14:29

Table 9: Integrated Recovery Summary (75% flow rate reduction)

		Event			
		1	**2**	**3**	**4**
subproblem costs ($)	SRM	0	0	0	100,000
	CRM	0	0	0	5,000
	PRM	15,316	22,198	51,336	40,489
solution metrics	mean flt delay	17:57	28:31	44:01	33:05
	cancelled flts (%)	0	0	0	2.3
	delayed flts (%)	28.7	38.1	42.1	56.4
	total deadheads	0	0	0	3
	mean PAX delay	22:19	20:36	41:44	36:19
	unassigned PAX	2	3	6	7
	CPU time	1:04	23:20	30:56	32:27

Disruption Scenario 3: Hub Closure   Finally we consider a full closure into and out
of a set of stations prohibiting all arrivals and departures within the disruption time which
are shown in Tables 10 and 11.

In both environments, a warm start is provided to the initial SRM that preserves all
scheduled routings incorporating the minimum possible delay with each flight (thereby
initially not considering flight cancellations). As a result the integrated and sequential
solutions may coincide if the warm start is optimal. This occurs in two of the scenarios

Table 10: Sequential Recovery Summary (hub closure)

		Event			
		1	**2**	**3**	**4**
subproblem costs ($)	SRM	0	0	0	175,000
	ARM	0	0	0	INFEAS
	CRM	0	0	0	INFEAS
	PRM	17,979	32,057	56,730	133,573
solution metrics	mean flt delay	28:10	25:28	27:51	49:47
	cancelled flts (%)	0	0	0	4.0
	delayed flts (%)		66.9	64.2	59.3
	total deadheads	0	0	0	INFEAS
	mean PAX delay	17:56	31:46	41:58	43:41
	unassigned PAX	4	4	3	36
	CPU time	0:35	17:50	31:01	20:41

Table 11: Integrated Recovery Summary (hub closure)

		Event			
		1	**2**	**3**	**4**
subproblem costs ($)	SRM	0	0	0	100,000
	CRM	0	0	0	5,000
	PRM	12,186	24,566	41,993	58,300
solution metrics	mean flt delay	19:25	23:24	29:39	34:40
	cancelled flts (%)	0	0	0	2.3
	delayed flts (%)	26.4	40.6	59.1	58.5
	total deadheads	0	0	0	3
	mean PAX delay	16:04	21:02	36:41	41:54
	unassigned PAX	2	4	3	7
	CPU time	1:46	24:09	31:00	24:22

explaining why the integrated recovery framework provides no improvement. Of course, relaxing the warm start will induce the integrated solution to dominate its sequential counterpart. No scenarios were encountered from the integrated model where no integer feasible solution was found to a subproblem after the Benders' framework has terminated.

We note that the 75 minute disruption seems to prohibit obtaining a solution in our 30 minute runtime goal. While about 60% of the flights are initially disruptable from the scheduled routings, all flights are disruptable through the process by which we limit the scope (Section 3.4). Moreover, the number of strings is vastly higher due to a longer maximum flight delay period. The multiple disruption scenario performs better, but is does not always meet the runtime goal in the integrated setting (Tables 7 and 9).

Moreover we note the improvement in solution quality the integrated approach delivers over the sequential one. First, note that 25% of the scenarios show the sequential approach is infeasible where the integrated approach always delivers a solution. Secondly, we note a reduction in the key performance metrics that include flight delay, passenger delay, and cost of recovery. Table 12 shows how the integrated module reduces mean passenger delay, mean flight delay, and passenger reaccommodation costs by averaging across the 50%, 75%, and 100% capacity reduction scenarios. Of particular interest in the behavior of mean passenger delay which is reduced by as much as 14.6% in the 75 minute disruption. The integrated model also reduces passenger reaccommodation costs considerably; saving over half the reaccommodation costs from the multiple disruption scenario.

Table 12: Summary of Improvement from Integrated Model

Event	Performance Metric Improvement		
	Mean Passenger Delay (%)	Mean Flight Delay (%)	PRM cost (%)
30 minute disruption	2.9	13.2	12.9
60 minute disruption	13.7	6.2	22.3
75 minute disruption	14.5	3.3	20.3
multiple disruptions	12.4	25.1	54.2

Another question of interest is how the solution quality changes with respect to input parameters. Figure 20 shows two experiments of interest using the 60-minute hub closure disruption scenario. Panel 20a shows how the cancellation rate changes with respect to the cost of flight cancellations c_f^{cancel}. As mentioned previously, the airline under consideration is highly adverse to flight cancellations due their own idiosyncratic requirements. The figure shows that as long as the cost associated with a cancellation exceeds \$15,000 per flight, the same recovery tactic that considers only delays remains optimal. Cancellations only become desirable when the cancellation penalty is between \$10,000 and \$15,000 per flight. Panel 20b illustrates the tradeoff between the severity of passenger delay and cancellations by changing the cost of unassigned passengers c_i^{unassign}. The solution summarized in Table 11 (setting c_i^{unassign} to \$2,500 for all $i \in OD$) remains the optimal solution for all values c_i^{unassign} that exceed \$1,000. The tradeoffs between passenger delay and flight cancellations change the solution only when the penalty parameter is between \$500 and \$1,000 per passenger.

Therefore the optimal solution attained in the integrated model for the one-hour hub closure are robust with respect to these two input parameters under consideration.
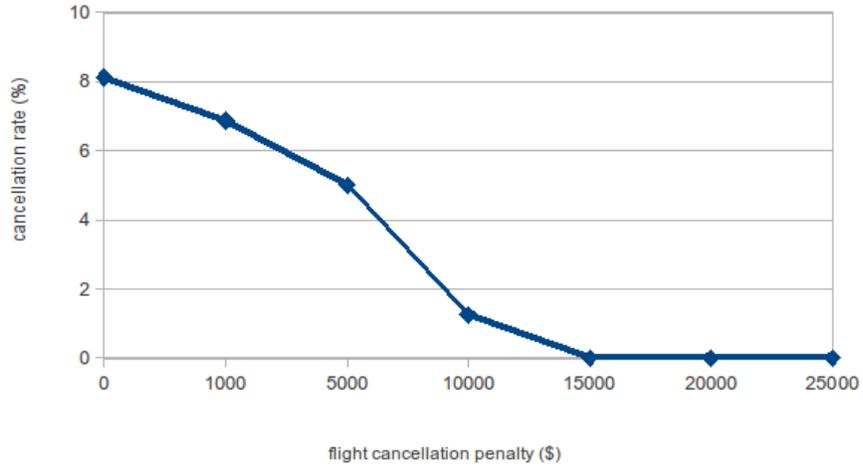
(a) Changing c_f^{cancel}
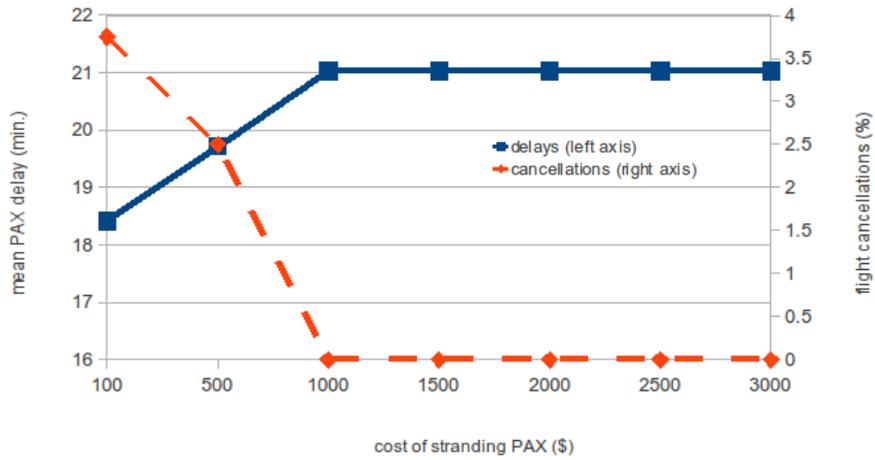
(b) Changing c_i^{unassign}

Figure 20: Sensitivity Analyses for a One-Hour Hub Closure

CHAPTER IV

SIMULTANEOUS ROW AND COLUMN GENERATION

4.1 Introduction

Consider a standard mathematical programming problem of the form

$$z^* = \min\left\{cx : x \in S\right\} \tag{P}$$

where $S = \left\{x \in \mathbb{R}_+^n : Ax = b\right\}$ and n is large. Two classical methods to solve such large-scale problems are decomposition methods and column generation methods.

Decomposition methods seek to decompose (P) into several smaller problems. Primal and dual decomposition techniques seek to partition the columns or rows, respectively of the resource matrix A to exploit the structure of the problem in a way that is amenable to faster computation. Benders' decomposition and Lagrangean decomposition give examples of the respective schemes. In each of these settings, a reformulation of the problem relies on an iterative scheme whereby a restricted master problem is solved over a subset of variables or constraints, and successive cuts are added based on a series of subproblems. This chapter will specifically focus on how Benders' cuts and column generation may be used simultaneously.

Column generation works off a sequence of smaller problems similar to (P). For $N' \subset N$ a problem of the form

$$\widetilde{x} = \arg\min\left\{cx : x \in S(N')\right\} \tag{P'}$$

is solved where $S(N') = \left\{x \in \mathbb{R}_+^{|N'|} : A(N')x = b\right\}$, and $A(N')$ is a submatrix of A matrix obtained by removing all columns whose indices are not contained in N'. While $\widetilde{x}$ is feasible for (P), it is not necessarily optimal. Given $\widetilde{x}$ the *pricing problem* tries to find a variable (column) present in (P') but not in (P) whose reduced cost is strictly negative. If such a column is found it is appended to N' and (P') is re-solved. If no such column is

found, $\widetilde{x}$ is optimal for (P). For all $j \in N$ the reduced cost $\overline{c}_j$ is given by

$$\overline{c}_j = c_j - \pi A_{.,j}$$

where $A_{.,j}$ denotes the jth column of A. Let π^* denote a solution the dual variables for (P') corresponding to $\widetilde{x}$. The pricing problem

$$j^* = \arg\min \left\{ c_j - \pi A_{.,j} : j \in S(N \setminus N') \right\}$$

is solved. If $\pi^* A_{.,j^*} \leq c_{j^*}$ then $\widetilde{x}$ is optimal for (P). Else, $N' \leftarrow N \bigcup \{j^*\}$ and (P') is re-solved until no column prices out.

While decomposition and column generation methods are widely known, they are generally thought of as being mutually exclusive. It is of natural interest to integrate these two paradigms where possible. Given a cut $\pi x \leq \pi_0$ that has been generated with respect to $N' \subset N$ variables, the problem is to determine whether or not the cut remains valid over all variables in N, or if there exists some $j \in N \setminus N'$ such that $\pi x_j > \pi_0$ therefore invalidating the cut for the global problem (P). Surprisingly there has been little work done in the literature regarding this fundamental question. First, a literature review of related work is provided.

4.2   Literature Review

Van Roy [122] proposes an algorithm that simultaneously uses both primal and dual decomposition for problems that exhibit each type of structure. The proposed approach, referred to as *cross decomposition*, adds cuts and columns to the restricted master problem through solutions to the primal and dual subproblems, respectively. The algorithm begins by selecting initial values of Lagrangean multiplier and solves the dual subproblem whose solution is passed to the primal subproblem. If optimality is not verified, the primal solution is used to update the new Lagrangean multipliers and the process continues until it terminates, for which it is shown to do so in a finite number of steps. It is shown that fewer Benders' cuts (from primal decomposition) can be attained at the expense of additional constraints in the Lagrangean relaxation (dual decomposition).

Feillet et. al. [49] propose a methodology to add columns and cuts simultaneously within the context of branch-and-cut-and-price. Given the solution to a restricted master problem, they construct a feasible primal and solution to their respective master problems (the latter of which may be infeasible). The constructed solutions attain the same objective values as the solutions from the restricted problems. If the dual solution is feasible, then the optimality criterion is met. Else, a dual cut is added and the process continues. They show how to reconstruct the solutions from the restricted master solution by illustrating two examples in which they show their method yields considerable improvements in runtime.

Poggi de Aragão and Uchoa [92] introduced an alternative method to Dantzig-Wolfe decomposition by a reformulation into what they call an *Explicit Master* problem which which is equivalent to the relaxation of the master problem from a Dantzig-Wolfe structure, but fixes some of the reduced costs of its variables to zero. The pricing problem associated with the Explicit Master problem is then independent of the master problem allowing cuts to be added to the new master reformulation that do not change the structure of the reduced costs.

Problem-specific applications of managing row and column generation simultaneously are seen in other studies. Nemhauser and Park [87] were the first to provide such a procedure, which will be reviewed in Section 4.4.2.

Barnhart et. al. [20] show how flight strings that are generated dynamically from an aircraft routing problem can be used in conjunction with flight connectivity constraints. For each such constraint, an auxiliary variable is added and the pricing problem is modified in the underlying network to ensure the constraint remains valid.

Barnhart et al. [21] propose using branch-and-price-and-cut to solve integer multicommodity network flow problems. By reformulating the original problem, they are able to create the pricing problem and separation algorithms in a way that do not depend on each other. Moreover, their branching strategy does not explicitly add constraints to the underlying problem so the pricing problem is unchanged.

Fukasawa et al. [53] studies an exact algorithm to solve the capacitated vehicle routing problem (CVRP) that makes use of Lagrangean relaxation and column generation in a

simultaneous manner. They are able to reformulate the problem over the intersection of two polytopes for which 8 families of cuts known in the CVRP literature were augmented by additional columns. The cuts were generated from the reformulation, transformed, and subsequently added to the master problem . Column generation is preformed dynamically trading off the bound quality in the branch-and-cut tree and time spent pricing out new columns. Their method is able to solve problems more than twice the size of instances that can be consistently solved using just either branch-and-cut or column generation.

Valério de Carvalho [121] study how the one-dimensional cutting stock problem may be expedited by using cuts generated from the dual space that ultimately reduce the number of degenerate iterations. The dual space is restricted during column generation, and the primal space is relaxed by inserting new columns. The procedure works off these extended spaces for which a solution can be restored to the original space.

Alves and Valéerio de Carvalho [8] use a branch-and-price-and-cut algorithm on the multiple length cutting stock problem which they show to outperform other exact methods. Similar to [121], for each node in the branch-and-bound tree, they restrict the dual space by adding valid inequalities that accelerates the column generation procedure.

4.3   Review of Benders' Algorithm

This section is a review of the seminal work of Benders [26]. As this work serves as the core of this chapter and the next, a brief review of the algorithm is now provided. Consider the following mathematical programming problem:

$$
\begin{array}{rlllllll}
z = & \min & cx & + & f_1 y_1 & + & f_2 y_2 & + & \cdots & + & f_k y_k \\
\text{s.t.} & & Ax & & & & & & & = & b \\
& & B_1 x & + & F_1 y_1 & & & & & = & d_1 \\
& & B_2 x & & & + & F_2 y_2 & & & = & d_2 \\
& & \vdots & & & & & \ddots & & & \vdots \\
& & B_k x & & & & & + & F_k y_k & = & d_k \\
& & & & & & & & x & \in & \mathbb{Z}^n \\
& & & & & & & & y_j & \in & \mathbb{R}_+^{n_j}, \quad 1 \le j \le k
\end{array}
\tag{P}
$$

83

where $c \in \mathbb{R}_+^n, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, and for $j = 1, 2, \ldots, k, f_j \in \mathbb{R}_+^{n_j}, F_j \in \mathbb{R}^{m_j \times n_j}$, and $d_j \in \mathbb{R}^{m_j}$ (the non-negative cost coefficients are assumed without loss of generality). Notice that (P) is initially assumed to be a mixed-integer programming (MIP) problem, although variants of the algorithm exit for when x is real-valued or the y vectors are discrete. The constraints form a block system that are amenable to decomposition described as follows. Given a feasible x^* for (P), the jth subproblem denoted by $SUB(j; x^*)$ depends only on y_j and thus can be solved in isolation and is formulated as

$$z_j = \min \left\{ f_j y_j \quad \text{s.t.} \quad F_j y_j = d_j - B_j x^*, y_j \in \mathbb{R}_+^{n_j} \right\} \qquad (SUB(j; x^*))$$

whose dual is

$$\omega_j = \max \left\{ \pi^j \left(d_j - B_j x^* \right) \quad \text{s.t.} \quad \pi^j F_j \leq f_j \right\}. \qquad (DSUB(j; x^*))$$

It is assumed throughout that the dual polyhedron $Q_j \equiv \{ \pi \in \mathbb{R}^{m_j} : \pi F_j \leq f_j \}$ is nonempty for all subproblems. Let Π^j denote the set of all dual extreme point from the jth dual polyhedron. The variables x from (P) are referred to as *linking variables* whereas variables y_j are often referred to as *local variables* or *subproblem variables* for the jth subproblem $(SUB(j; x^*))$. Benders' decomposition algorithm is a reformulation of (P) with the following $n + k$ variables:

- The original n linking variables x

- k continuous variables $\eta_1, \eta_2, \ldots, \eta_k$ where

$$\eta_j = \max \left\{ \pi \left(d_j - B_j x^* \right) : \quad \pi \in \Pi^j \right\}.$$

At each step of the algorithm a candidate solution x^* is given from the *Master Problem (MP)*, whose variables contain only the linking variables x and auxiliary variables $\eta_1, \eta_2, \ldots, \eta_k$. In general characterizing Π^j is intractable. Instead, the procedure relies on a proper subset $\widetilde{\Pi}^j$ of Π^j which is appended if the given dual solution is shown to be suboptimal. The *Restricted Master Problem (RMP)* is the same problem as the MP but does not rely on a complete characterization of the dual polyhedra. A solution x^* to the RMP may lead to an infeasible or suboptimal solution for some subproblem where inequalities

are added to the RMP to cut off such solutions. If (x^*, η^*) is a solution from the MP, and for all $j = 1, 2, \ldots, k$ there exists a dual optimum $(\pi^j)^*$ such that $\eta_j^* \geq (\pi^j)^* (d_j - B_j x^*)$, then the procedure terminates with $(x^*, y_1^*, y_2^*, \ldots, y_k^*)$ being optimal for (P). Otherwise, if $\eta_j^* < (\pi^j)^* (d_j - B_i x^*)$ then weak duality has been violated for which $\widetilde{\Pi}^j$ is appended with the inclusion of $(\pi^j)^*$, and the constraint $(\pi^j)^* (d_i - B_i x) \leq \eta_i$ is added to the RMP. This inequality is said to be a *Benders' optimality cut*.

Suppose the dual subproblem is unbounded for a candidate solution x^*. Then by duality, the original primal subproblem $(SUB(j; x^*))$ is infeasible. An unbounded dual extreme ray is a feasible dual vector $r \in \mathbb{R}^{m_j}$ such that $r(d_j - B_j x^*) > 0$. Therefore if R^j denotes the set of all dual extreme rays from the jth dual polyhedron, $r_q^j (d_j - B_j x^*) \leq 0$ for all dual extreme rays $q \in R^j$. Similar to the set of dual extreme points discussed above, obtaining a complete characterization of R^j is usually intractable, so the RMP works iteratively off a subset $\widetilde{R}^j$ of R^j. For a given x^* if the jth dual subproblem is unbounded, then an extreme ray r_q^j is found by which $r^j (d_j - B_j x^*) > 0$, then the set $\widetilde{R}^j$ is appended by r_q^j and the inequality $r_q^j (d_j - B_j x^*) \leq 0$ is added to the RMP. This inequality is said to be a *Benders' feasibility cut*.

A given iteration of the RMP is given by a set $\widetilde{\Pi}_p^j \subseteq \widetilde{\Pi}^j$ of dual extreme points and a set $\widetilde{R}^j \subseteq R^j$ of dual extreme rays for dual polyhedron $j = 1, 2, \ldots, k$. Let $\widetilde{\Pi} = \bigcup_{j=1}^k \widetilde{\Pi}^j$ and $\widetilde{R} = \bigcup_{j=1}^k \widetilde{R}^j$ be the set of all dual extreme points and dual extreme rays, respectively. Given these sets the RMP is formulated as

$$
\begin{aligned}
z \quad = \quad & \min \quad cx + \sum_{j=1}^k \eta_j \\
& \text{s.t.} \quad Ax = b \\
& \qquad \pi_p^j (d_j - B_j x) \leq \eta_j \quad \forall p \in \widetilde{\Pi}^j, \quad j = 1, 2, \ldots, k \qquad (\text{RMP}(\widetilde{\Pi}, \widetilde{R})) \\
& \qquad r_q^j (d_j - B_j x) \leq 0 \quad \forall q \in \widetilde{R}^j, \quad j = 1, 2, \ldots, k \\
& \qquad (x, \eta) \in \mathbb{Z}_+^n \times \mathbb{R}_+^k.
\end{aligned}
$$

If y_j^* are such that $f_j y_j^* \leq \eta_j^*$ for all $j = 1, 2, \ldots, k$, an optimal solution (x^*, y^*) is attained.

While there are subtle different implementations of the algorithm, a standard one is seen in Algorithm 2.

Algorithm 2 Benders' Decomposition Algorithm

1: **given:** an initial feasible solution x^* to RMP$(\emptyset, \emptyset)$
2: **initialize** $isOptimal = $ false
3: **for all** $j = 1, 2, \ldots, k$ **do**
4:     Solve $DSUB(j; x^*)$
5:     **if** $DSUB(j; x^*)$ has a finite optimum **then**
6:         instantiate $\widetilde{\Pi}^j = (\pi^j)^*$
7:     **else if** $DSUB(j; x^*)$ is unbounded **then**
8:         instantiate $\widetilde{R}^j = (r^j)^*$
9:     **end if**
10: **end for**
11: **while** $isOptimal = $ false **do**
12:     solve RMP, let (x^*, η^*) denote an optimal solution
13:     **for all** $j = 1, 2, \ldots, k$ **do**
14:         solve $DSUB(j; x^*)$; obtain an optimal extreme point $(\pi_p^j)^*$ or extreme ray $(r_q^j)^*$
15:         **if** $DSUB(j; x^*)$ has a finite optimum **then**
16:             **if** $\eta_j^* < (\pi_p^j)^* (d_j - B_j x^*)$ **then**
17:                 $\widetilde{\Pi}^j \leftarrow \widetilde{\Pi}^j \cup (\pi_p^j)^*$
18:                 add Benders feasibility cut $(\pi_p^j)^* (d_i - B_i x) \leq \eta_j$ to RMP
19:                 **break; return** to RMP (line 12)
20:             **end if**
21:         **else**
22:             $\widetilde{R}^j \leftarrow \widetilde{R}^j \cup (r_q^j)^*$
23:             add Benders optimality cut $(r_q^j)^* (d_i - B_i x) \leq 0$ to RMP
24:             **break; return** to RMP (line 12)
25:         **end if**
26:     **end for**
27:     $isOptimal = $ true
28: **end while**

4.4 Simultaneous Benders' Cut and Column Generation Algorithm

Consider a problem of the form

$$
\begin{aligned}
z^* = \quad &\min \quad c^T x \quad + \quad \sum_{i=1}^{k} f^i y_i \\
&\text{s.t.} \qquad Ax \qquad\qquad\qquad = \quad b \\
&\qquad\qquad B^i x \quad + \quad G^i y_i \qquad = \quad d_i, \quad i = 1, 2, \ldots, k \\
&\qquad\qquad (x, y_1, y_2, \ldots, y_k) \quad \in \quad \mathbb{Z}_+^n \times \mathbb{R}_+^{n_1} \times \cdots \times \mathbb{R}_+^{n_k}
\end{aligned}
$$

where $c \in \mathbb{R}_+^n$, $f^i \in \mathbb{R}_+^{n_i}$, $G^i \in \mathbb{R}^{m_i \times n_i}$, and $d_i \in \mathbb{R}^{m_i}$ for $i = 1, 2, \ldots, k$.

Such a structure is amenable to Benders' decomposition. Note that the dual polyhedron associated with subproblem i is $Q^i \equiv \{\pi \in \mathbb{R}^{m_i} : \pi G^i \le f^i\}$. Let $\Pi^i = \{\pi_1^i, \pi_2^i, \ldots, \pi_{\alpha_i}^i\}$ denote the set of all extreme points of Q^i and $R^i = \{r_1^i, r_2^i, \ldots, r_{\beta_i}^i\}$ denote all extreme rays of Q^i.

The complete Master Problem is of the form

$$
\begin{aligned}
z^* = \quad & \min c^T x + \sum_i^k \eta_i \\
\text{s.t.} \quad & Ax = b \\
& \pi_j^i \left(d_i - B^i x\right) \le \eta_i, \quad \forall \pi_j^i \in \Pi^i, \quad i = 1, 2, \ldots, k \qquad (\mathrm{MP}(\Pi, R)) \\
& r_j^i \left(d_i - B^i x\right) \le 0, \quad \forall r_j^i \in R^i, \quad i = 1, 2, \ldots, k \\
& (x, \eta_1, \ldots, \eta_k) \in \mathbb{Z}_+^n \times \mathbb{R}_+ \times \cdots \times \mathbb{R}_+.
\end{aligned}
$$

As the sets Π^i, R^i are generally not completely characterized, Benders' algorithm seeks to solve a series of restricted master problems where the sets Π^i, R^i in $(\mathrm{MP}(\Pi, R))$ are replaced by $\widetilde{\Pi}^i \subseteq \Pi^i$ and $\widetilde{R}^i \subseteq R^i$ and form the basis for the Restricted Master Problem (RMP). Let $\widetilde{\Pi} = \bigcup_{0 \le i \le k} \widetilde{\Pi}^i$ and $\widetilde{R} = \bigcup_{0 \le i \le k} \widetilde{R}^i$. The RMP is then given by

$$
\begin{aligned}
z^* = \quad & \min c^T x + \sum_i^k \eta_i \\
\text{s.t.} \quad & Ax = b \\
& \pi_j^i \left(d_i - B^i x\right) \le \eta_i, \quad \forall \pi_j^i \in \widetilde{\Pi}^i, \quad i = 1, 2, \ldots, k \qquad (\mathrm{RMP}(\widetilde{\Pi}, \widetilde{R})) \\
& r_j^i \left(d_i - B^i x\right) \le 0, \quad \forall r_j^i \in \widetilde{R}^i, \quad i = 1, 2, \ldots, k \\
& (x, \eta_1, \ldots, \eta_k) \in \mathbb{Z}_+^n \times \mathbb{R}_+ \times \cdots \times \mathbb{R}_+.
\end{aligned}
$$

Consider a solution $(x^*, \eta_1^*, \ldots, \eta_k^*)$ to $(\mathrm{RMP}(\widetilde{\Pi}, \widetilde{R}))$ for which $y_1^*, y_2^*, \ldots, y_k^*$ are solutions for the subproblems. x^* is optimal for $(\mathrm{MP}(\Pi, R))$ if and only if for all $i = 1, 2, \ldots, k$, $\eta_i^* \ge \pi^* \left(d_i - B^i x^*\right)$ where $\pi^* \in Q^i$.

While the dual polyhedron Q^i need not be fully characterized, traditional Benders' decomposition assumes that the set of all subproblem variables y_i are all generated. However this may be impractical for a host of applications whose subproblems are combinatorial. Let J_i index the set of all variables for subproblem i. Suppose a column generation scheme is being used in which at a given iteration $J_i' \subset J_i$ columns have been generated. Then the cut coefficients π_j^i (for an optimality cut) or r_j^i (for a feasibility cut) are valid only over the

set J_i'. The cut is said to be *invalid* if there exists $j \in J_i \setminus J_i'$ such that the ith subproblem becomes feasible or infeasible over $J_i' \bigcup \{j\}$. The fundamental question that arises is how one can determine from J_i' if the given *candidate cut* generated over J_i' is valid over J_i. The method proposed in this chapter relies on a certificate that guarantees the validity of a candidate cut. The certificate is based on the Theorem of the Alternative, which is a corollary to Farkas' Lemma.

Theorem 4.4.1 (Theorem of the Alternative). *Given* $(A, b) \in \mathbb{R}^{m \times n} \times \mathbb{R}^m$, *exactly one of the following two systems has a solution:*

(i) $\quad \{x \in \mathbb{R}_+^n : Ax = b\}$

(ii) $\quad \{\pi \in \mathbb{R}^m : \pi b < 0, \pi A \geq 0\}$

Theorem 4.4.1 gives rise to a number of variants of alternative systems. One that will be of use to our approach is given as follows.

Corollary 4.4.2. *Given* $(A, b) \in \mathbb{R}^{m \times n} \times \mathbb{R}^m$, *exactly one of the following two systems has a solution:*

(i′) $\quad \{x \in \mathbb{R}_+^n : Ax = b\}$

(ii′) $\quad \{(\pi, \Delta) \in \mathbb{R}^m \times \mathbb{R} : \pi b + \Delta < 0, \pi A + \Delta \geq 0\}$

Proof. Consider system (i′) with three mutually exclusive conditions: $\mathbf{1}x = 1$, $\mathbf{1}x < 1$, and $\mathbf{1}x > 1$ (where $\mathbf{1}$ denotes a vector of ones).

Case 1: $\mathbf{1}x = 1$. Then for the system

$$\widetilde{A} \equiv \begin{pmatrix} A \\ \mathbf{1} \end{pmatrix} \quad \text{and} \quad \widetilde{b} = \begin{pmatrix} b \\ 1 \end{pmatrix}$$

the result follows from Theorem 4.4.1 by letting $\pi = (\pi, \Delta) \in \mathbb{R}^m \times \mathbb{R}$.

Case 2: $\mathbf{1}x < 1$. The alternative system for (i′) is then

$$\pi A + \Delta \mathbf{1} \geq 0$$

$$\pi b + \Delta < 0$$

$$\Delta \geq 0.$$

Case 3: $\mathbf{1}x > 1$. By rewriting the additional condition as $-\mathbf{1}x < -1$ and applying the result the alternative system is

$$\pi A - \Delta \mathbf{1} \geq 0$$

$$\pi b - \Delta < 0$$

$$\Delta \geq 0$$

Thus (i') has a solution if and only if (ii') does not. $\qquad\square$

For a given pair $(\alpha, t) \in \mathbb{R}^m \times \mathbb{R}$ let $H(\alpha, t)$ denote the hyperplane $H(\alpha, t) = \{x \in \mathbb{R}^m : \alpha x = t\}$. The geometry of the previous results shows that a hyperplane $H(\pi, \Delta)$ separates the right-hand side vector b from the convex hull of the columns of A, $\text{conv}\{A_1, A_2, \ldots, A_n\}$, if and only if the system $\{x : Ax = b, x \geq 0\}$ has no solution.

Returning to the Benders framework from above suppose that the ith subproblem is infeasible at a given iteration over $J_i' \subset J_i$. The preceding result provides a basis for validating that the subproblem is infeasible over all J_i that can be obtained in the column generation phase, which shows the validity of a given candidate Benders' cut. The first result gives a simple sufficient condition verifying the *infeasibility* of the subproblem over J_i, even with only J_i' variables having been generated. Intuitively, the result states that if the solution to the pricing problem is sufficiently far enough away from the vector $d - B^i x^*$, then one can construct a solution to the alternative system.

Theorem 4.4.3. *Suppose that for a given solution x^* from the restricted master problem $(RMP(\widetilde{\Pi}, \widetilde{R}))$ there exists some $i \in \{1, 2, \ldots, k\}$ such that subproblem i is infeasible over a set $J_i' \subset J_i$. Let $j^* = \arg\max\left\{\pi G_j^i : j \in J_i \setminus J_i'\right\}$ denote the solution to the pricing problem where π represents the dual variables associated with the Phase I LP of the subproblem. If $\pi G_{j^*}^i < \pi\left(d^i - B^i x^*\right)$, then the subproblem is infeasible over all J_i.*

Proof. Suppose that the ith subproblem is infeasible for a given x and J_i'. By Corollary 4.4.2, it suffices to show the existence of a solution $(\alpha, \Delta) \in \mathbb{R}^{m_i} \times \mathbb{R}$ to the alternative system

$$\begin{aligned}
\alpha\left(d^i - B^i x^*\right) &< \Delta \\
\alpha G_j^i &\geq \Delta \quad \forall j \in J
\end{aligned} \qquad\qquad \text{(ALT-FEAS}(i)\text{)}$$

Consider the Phase I problem

$$\min\left\{ \mathbf{1}s : G^i y + Is = d^i - B^i x, (y, s) \in \mathbb{R}_+^{|J_i'|} \times \mathbb{R}_+^{m_i} \right\}.$$

Since the subproblem is infeasible, $\pi\left(d^i - B^i x\right) > 0$ by strong duality where π denotes the dual variables to the Phase I LP.

Consider now the pricing problem for which a new column $A_{\cdot j^*}$ is returned, i.e.

$$j^* \in \arg\max\left\{ \pi G_j^i : j \in J_i \setminus J_i' \right\}.$$

Therefore for all $j \in J_i$, $\pi G_j^i \le \pi G_{j^*}^i$ and $G_{j^*}^i < \pi\left(d^i - B^i x\right)$ by assumption. Setting $\alpha = -\pi$ and $\Delta = -G_{j^*}^i$ shows the existence of a solution for (ALT-FEAS(i)), and thus shows the result by by Corollary 4.4.2. □

Figure 21 illustrates the geometry behind Theorem 4.4.3. Given the subproblem is infeasible over a subset J_i', there exists a hyperplane $H(\pi, 0)$ separating $d^i - B^i x$ and $conv\left\{ G_1^i, G_2^i, \ldots, G_{|J_i'|}^i \right\}$. If the solution to the pricing problem is small enough, $H(\pi, 0)$ can be affinely transformed to some new hyperplane $H(\pi, \Delta)$ where $\Delta \equiv \pi G_{j^*}^i$ that separates $d^i - B^i x$ from $conv\left\{ G_1^i, G_2^i, \ldots, G_{|J_i|}^i \right\}$.

Theorem 4.4.3 showed how Benders' *feasibility* cuts can be handled concurrently with column generation in a subproblem. However an analogous result holds from the following result with respect to Benders' *optimality* cuts.

Theorem 4.4.4. *Let (x^*, η^*) be a given solution from the restricted master problem (RMP($\widetilde{\Pi}, \widetilde{R}$)). Suppose there exists some $i \in \{1, 2, \ldots, k\}$ for which subproblem i is suboptimal over a set $J_i' \subset J_i$. Let j^* denote a newly generated column so that*

$$j^* \in \arg\min\left\{ \underbrace{f_j^i - \pi G_j^i}_{= \overline{f}_j^i} : j \in J_i \setminus J_i' \right\}.$$

If $\overline{f}_{j^*}^i > \eta_i^* - \pi\left(d^i - B^i x^*\right)$, then the subproblem remains suboptimal over all J_i.
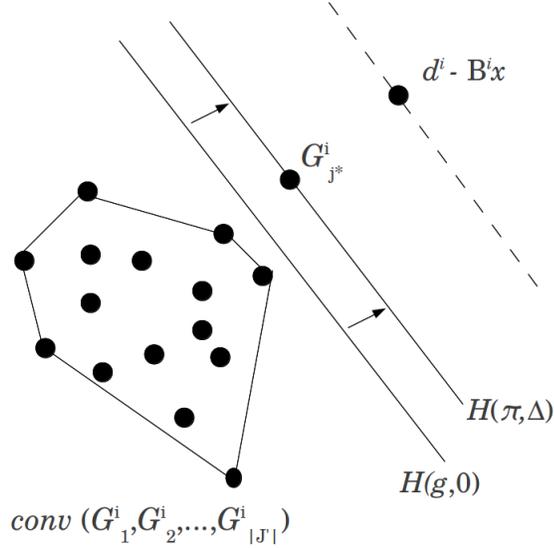
Figure 21: Illustration of Theorem 4.4.3

Proof. In order to show that under the assumption listed, it suffices to show the system

$$
y \in \mathbb{R}_+^{m_i} : \left\{
\begin{array}{rl}
G^i y & = d^i - B^i x^* \\
f^i y & \leq \eta_i^* \\
y_j & \geq 0 \quad \forall j \in J_i
\end{array}
\right\}
\tag{36}
$$

does not have a solution. By Theorem 4.4.1 and Corollary 4.4.2 this is equivalent to showing that there exists a solution $(\alpha, \beta, \Delta) \in \mathbb{R}^{m_i} \times \mathbb{R}_+ \times \mathbb{R}$ to the following alternative system:

$$
\begin{array}{rl}
\alpha \left(d^i - B^i x^* \right) + \beta \eta_i^* & < \Delta \\
\alpha G_j^i + \beta f_j^i & \geq \Delta \quad \forall j \in J_i \\
\beta & \geq 0
\end{array}
\tag{ALT-OPT(i)}
$$

Note that $(-\pi, 1, 0)$ is a solution to (ALT-OPT(i)). The first condition yields $\pi \left(d^i - B^i x^* \right) > \eta_i^*$ which holds since the subproblem is suboptimal over J_i'. The latter condition amounts to $\overline{f}_j^i \geq 0 \quad \forall j \in J_i'$ which holds as the optimality criterion over the set J_i'. Now let us consider the alternative system over the set $J_i \setminus J_i'$. Let $\Delta^i \equiv \min \left\{ \overline{f}_j^i : j \in J_i \setminus J_i' \right\}$ be the minimum reduced cost from the pricing problem. Observe

91

$\overline{f}^i_j \geq \Delta^i \quad \forall j \in J_i \setminus J'_i$. Then $(-\pi, 1, \Delta)$ is a solution to (ALT-OPT(i)): the first condition holds by assumption (bounding the optimality gap) and the second by construction (bounding reduced cost). The first condition in (ALT-OPT(i)) holds by assumption. The second condition holds for all $j \in J'_i \cup (J_i \setminus J'_i) = J_i$ showing the subproblem is infeasible for all J_i given x^*. $\qquad \square$

A geometric argument holds analogous to that seen in the proof of Theorem 4.4.3 to a higher-dimension space. For a problem of the form

$$\min \left\{ cx + fy : Ax = b, Bx + Gy = d, (x, y) \in \mathbb{Z}^n_+ \times \mathbb{R}^m_+ \right\}$$

Algorithm 3 summarizes how Benders' cuts and column generation are handled simultaneously.

The approach above is similar to recent work done by Codato and Fischetti [52] that seek to improve the selection of Benders' cuts for a MIP with no column generation (i.e. $J_i = J'_i$ for all $i = 1, 2, \ldots, k$). They observe that Benders' cut separation can be posed by a feasibility problem of the form (36) attained by minimizing an objective of 0 while re-writing the second constraint as $-f^i y \geq -\eta^*$. This is equivalent to evaluating the dual problem

$$\max \left\{ r \left(d^i - B^i x^* \right) - r_0 \eta^*_i : r G^i_j - r_0 f^i_j \leq 0 \quad \forall j \in J_i, \quad (r, r_0) \in \mathbb{R}^{m_i} \times \mathbb{R}_+ \right\} \qquad (37)$$

which is unbounded if x^* is suboptimal for the given subproblem (as the dual polyhedron contains the origin). By normalizing the objective value of (37) one may define the truncated cone truncated cone

$$T^i = \left\{ (r, r_0) \in \mathbb{R}^{m_i} \times \mathbb{R}_+ : \quad r G^i_j - r_0 f^i_j \leq 0 \quad \forall j \in J'_i, \quad r \left(d^i - B^i x^* \right) - r_0 \eta^*_i = 1 \right\} \quad (38)$$

whose vertices have been shown by Gleeson and Ryan [58] to define the support for the rows of the Minimal Infeasible Subsystem (MIS) corresponding to the rows of (36). By defining coefficients (γ, γ_0) one can generate violated cuts of the rows of (36) by solving the problem

$$\max \left\{ \gamma r + \gamma_0 r_0 : (r, r_0) \in T^i \right\} \qquad (39)$$

Algorithm 3 A Simultaneous Row and Column Generation Algorithm

1: given master solution x^*, κ^*, η^*, tolerance parameter $\epsilon > 0$ and variable set $J' \subset J$
2: solve subproblem
3: **if** subproblem is feasible **then**
4:     **if** (x^*, y^*) is optimal **then**
5:         **terminate**, return (x^*, y^*) as an optimal solution
6:     **else**
7:         generate candidate optimality cut $\pi x \leq \pi_0 + \eta$
8:         get dual variables π
9:         price out; let $j^* = \arg\min \{f_j - \pi G_j : j \in J \setminus J'\}$
10:         **if** $\overline{f}_{j^*} > \eta_i^* - \pi(d - Bx^*)$ **then**
11:             cut is valid over all J (by Theorem 4.4.4), add cut $\pi x \leq \pi_0$ to master problem
12:             update $J' \leftarrow J' \bigcup \{j^*\}$ and return to master problem
13:         **else**
14:             update $J' \leftarrow J' \bigcup \{j^*\}$ and re-solve subproblem
15:         **end if**
16:     **end if**
17: **else**
18:     solve Phase I LP, let ρ denote dual variables
19:     price out; let $j^* = \arg\max \{\rho G_j : j \in J \setminus J'\}$
20:     **if** $\rho G_{j^*} \leq 0$ **then**
21:         **terminate**, no column prices out so problem is infeasible
22:     **else**
23:         **if** $\pi G_{j^*} < \rho(d - Bx^*)$ **then**
24:             problem is infeasible over x^* (by Theorem 4.4.3)
                obtain feasibility cut $rx \leq r_0$ and to master problem
25:         **else**
26:             update $J' \leftarrow J' \bigcup \{j^*\}$ and re-solve subproblem
27:         **end if**
28:     **end if**
29: **end if**

which have been shown to expedite computation to various network design problems. The difficulty of this procedure is to choose values of the objective function coefficients that yield stronger cuts.

Using MIS information to generate Benders' cuts was an idea first addressed by Hooker [67]. This idea has been extended by Codato and Fischetti [40] to generate combinatorial Benders' cuts that exist for conditional relations between inequalities.

The relationship between the method for simultaneously generating Benders' cuts and subproblem variables, and that of selecting strong Benders' cuts given by Fischetti et al. [52] is summarized in the subsequent result.

Theorem 4.4.5. *Any feasibility cut that is generated by Algorithm 3 is a cut that may be generated by solving (37).*

Proof. For a feasibility cut to have been added in accordance to Algorithm 3 the condition $\overline{f}_{j^*} > \eta^* - \pi\left(d - Bx^*\right)$ must be met. This is equivalent to $\pi G_{j^*} + \eta^* < f_i + \pi\left(d - Bx^*\right)$. My maximizing the right-hand side, one obtains the objective function in (39) for $(r, r_0) = (G_{j^*}, \eta^*)$.

Since the subproblem is optimal of J'_i then $f^i_j - \pi G^i_j \geq 0$ for all $j \in J'_i$. Moreover, as x^* induces a suboptimal solution for the ith subproblem, $\pi\left(d^i - B^i x^*\right) > \eta^*_i$. By normalizing this difference to unity, the polyhedron (38) contains the solution $(\pi, 1)$, and therefore generates a violated cut. $\qquad \square$

Because the result derived in this chapter applies to systems that may or may not contain all columns, this result is a generalization of Fischetti et al. [52].

4.4.1 Application to Airline Integrated Recovery

The simultaneous row and column generation procedure is first tested in the Airline Integrated Recovery (AIR) model as presented in Chapter 3. The Crew Recovery Model (CRM) employs column generation of crew pairing variables and therefore exhibits the structure necessary to use the method proposed above.

Recall that $y_{k,p}$ are binary variables that assign a crew member $k \in K$ to a pairing p. Let P_k denote the set of all pairings eligible for crew k and suppose that $P'_k \subset P_k$ represent

the variables that have been generated. A *deadhead* occurs when a crew member is assigned to a flight but does not operate the flight as an active crew. Deadheads can either be within a pairing in which the deadhead is to position the crew to operate a subsequent flight, or can be used to deadhead back to the crew base, typically employed if no (legal) schedule is available for the crew member during a disruption. Let s_f denote the number of crew members who deadhead on a flight f and ν_k are binary variables that equal 1 if crew k is to deadhead back to their given crew base.

Given re-timing and cancellations decisions, the Crew Recovery Model (CRM) is given by

$$\min \quad \sum_{k \in K} \sum_{p \in P} c_{k,p}^{\text{assign}} y_{k,p} + \sum_{f \in F} c_f^{\text{dhd}} s_f + \sum_{k \in K} c_k^{\text{base}} \nu_k \tag{40}$$

$$\text{s.t.} \quad \sum_{k} \sum_{p \ni f} y_{k,p} - s_f = 1 - \kappa_f \quad \forall f \in F \tag{41}$$

$$\sum_{p} y_{k,p} + \nu_k = 1 \quad \forall k \in K \tag{42}$$

$$(y, s, \nu) \in \{0,1\}^{|K| \times |P|} \times \mathbb{Z}_+^{|F|} \times \{0,1\}^{|K|}.$$

The objective (40) seeks to minimize the total cost of assigning crew members to their (possibly new) pairings, as well as assigning deadheads - both for deadheading within a pairing as well as deadheading back to their base. Constraints of the form (41) are cover constraints ensuring that all flights that are not cancelled are contained in at least one pairing. Variables s_f correspond to surplus variables that capture crew deadheads within a pairing. (42) are crew assignment constraints that ensure crew members either be assigned to precisely one duty or are to deadhead back to their base.

Columns are generated through a crew's duty network. Let $G^k = (\mathcal{D}, \mathcal{A})$ denote the duty network for crew k where $\mathcal{D}$ denotes the set of all eligible duties and $\mathcal{A}$ represents the connection between successive duties. A connection between duty i and j is valid if the arrival of the last flight in duty i coincides with the departure of the first flight in duty j and a host of legality requirements are met (for example, allowing sufficient rest time). An artificial source and sink node are added denoted by s and t, respectively. An arc (s, d)

exists if the initial flight of duty d departs from where the crew member is at the time of the disruption. Moreover an arc (d, t) is added if the final flight of duty d arrives at the station the crew is required to be at the end of the disruption. Given the flight cover dual variables the reduced cost of all arcs are computed, and discarded if the reduced cost exceeds some parameter $\epsilon > 0$. All $s - t$ paths are generated over the reduced network that correspond to newly generated crew pairings, and whose total reduced cost is within ϵ of the minimum reduced cost.

For notational convenience let π and ρ denote coefficients of the Benders' cut that correspond to a dual extreme ray (for a feasibility cut) or dual extreme point (for an optimality cut) over P'_k. The candidate Benders' cuts are then

$$\sum_{f \in F} (1 - \kappa_f) \pi_f + \sum_{k \in K} \rho_k \leq \begin{cases} 0 & \text{if feasibility cut} \\ \eta & \text{if optimality cut} \end{cases}$$

where η^{CRM} represents the master variable governing the optimality cuts. These are said to be *candidate cuts* as their validity is only certain over P'_k but not necessarily over P_k. The following results give certificates verifying the validity of these cuts using the framework introduced above. Theorem 4.4.3 then gives the following result that provides a sufficient condition for when the SRM solution is infeasible over all crew pairings.

Lemma 4.4.6. (Extending CRM feasibility cuts over new pairings) *Suppose the CRM is infeasible over a subset of pairings $P' \subset P$. Let $\{\pi_f\}, \{\rho_k\}$ denote the duals corresponding to the Phase I LP-Relaxation of the CRM. If*

$$\sum_{f \in F} (1 - \kappa_f) \pi_f + \sum_{k \in K} \rho_k > \max_{k \in K} \left\{ \max_{p \in P_k \backslash P'_k} \left(\sum_{f \in p} \pi_f + \rho_k \right) \right\}$$

then the CRM is infeasible over all P, and the candidate Benders feasibility cut is valid over all strings and pairings.

The analog of Lemma 4.4.6 for the CRM optimality cut is seen in Lemma 4.4.7, which is derived from Theorem 4.4.4.

Lemma 4.4.7. (Extending CRM optimality cuts over new pairings) *Suppose the CRM is suboptimal over a subset of pairings $P' \subset P$. Let $\{\pi_f\}, \{\rho_k\}$ denote the CRM duals, and*

let η^* denote the continuous master variable corresponding to the optimal objective of the CRM. If

$$\sum_{f \in F} (1 - \kappa_f) \pi_f + \sum_{k \in K} \rho_k + \min_{k \in K} \left\{ \min_{p \in P_k \setminus P'_k} c_{k,p} - \sum_{f \in p} \pi_f - \rho_k \right\} > \eta^*$$

then the CRM is suboptimal over all P, and the candidate Benders optimality cut is valid over all strings and pairings.

If the given sufficient condition does not exist, then the candidate Benders cut is not added to the RMP, but the subproblem is resolved with the larger set of generated pairings and the procedure continues until either the certificate is found, or the newly generated variables induce feasibility or suboptimality.

Algorithm 4 shows how Algorithm 3 is applied to a tangible problem for the case of feasibility cuts only.

In order to test the efficacy of the method above, the AIR model is benchmarked against the incumbent method whose algorithm is the same but does not check for the given sufficient conditions. In the incumbent method, cuts are added only after no other variables price out. Given that enumerating paths over the possibly dense crew duty network is generally time-consuming, the above framework has the potential to deliver a solution in considerably shorter time, which is of considerable value to a problem whose solution needs to be found in a reasonable runtime.

Of interest are both the number of iterations required and total runtime. Similar data is used to that from Chapter 3 with some changes to the flight schedule in order reduce the number of feasible solutions from the SRM to the CRM to illustrate the improvements to the new model.

Table 13 shows how many cuts and column generation calls are used in the incumbent approach versus the SRCG algorithm. The runtime results are shown in Table 14.

The results show that the SRCG method solves in over 6.15% faster on average (Table 14) relative to the incumbent procedure. While more cuts are added in the SRGC method, the increased runtimes associated with more complex RMP models are more than offset by the savings in time associated with generating paths over the crew duty networks (Table

Algorithm 4 Managing Column Generation and Benders Feasibility Cuts Simultaneously

1: **solve** LP-Relaxation for CRM
2: **initialize** $validCut$ = false
3: **if** CRM is infeasible over P' **then**
4:    Extract dual extreme ray (π_f, ρ_k) and Phase-I duals $\left(\pi_f^I, \rho_k^I\right)$.
5:    Let $\sum_{f \in F} \left(1 - \kappa_f\right) \pi_f + \sum_{k \in K} \rho_k \leq 0$ denote the candidate Benders feasibility cut
6:    **for all** crew $k \in K$ **do**
7:       Construct subgraph $\widetilde{G}^k(D, A)$ of crew duty network G^k from $\left(\pi_f^I, \rho_k^I\right)$
8:       Generate new columns P_k^{new} over the ϵ-residual network over $\widetilde{G}^k(D, A)$
9:       **if** a new column exhibits a negative reduced cost **then**
10:          Set $\Delta_k = \max_{p \in P_k} \sum_{f \in p} \pi_f^I + \rho_k^I$
11:       **else**
12:          Set $\Delta_k = 0$
13:       **end if**
14:    **end for**
15:    Set $\Delta = \max_{k \in K} \Delta_k$
16:    **if** $\sum_{f \in F} \left(1 - \kappa_f\right) \pi_f^I + \sum_{k \in K} \rho_k^I > \Delta$ **then**
17:       set $validCut$ = true
18:    **end if**
19:    **if** $validCut$ = true **then**
20:       add candidate Benders cut to master problem
21:    **else**
22:       update columns $P' \leftarrow P' \bigcup_{k \in K} P_k^{\text{new}}$, and re-solve CRM relaxation
23:    **end if**
24: **end if**

Table 13: Benders' Cuts and Generated Paths in AIR

flow rate reduction	disruption length	CRM cuts generated		new paths generated	
		incumbent	SRCG	incumbent	SRGC
50%	30 min hub	1	1	62	62
	60 min hub	2	2	456	456
	75 min hub	2	3	2779	1777
	60 min hub, 200 min spoke	5	7	880	639
75%	30 min hub	0	0	47	47
	60 min hub	1	1	303	303
	75 min hub	2	3	2820	1283
	60 min hub, 200 min spoke	4	6	554	309
100%	30 min hub	0	0	46	46
	60 min hub	1	2	241	137
	75 min hub	2	3	3806	2840
	60 min hub, 200 min spoke	3	6	667	408

13). Recall that all computational experiments are from a single-day time window. With longer time horizons spanning multiple days it is believed that the differences would be even more pronounced as the crew duty networks become exponentially more complex.

4.4.2 Edge Coloring

The preceding sections illustrate how rows and columns can be handled in a simultaneous fashion when the underlying structure is solved by Benders' decomposition. It is now considered how the approach above may be used in order to solve a more generalized problem from combinatorial optimization. Given an undirected graph $G = (V, E)$ an edge coloring of G is a collection of independent sets so that all edges incident to every vertex receives different colors. The *edge chromatic index* of G is the minimum number of colors used in a coloring of G, and is denoted by $\chi(G)$. The *edge coloring problem* is to find a coloring of G using $\chi(G)$ colors and is shown by Holyer [66] to be NP-Complete.

Now consider 3-regular graphs. Vizing [124] showed that for all simple graphs $\chi(G) = \Delta(G)$ or $\Delta(G) + 1$ where $\Delta(G)$ represents the maximum degree of the vertices of G. Therefore the edge coloring problem of a 3-regular graph is to determine if $\chi(G) = 3$ or 4.

Table 14: Incumbent vs. SRCG Runtimes in AIR

flow rate reduction	disruption length	incumbent (sec)	SRCG (sec)	improvement
	30 min hub	181	158	0.0%
	60 min hub	1987	1991	0.0%
50%	75 min hub	3724	3512	5.7%
	60 min hub, 200 min spoke	2943	2670	9.3%
	30 min hub	114	115	-0.1%
	60 min hub	1739	1745	0.0%
75%	75 min hub	3280	2791	14.9%
	60 min hub, 200 min spoke	2831	2605	8.0%
	30 min hub	109	111	-1.8%
	60 min hub	1664	1567	5.8%
100%	75 min hub	2409	2036	15.5%
	60 min hub, 200 min spoke	2271	1897	16.5%

Nemhauser and Park [87] provide a polyhedral approach of the edge coloring problem that relies on cutting planes and column generation, which is perhaps the first known work to use these two methods simultaneously (Lee and Leung [76] provide an alternative approach in later work). Let A be the edge-matching incidence matrix representing G. That is, if $a_{ij} = 1$ then edge i is present in matching j, and 0 otherwise. Suppose A is $m \times n$ where $m = |E|$ denotes the number of edges of G and n denote the number of matchings on G. The edge coloring problem has a simple integer programming formulation. Using this notation the edge coloring problem yields the following simple integer programming formulation

$$\chi(G) = \min\left\{\mathbf{1}x : Ax \geq \mathbf{1}, x \in \{0,1\}^n\right\} \tag{43}$$

where $x_j = 1$ if matching j is used in the optimal coloring and 0 otherwise.

The LP-Relaxation of (43) is known as the fractional edge coloring problem

$$\chi_{\mathrm{LP}}(G) = \min\left\{\mathbf{1}x : Ax \geq \mathbf{1}, x \in \mathbb{R}_+^n\right\}. \tag{44}$$

The authors give the correspondence between $\chi_{\mathrm{LP}}(G)$ and $\chi(G)$ seen in Proposition 4.4.8.

Proposition 4.4.8 (Nemhauser and Park). *If $\chi_{LP}(G) > \Delta(G)$, then $\chi(G) = \Delta(G) + 1$, and if $\chi_{LP}(G) = \Delta(G)$ and there is an integral optimal solution to (44), then $\chi(G) = \Delta(G)$.*

Notice that the preceding result is indeterminate when $\chi_{\mathrm{LP}}(G) = \Delta(G)$ but there is a fractional component to the optimal solution. Therefore the authors suggest tightening (44) by adding valid inequalities to the convex hull of integer solutions to (43). The following is a result of Seymour [103] and Stahl [110].

Theorem 4.4.9 (Seymour and Stahl). *Let $U \subseteq V$ and $E(U) = \{(i,j) \in E : i, j \in U\}$. The following inequalities are valid for the convex hull of integer solutions to (43)*

$$\sum_{\{j:M_j \cap E' \neq \emptyset\}} x_j \geq \left\lceil \frac{|E'|}{|U|/2} \right\rceil \quad \forall U \subseteq V \text{ and } E' \subseteq E(U) \tag{45}$$

where x_j is the variable corresponding to the maximal cardinality matching M_j.

For cubic graphs, the right-hand side of (45) is 3. Then the application to our particular case of 3-regular graphs give the following family of *odd circuit inequalities*

$$\sum_{\{j:M_j \cap C \neq \emptyset\}} x_j \geq 3 \quad \text{for all odd circuits } C \tag{46}$$

Combining the preceding valid inequalities with the LP (44) gives rise to the following *augmented fractional edge coloring problem*

$$\chi_{\mathrm{ALP}}(G) = \min \left\{ \mathbf{1}x : Ax \geq \mathbf{1}, Cx \geq \mathbf{3}, x \in \mathbb{R}^n_+ \right\} \tag{47}$$

where C is the edge-odd circuit incidence matrix where $c_{ij} = 1$ if edge i is contained in circuit j and 0 otherwise. The correspondence between $\chi_{\mathrm{ALP}}(G)$ and $\chi(G)$ is that if G is 3-regular and $\chi(G) = 4$, then $\chi_{\mathrm{ALP}}(G) > 3$.

Because characterizing all matchings (columns) and odd-circuits (cuts) explicitly may be intractable, a procedure is needed to handle these dynamically. The authors propose a row and column generation to solve (47) for cubic graphs. If at a given iterate $\chi_{\mathrm{ALP}}(G) = 3$ and the solution is fractional (clearly if the solution is integral, then $\chi(G) = 3$), a simple separation procedure is conducted that removes one of the maximum cardinality matchings with positive weight and checking whether the resulting subgraph contains any odd cycles. If

$\chi_{\text{ALP}}(G) > 3$, then columns are generated by solving a maximum-weight matching problem. If a column prices out, the ALP is re-solved. If no column prices out, then $\chi(G) = 4$.

The same problem is solved through a different procedure that attempts to reduce the total number of iterations to attain an optimal solution relative to that of Nemhauser and Park. Because Benders' decomposition is not employed, the framework introduced in Section 4.4 does not exactly apply. With the case of dynamically generating cuts through separation, the odd-circuit inequalities are always valid over the convex hull of integer solutions to the IP (43). However an analogous problem can be solved which relies on another variant of Theorem 4.4.1, the Integer Farkas' Lemma (see Schrijver [102]).

Theorem 4.4.10 (Integer Theorem of the Alternative). *Given* $(A, b) \in \mathbb{Q}^{m \times n} \times \mathbb{Q}^m$, *exactly one of the following two systems has a solution:*

(i) $\quad \{x \in \mathbb{Z}_+^n : Ax = b\}$

(ii) $\quad \{\pi \in \mathbb{R}^m : \pi b \notin \mathbb{Z}, \pi A \in \mathbb{Z}^m\}$

Now consider the following system which is the feasibility of ALP (47) along with an auxiliary equation

$$\sum_{j \in J} \mathbf{1}_{j,e} x_j \geq 1 \quad \forall e \in E$$

$$\sum_{\{J : M_j \cap c \neq \emptyset\}} x_j \geq 3 \quad \forall c \in C \qquad (F(J, C))$$

$$\sum_{j \in J} x_j = 3$$

where $\mathbf{1}_{j,e}$ is an indicator variable that equals 1 if matching j contains edge e, and 0 otherwise. If $(F(J, C))$ has no integer solution, then $\chi(G) = 4$. This occurs if and only if there exists $(\alpha, \beta, \gamma) \in \mathbb{R}^{|E|} \times \mathbb{R}^{|C|} \times \mathbb{R}$ to the following alternative system for $(F(J, C))$:

$$\sum_{e \in E} \alpha_e + 3 \sum_{c \in C} \beta_c + 3\gamma \not\equiv 0 \pmod{1}$$

$$(\overline{F}(J, C))$$

$$\sum_{e \in j} \alpha_e + \sum_{\{c : M_j \cap c \neq \emptyset\}} \beta_c + \gamma \equiv 0 \pmod{1} \quad \forall j \in J.$$

Suppose $\chi(G) > 3$ for some $J' \subset J$ and $C' \subseteq C$. Then $F(J', C')$ has no integer solution, and thus there exists some α, β, γ for $(\overline{F}(J', C'))$.

Similar to the strategy developed in Section 4.4.1 an algorithm is developed that takes a solution to $F(J', C')$ (where $J' \subset J$ and $C' \subseteq C$), prices out new columns, and from the pricing solution attempts to populate a solution to $(\overline{F}(J, C))$.

The pricing problem associated with a new column $j \in J \setminus J'$ is a new matching. Let $\{\pi_e\}_{e \in E}$ and $\{\rho_c\}_{c \in C}$ denote dual variables from the ALP. Given the reduced cost $\bar{c}_j = 1 - \sum_{e \in j} \pi_e - \sum_{c:M_j \cap c \neq \emptyset} \rho_c$ the pricing problem amounts to the following problem

$$\max \quad \sum_{e \in E} \pi_e x_e + \sum_{c \in C'} \rho_c y_c \tag{48}$$

$$\text{s.t.} \quad \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V \tag{49}$$

$$\sum_{e \in U(S)} \leq \frac{1}{2}(|S| - 1) \quad \forall S \subseteq V : \; S \text{ an odd set} \tag{50}$$

$$y_c - \sum_{e \in C'} x_e \leq 0 \quad \forall c \in C' \tag{51}$$

$$x_e \in \{0, 1\} \quad \forall e \in E$$

$$y_c \in \{0, 1\} \quad \forall c \in C'.$$

If $C' = \emptyset$ the pricing problem amounts to a maximum-weight matching problem (see Nemhauser and Wolsely [88]). Otherwise it is similar to the max-weight matching problem with some differences. The objective (48) seeks to find a maximum-weight of the matching and circuit over G given (π, ρ). As a matching can contain no more than one edge incident to every vertex, constraints (49) are added. (50) are referred to as *blossom inequalities*. In the left-hand side $U(S) \equiv \{(i, j) \in E : i, j \in S\}$ for all $S \subseteq V$. An odd set is any subset S of V whose cardinality is an odd integer. Constraints (51) are needed to ensure at least one edge is included in the matching from a given circuit.

$\chi_{\text{ALP}}(G) \notin \mathbb{Z}$ and no column prices out then $\chi(G) = 4$. However, if a column does price out then the following result provides a certificate that shows the existence of a solution to $(\overline{F}(J, C))$ if a set of conditions are satisfied.

Theorem 4.4.11. *Suppose $\chi_{ALP}(G) \notin \mathbb{Z}$ over some set of matchings $J' \subset J$. Let (π, ρ) denote dual variables associated with two constraint classes in (47), respectively. Let J^{new} denote all matchings generated from the subproblem whose reduced cost is strictly negative. That is,*

$$J^{new} = \bigcup \left\{ j \in J \setminus J' : \sum_{e \in j} \pi_e + \sum_{\{c : M_j \cap c \neq \emptyset\}} \rho_c > 1 \right\}.$$

Assume that $J^{new} \neq \emptyset$. If $\exists p_1, p_2 \in \mathbb{Z}_+$ and $q \in \mathbb{Z}_{++}$ such that following three conditions hold, then $\chi(G) = 4$.

(i) $\bar{c}_j \equiv \frac{p_1}{q}$ *(mod 1)* $\forall j \in J' : \bar{c}_j \notin \mathbb{Z}$

(ii) $\bar{c}_j \equiv \frac{p_2}{q}$ *(mod 1)* $\forall j \in J^{new} : \bar{c}_j \notin \mathbb{Z}$

(iii) $q \cdot \left(\sum_{e \in E} \pi_e + 3 \sum_{c \in C'} \rho_c \right) \not\equiv 0$ *(mod 1)*

Proof. Suppose that conditions (i) through (iii) hold. By (i) and (ii)

$$1 - \sum_{e \in j} \pi_e - \sum_{\{c : c \cap M_j \neq \emptyset\}} \rho_c \begin{cases} \equiv \frac{p_1}{q} \ (\text{ mod } 1) & \forall j \in J' : \bar{c}_j \notin \mathbb{Z} \\[2mm] \equiv \frac{p_2}{q} \ (\text{ mod } 1) & \forall j \in J^{\text{new}} : \bar{c}_j \notin \mathbb{Z} \\[2mm] \equiv 0 \ (\text{mod } 1) & \forall j \in J' \cup J^{\text{new}} : \bar{c}_j \in \mathbb{Z}. \end{cases}$$

By setting $\widehat{\pi}_e = \pi_e \cdot q$ for all $e \in E$ and $\widehat{\rho}_c = \rho_c \cdot q$ then

$$1 - \sum_{e \in j} \widehat{\pi}_e - \sum_{\{c : M_j \cap c \neq \emptyset\}} \widehat{\rho}_c \in \mathbb{Z} \quad \forall j \in J' \cup J^{\text{new}} : \bar{c}_j \notin \mathbb{Z} \tag{52}$$

and for those $j \in J' \cup J^{\text{new}}$ such that $\bar{c}_j$ is integral, then the left hand side of (52) remains integral as $q \in \mathbb{Z}$. Therefore $1 - \sum_{e \in j} \widehat{\pi}_e - \sum_{\{c : M_j \cap c \neq \emptyset\}} \widehat{\rho}_c \equiv 0$ (mod 1) for all $j \in J' \cup J^{\text{new}}$.

Condition (iii) holds if and only if $-q \left(\sum_{e \in E} \pi_e + 3 \sum_{c \in C'} \rho_c \right) \not\equiv 0$ (mod 1). Thus, $1 - q \left(\sum_{e \in E} \pi_e + 3 \sum_{c \in C'} \rho_c \right) \not\equiv 0$ (mod 1). Therefore setting $(-\widehat{\pi}, -\widehat{\rho}, 1)$ is a solution to $\overline{F}(J' \cup J^{\text{new}}, C')$.

Since J^{new} denotes *all* new matchings whose reduced cost is strictly negative, it suffices to show a solution exists over all $J \cup J^{\text{new}}$. Finally, extending the solution to all of C is trivial by setting $\rho_c = 0$ for all $c \in C \setminus C'$. $\square$

We now examine our results on a series of edge coloring problems. For cubic graphs whose chromatic index is 3, the two algorithms are identical. Therefore compare the algorithm as given by Nemhauser and Park to our modified algorithm (SRGC) on *snarks*, which

are cubic graphs whose chromatic index is 4. Table 15 summarizes the results. We note that on average 22.3% fewer calls to the ALP are required for the modified algorithm.

Table 15: Benchmark results: Edge Coloring

		Number of iterations required		
Graph	$|V|$	**Nemhauser & Park**	**SRCG**	**Improvement (%)**
Petersen	10	7	5	28.6
Double Star	30	66	37	43.9
Flower Snark	12	24	19	20.8
Flower Snark	20	43	39	9.3
Flower Snark	28	65	47	27.7
Flower Snark	36	73	61	16.4
Flower Snark	44	89	75	15.7
Watkins Snark	50	131	110	16.0

While the computational results indicate fewer iterations required to obtain the optimal solution, the runtimes for the SRGC method were substantially higher. This is due to the fact that *all* columns that price out negatively are required to be found. While modern commercial solvers support such functionality, these routines may take considerable time for large pool sizes (all computations used CPLEX 12.2).

CHAPTER V

ACCELERATING BENDERS DECOMPOSITION

Benders' decomposition is unequivocally one of the most widely-used algorithms in mathematical programming and is used in a host of applications. In spite of its ubiquity, the algorithm is known to often exhibit slow convergence precluding its use in some applications. It is therefore of natural interest to study if enhancements to the standard algorithm exist that lead to improved convergence. How the algorithm may be expedited is a straightforward, pragmatic, question that is not particularly well-known throughout the literature. A survey of some recent studies that have examined accelerating the algorithm through various techniques is first discussed. This chapter presents a different approach by taking a standard cut and strengthening it to improve the efficacy of the procedure. Four possible cut-strengthening procedures are introduced in a general context before examining their use on the Airline Integrated Recovery (AIR) problem as formulated in Chapter 3.

The reader is referred to Section 4.3 of Chapter 4 for a review of Benders' Decomposition.

For the remainder of this chapter, the following notation is introduced. Denote a generic Benders' cut by

$$\psi x \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta \tag{53}$$

where ψ_0 is a constant term derived by the subproblem constraints whose right-hand sides do not depend on linking variables, $\mathbf{1}^{\text{opt}}$ is an indicator variable determining whether the cut is an optimial cut (equal to unity) or a feasibility cut (equal to zero), and

$$\psi = \begin{cases} \pi & \text{if optimality cut} \\ r & \text{if feasibility cut.} \end{cases}$$

This chapter seeks to expedite Benders' decomposition by returning a cut that dominates (53).

5.1 Literature Review of Benders' Decomposition

A number of studies have examined extensions to the seminal work of Benders [26]. Geoffrion [57] showed how Benders' algorithm may be employed to a problem structure in which the subproblems are nonlinear programming problems, and the cutting planes are generated by nonlinear convex duality theory. The generalized procedure is shown to converge when the space of master variables is either a finite discrete set or a compact and convex set of infinite cardinality.

Because the addition of Benders' cuts introduce complexity to the Restricted Master Problem (RMP), finding ways to expedite solving the RMP have been proposed. McDaniel and Devine [83] propose solving the RMP as a linear relaxation as cuts can be generated from any extreme point or extreme ray of the dual subproblem polyhedron. Côté and Laughton [46] show that valid cuts can be added for any integer solution (which may be suboptimal) from the RMP. This can be done through a simple heuristic, although the convergence property of Benders' algorithm is no longer valid. Holmberg [65] proposed using Lagrangean relaxation to overcome possible difficulties in solving the RMP, although he showed that the lower bound delivered from this approach is dominated by using Lagrangean relaxation to the original MP.

Other studies have sought to accelerate the original framework. This chapter is most closely related to these studies. For an original MIP problem, Magnanti and Wong [80] suggest accelerating Benders' original algorithm by exploiting the 'best' possible cut when there are multiple solutions from the dual subproblem. This is accomplished by evaluating each dual solution at an arbitrary point from the relative interior of the convex hull of integer solutions from the feasible set of the RMP (referred to as a *core point*). The cut associated with the maximum value is then added and is shown to be Pareto-optimal (or equivalently, it dominates every other candidate cut). They also discuss the profound importance of model formulation of the Full Master Problem (FMP) in the efficacy of the decomposition scheme; in a certain class of facility location problems, for instance, they show that strong model formulations require only a single Benders' cut for convergence.

Rei et al. [94] use a concept of local branching, as introduced by Fischetti and Lodi

[51], to accelerate the convergence of Benders' algorithm for a 0-1 MIP. The authors show how both the lower bound and upper bounds can be improved by the addition of local branching constraints from a partition of the original feasible set. They show the value in such an approach on both a deterministic multicommodity capacitated fixed-charge network design problem and a stochastic integrated model for logistics network design. In both instances they have shown that local branching can substantially enhance the performance of a traditional Benders' implementation.

Fischetti et al. [52] propose a new selection rule of Benders' cuts. For a given solution to the restricted master problem (x^*, η^*) the authors examine the polyhedron Ω consisting of the feasible region of a given subproblem along with an auxiliary inequality that conjectures the (x^*, η^*) is suboptimal. Note that this system is empty if and only if a Benders cut is added to the master problem in the subsequent iteration. By defining an arbitrary objective function measuring the magnitude of a potential infeasibility and optimizing over Ω, a Benders cut is added if the corresponding optimization problem is empty, or equivalently if the dual problem is unbounded (it is assumed the dual problem has a feasible solution). The dual optimization problem induces the cut generating linear program (CGLP) that is similar to disjunctive programming studies by, among others, Balas et al. [13].

Most of these expositions studying possible acceleration mechanisms to Benders' algorithm belong to one of two classes. The first seeks ways to increase the convergence by reducing the number of iterations necessary to obtain a solution. The second involves ways to reduce the time spent by obtaining successive solutions to the RMP which we have remarked may be increasingly difficult as more Benders' cuts are added. Within the first class of studies, acceleration techniques may involve novel modeling strategies, or methods to find stronger cuts in the presence of several cut candidates. While the efficacy of cuts may be influenced by such techniques, most of the *structure* of the original Benders' cuts remains the same. The purpose of this section is to explore the performance of the traditional Benders' technique by strengthening the structure of a standard inequality to induce deeper cuts. We study four alternatives to expediting the algorithm. The techniques are all derived from the theory of integer programming and are first reviewed. Then their

application to a generic MIP exhibiting a block structure are studied. Finally, each method is performed on the Airline Integrated Recovery model studied in Chapter 3 where it is shown that acceleration is possible.

5.2 Strengthening by Cut-Pushing

Consider a mixed integer programming problem of the form

$$\min \left\{ cx + fy \quad \text{s.t.} \quad Ax = b, Bx + Fy = d, (x, y) \in \mathbb{Z}_+^n \times \mathbb{R}_+^p \right\}. \tag{P}$$

where the presence of a single subproblem is assumed without loss of generality. Using the notation in Section 4.3 the RMP is

$$
\begin{aligned}
\min \quad & cx + \eta \\
\text{s.t.} \quad & Ax = b \\
& \pi_p \left(d - Bx \right) \leq \eta \quad \forall p \in \widetilde{\Pi} \qquad (RMP(\widetilde{\Pi}, \widetilde{R})) \\
& r_q \left(d - Bx \right) \leq 0 \quad \forall q \in \widetilde{R} \\
& (x, \eta) \in \mathbb{Z}_+^n \times \mathbb{R}_+.
\end{aligned}
$$

For notational convenience let

$$
S\left(\widetilde{\Pi}, \widetilde{R}\right) = \left\{ (x, \eta) \in \mathbb{Z}_+^n \times \mathbb{R} : \begin{array}{l} Ax = b, \\ \pi_p \left(d - Bx \right) \leq \eta \quad \forall p \in \Pi' \\ r_q \left(d - Bx \right) \leq 0 \quad \forall q \in R' \end{array} \right\}
$$

denote the feasible set of $(RMP(\widetilde{\Pi}, \widetilde{R}))$ given the set of dual extreme points $\widetilde{\Pi} \subseteq \Pi$ and dual extreme rays $\widetilde{R} \subseteq R$.

Recall the generic Benders' cut (53): $\psi x \leq \psi_0 + \mathbf{1}^{\mathrm{opt}} \eta$. The cut may or may not be binding at the incumbent point, which would depend on both whether the cut is an optimality or feasibility cut, and whether the solution from the RMP is integral (recall that a cut may be generated for any solution to the RMP). One can determine the minimum distance the cut can be shifted to the interior of $S\left(\widetilde{\Pi}, \widetilde{R}\right)$ by solving the following auxiliary MIP:

$$
\begin{aligned}
\Delta^* = \quad &\min \quad \Delta \\
&\text{s.t.} \quad Ax = b \\
&\qquad \pi_p \left(d - Bx \right) \le \eta \quad \forall p \in \widetilde{\Pi} \\
&\qquad r_q \left(d - Bx \right) \le 0 \quad \forall q \in \widetilde{R} \\
&\qquad \psi x = \psi_0 + \mathbf{1}^{\mathrm{opt}}\eta - \Delta \\
&\qquad (x, \eta, \Delta) \in \mathbb{Z}^n_+ \times \mathbb{R}_+ \times \mathbb{R}_+.
\end{aligned}
\tag{54}
$$

Let $H(x; t, t_0) \equiv \{x \in \mathbb{Z}^n : tx \le t_0\}$ denote the set of integer points in the halfspace defined for some $(t, t_0) \in \mathbb{R}^n \times \mathbb{R}$. Define

$$
S' = S\left(\widetilde{\Pi}, \widetilde{R} \right) \bigcap H\left(x; \psi, \psi_0 - \Delta^* + \mathbf{1}^{\mathrm{opt}}\eta \right).
\tag{55}
$$

as the new feasible set for the RMP which includes the new Benders' cut (53).

If $\Delta^* = 0$ then there exists a point $x^* \in S'$ such that the cut is binding at x^*. If $\Delta^* > 0$, then the cut is not binding at an integer point in S', and the inequality

$$
\psi x \le \psi_0 + \mathbf{1}^{\mathrm{opt}}\eta - \Delta^*
\tag{56}
$$

remains valid for S' and binding at some $x^* \in S'$. Augmenting the original cut (53) by an affine transformation (56) amounts to pushing the cut further into the interior of $conv(S')$.

Figure 22 illustrates the concept of the strengthened cut.

Definition 5.2.1 (Dominating Cut). *An inequality $\pi x \ge \pi_0$ is said to dominate another inequality $\pi' x \ge \pi_0$ if $\pi x \ge \pi' x$ for all x.*

Lemma 5.2.2. *If $\Delta^* > 0$ is chosen from (54) then the cut $\psi x \le \psi_0 + \mathbf{1}^{opt}\eta - \Delta^*$ dominates $\psi x \le \psi_0 + \mathbf{1}^{opt}\eta$.*

Proof. Note that $\psi x \le \psi_0 + \mathbf{1}^{\mathrm{opt}}\eta - \Delta^*$ holds if and only if $\psi x + \Delta^* \le \psi_0 + \mathbf{1}^{\mathrm{opt}}\eta$. The result follows since $\psi x + \Delta^* \ge \psi x$. $\qquad\square$
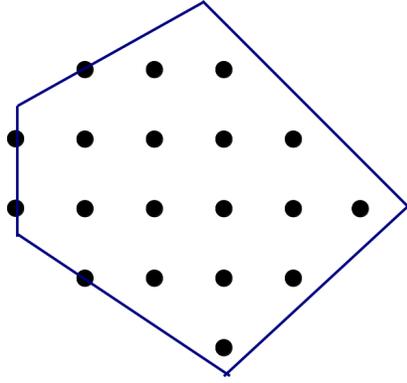
(a) Feasible set $S\left(\widetilde{\Pi}, \widetilde{R}\right)$
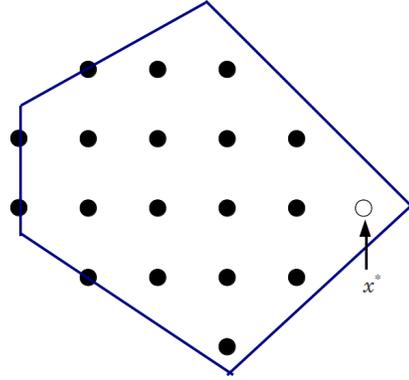
(b) Solution $x \in S\left(\widetilde{\Pi}, \widetilde{R}\right)$ from RMP
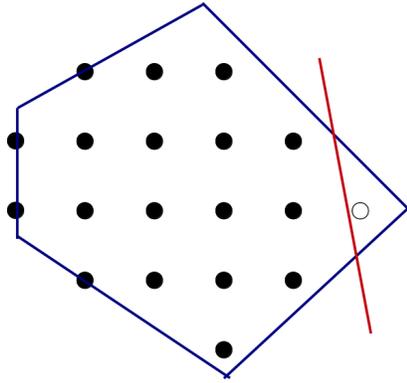
(c) Generated Benders' cut $\psi x \leq \psi_0 + \mathbf{1}^{\text{opt}}\eta$
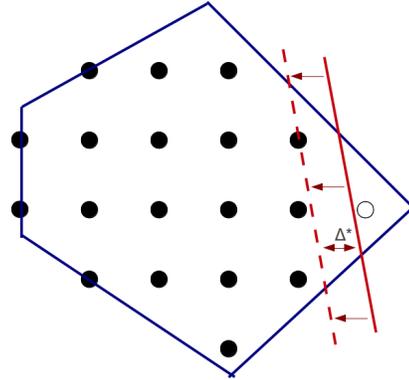
(d) Augmented Cut $\psi x \leq \psi_0 + \mathbf{1}^{\text{opt}}\eta - \Delta^*$ dominates original cut

Figure 22: Strengthening Benders' Cut by Pushing

An illustration of the cut-pushing method is seen in the following example.

Example 5.2.3. *Consider the MIP*

$$
\begin{array}{rlllllll}
\max & x_1 & + & 3x_2 & - & y_1 & + & 4y_2 \\
\text{s.t.} & x_1 & + & x_2 & & & & \le & 20 \\
& 2x_1 & + & x_2 & & & & \le & 30 \\
& -3x_1 & + & 2x_2 & & & & \le & 20 \\
& 8x_1 & & & & & & \le & 87 \\
& x_1 & + & x_2 & + & y_1 & + & y_2 & \le & 25 \\
& -3x_1 & - & x_2 & - & 3y_1 & + & 2y_2 & \le & -58 \\
& & & & & (x,y) & \in & \mathbb{Z}_+^2 \times \mathbb{R}_+^2.
\end{array}
$$

Given $\widetilde{\Pi} = \emptyset$, $\widetilde{R} = \emptyset$ *the initial RMP is*

$$
RMP\,(\emptyset,\emptyset) = \max \left\{ x_1 + 3x_2 : \begin{array}{rcrl}
x_1 & + & x_2 & \le 20 \\
2x_1 & + & x_2 & \le 30 \\
-3x_1 & + & 2x_2 & \le 20 \\
8x_1 & & & \le 87 \\
& x \in \mathbb{Z}_+^2 &
\end{array} \right\}
$$

whose solution is $x^* = (4,16)$. *The dual subproblem*

$\min\left\{ 5q_1 - 30q_2 : q_1 - 3q_2 \ge 1, q_1 + 2q_2 \ge 4, q \in \mathbb{R}_+^2 \right\}$ *is unbounded with extreme ray* $\left(1, \frac{1}{3}\right)$ *leading to the Benders' cut*

$$
2x_2 \le 17. \tag{57}
$$

The following auxiliary problem is the cut-shifting subproblem as described in Section 5.6.1:

$$
\begin{array}{rlll}
\Delta^* & = & \min & \Delta \\
\text{s.t.} & x_1 + x_2 & \le 20 \\
& 2x_1 + x_2 & \le 30 \\
& -3x_1 + 2x_2 & \le 20 \\
& 8x_1 & \le 87 \\
& 2x_2 + \Delta & = 17 \\
& (x,\Delta) \in \mathbb{Z}_+^2 \times \mathbb{R}_+
\end{array}
$$

whose optimal objective value is $\Delta^* = 1$ leading to the strengthened Benders' cut

$$2x_2 \leq 16.$$

Lemma 5.2.4. *Consider a MIP of the form (P) and suppose that Benders' decomposition is applied by solving the RMP as a MIP. Then for any solution to the RMP that induces an optimality cut, the optimal value of (54) is 0.*

Proof. Let (x^*, η^*) be a solution to the RMP. Suppose π is a dual feasible solution that induces an optimality cut of the form $\pi x \leq \pi_0 + \eta$. Then the optimality cut is binding at the point (x^*, η^0) where η^0 is the value of the subproblem evaluated at x^*. □

Recall that there are variants of Benders' decomposition other than that shown in Algorithm 2 from Chapter 4. One is by solving the continuous relaxation of the RMP for some iterations. The following result shows that the optimum value of (54) may be nonzero for such cases.

Lemma 5.2.5. *Consider a MIP of the form (P) and suppose that Benders' decomposition is applied by solving the linear relaxation for some iteration in solving the RMP. Then for any solution (x^0, η^0) to the RMP for which $x^0 \notin \mathbb{Z}_+^n$ that induces a feasibility cut, the optimal value of (54) may be strictly positive.*

The optimal objective of (54) may be nonzero for feasibility cuts, however, even if the RMP is solved as a MIP.

Lemma 5.2.6. *Consider a MIP of the form (P) and suppose that Benders' decomposition is applied. If for some iteration the solution to the RMP induces a feasibility cut, then the optimal value of (54) may be strictly positive.*

Proof. See Example 5.2.3. □

Pushing cuts into the interior of a feasible set to strengthen the linear relaxation has been explored in other contexts. Bowman and Nemhauser [29] study how to strengthen cuts from a tableau to make deeper cuts. They show that cuts from a tableau are deepest in the sense they are not dominated by cuts further into the interior. Goycoolea [60] proposed a

similar concept in the context of solving a quadratic mixed-integer programming problem. He proposed first solving the quadratic function over the linear relaxation. If the solution was not integral, a binding constraint may be pushed into the interior of the feasible set until it was binding at an integer solution. The so-called 'tangent cuts' were shown to be effective in a number of applications. Smith [106] studied the Origin-Destination Fleet Assignment Model (ODFAM), where Benders' cuts may be used to approximate a nonlinear revenue function. The cuts are tangent to the original concave function and overestimate the original feasible set. He observed that convergence could potentially be accelerated by pushing the Benders' inequalities down into the interior of the feasible set.

Recall the definition of S', shown in (55), denotes the new feasible set for the RMP with the addition of the new Benders' cut. The following results provide insight into the depth of the strengthened cut (56).

By choice of Δ^*, the support of the convex hull of the new feasible set S' can be partially characterized as follows.

Lemma 5.2.7. *Given $\widetilde{\Pi} \subseteq \Pi$ and $\widetilde{R} \subseteq R$, if Δ^* is chosen from (54), the strengthened inequality $\psi x \leq \psi_0 - \Delta^* + \mathbf{1}^{opt}\eta$ is a face of $conv\,(S')$ provided that the feasible set is nonempty.*

Proof. The result follows since there is at least one integer point in the set that is binding at the strengthened inequality which defines at least a 0-dimensional face of $conv\,(S')$. $\qquad\square$

Theorem 5.2.8. *Given $\widetilde{\Pi} \subseteq \Pi$ and $\widetilde{R} \subseteq R$, if Δ^* is chosen from (54), the strengthened inequality $\psi x \leq \psi_0 - \Delta^* + \mathbf{1}^{opt}$ is not necessarily a facet of $conv\,(S')$.*

Proof. Let

$$T = \left\{ (x,y) \in \mathbb{Z}_+^2 \times \mathbb{R}_+^2 : \begin{array}{c} 3x_1 + 2x_2 \leq 10 \\ 2x_1 + y_1 + y_2 \leq 10 \\ -x_2 - y_1 + y_2 \leq -2 \end{array} \right\}$$

and consider the problem $\max\,\{2x_1 + x_2 - y_1 + y_2 : (x,y) \in T\}$ which is solved by Benders' decomposition and solves the RMP as a MIP with linking variables x and subproblem

variables y. With $\widetilde{\Pi} = \emptyset$ and $\widetilde{R} = \emptyset$ the initial solution is $x^* = (3,0)$ which leads to an infeasible subproblem whose feasibility cut is $2x_1 - x_2 \leq 5$. The cut is not binding at any integer points from $T(\emptyset, \emptyset)$ and it can be shown that $\Delta^* = 1$ leading to the strengthened cut $2x_1 - x_2 \leq 4$. Facets of $conv(S')$ are then seen to be precisely $x_1 \geq 0$, $x_2 \geq 0$, $x_1 \leq 2$, and $3x_1 + 2x_2 \leq 10$, none of which correspond to the strengthened cut. $\qquad \square$

The preceding results suggest when a facet is defined for $conv(S')$.

Corollary 5.2.9. *The strengthened inequality is facet-defining for $conv(S')$ if and only if it is parallel to a facet for $conv(S')$.*

The summary of the cut-pushing algorithm is given in Algorithm 5.

Algorithm 5 Cut-Pushing Algorithm

1: **given:** optimal solution $x^* \in S(\widetilde{\Pi}, \widetilde{R})$ to RMP and Benders' cut

$$\psi x \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta$$

2: **if** feasibility cut **then**
3:     let $r \in R \setminus \widetilde{R}$ be the dual extreme ray that generated cut $rx \leq r_0$
4:     let Δ^* be optimal objective value to (54)
5:     $\widetilde{R} \leftarrow \widetilde{R} \cup \{r\}$
6: **else**
7:     let $\pi \in \Pi \setminus \widetilde{\Pi}$ be dual extreme point that generated the cut $\pi x \leq \pi_0 + \eta$
8:     **if** RMP was solved as continuous relaxation and $x \notin \mathbb{Z}_+^n$ **then**
9:         let Δ^* be optimal objective value to (54)
10:     **else**
11:         set $\Delta^* = 0$
12:     **end if**
13:     $\widetilde{\Pi} \leftarrow \widetilde{\Pi} \cup \{\pi\}$
14: **end if**
15: return $\psi x \leq \psi_0 - \Delta^* + \mathbf{1}^{\text{opt}}$

5.3 Strengthening by Benders-Induced Split Cuts

Cutting planes have long been used to solve mixed-integer programming (MIP) problems. Consider a linear relaxation of a given MIP for which its solution contains some component that violates an integrality restriction. A *cut* is an inequality that is valid for the convex hull of the feasible set but is violated by some fractional extreme point. Some cutting plane algorithms such as the Chvátal-Gomory cut for a pure integer program (IP), terminate after a finite number of iterations. While convergence is not guaranteed for more general problems, cutting planes still have the ability to expedite computation during branch-and-cut. Cutting planes are of great importance to the efficacy of modern commercial solvers.

Early research on cutting planes relied largely on the structure of the simplex tableau attained from a continuous relaxation of the original MIP. Recently strong cutting planes are thought to be driven by the underlying polyhedral structure of the problem (see, Conforti et al. [42] and Atamtürk [11] for surveys of such methods). This section studies the use of one well-known family of polyhedral cuts known as *split cuts* to be used together with Benders' decomposition.

5.3.1 Review of Split Cuts

Split cuts are a special case of a broader family of disjunctive cuts. This class of cutting planes was studied by Cook et al. [44] where it was shown that the split closure of a mixed integer set is a polyhedron. Consider a MIP problem of the form

$$\max \quad \{cx : x \in S\}$$

where $S \equiv \left\{Ax \leq b, x_j \in \mathbb{Z}, \forall j \in N_0, x \in \mathbb{R}_+^n\right\}$, A is an $m \times n$ matrix, $N = \{1, 2, \ldots, n\}$ is the index set of variables and $N_0 \subseteq N$ is the set of integer variables. Assume that for all binary variables $x_j, j \in N_0$ that the upper bound constraints $x_j \leq 1$ are present in the constraint set (A, b).

Given some $(\pi, \pi_0) \in \mathbb{Z}^n \times \mathbb{Z}$ such that $\pi_j = 0 \quad \forall j \in N \setminus N_0$ note that $\{x \in \mathbb{Z}^n : \pi_0 < \pi x < \pi_0 + 1\} = \emptyset$. Therefore every point in the feasible set must belong to exactly one region from the following disjunction:

116

$$
\underbrace{\begin{pmatrix} Ax \le b \\ \pi x \le \pi_0 \\ x \ge 0 \end{pmatrix}}_{\equiv \Pi_0} \quad \bigvee \quad \underbrace{\begin{pmatrix} Ax \le b \\ -\pi x \le -(\pi_0 + 1) \\ x \ge 0 \end{pmatrix}}_{\equiv \Pi_1}.
$$

The values (π, π_0) define a *split*. One class of polyhedral cuts seek to find a valid inequality of the form $\alpha x \le \alpha_0$ that are constructed from a given split that are valid for $conv\,(\Pi_0 \cup \Pi_1)$. Such an inequality is referred to as a *split cut*, and is illustrated in Figure 23.
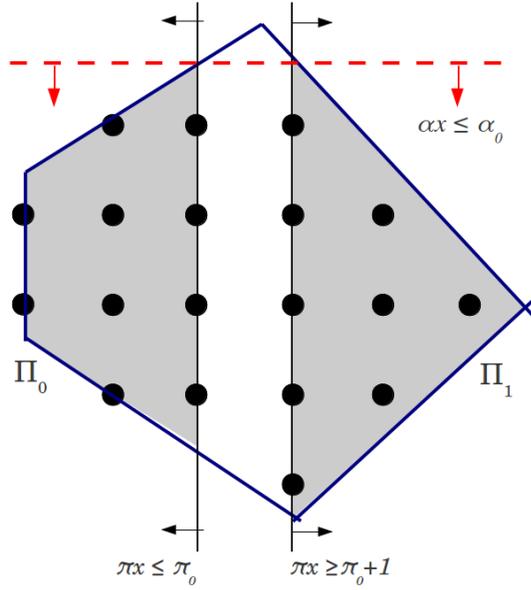
Figure 23: An Example of a Split Cut

A split cut $\alpha x \le \alpha_0$ is derived from a split (π, π_0). Since the inequality $\alpha_0 \ge \alpha x$ is required to be valid for $\Pi_0 \cup \Pi_1$,

$$
\alpha_0 \ge \max \{\alpha x : x \in \Pi_0\}
$$

$$
= \max \{\alpha x : Ax \le b, \pi x \le \pi_0, x \ge 0\}
$$

$$
= \min \{ub + u_0 \pi_0 : uA + u_0 \pi \ge \alpha, (u, u_0) \in \mathbb{R}_+^m \times \mathbb{R}_+\}
$$

where the latter relation holds from duality. Similarly,

$$\alpha_0 \geq \max\{\alpha x : x \in \Pi_1\}$$

$$= \max\{\alpha x : Ax \leq b, -\pi x \leq -(\pi_0 + 1), x \geq 0\}$$

$$= \min\{vb - v_0(\pi_0 + 1) : vA - v_0\pi \geq \alpha, (v, v_0) \in \mathbb{R}_+^m \times \mathbb{R}_+\}.$$

Given an incumbent solution $\bar{x}$, and a split (π, π_0), coefficients of the split cut can therefore be determined by solving the following problem referred to as the Cut Generating Linear Program (CGLP)

$$
\begin{aligned}
\max \quad & \alpha\bar{x} - \alpha_0 \\
\text{s.t.} \quad & \alpha_0 \geq ub + u_0\pi_0 \\
& \alpha_0 \geq vb - v_0(\pi_0 + 1) \\
& \alpha \leq uA + u_0\pi \\
& \alpha \leq vA - v_0\pi \\
& \mathbf{1}u + \mathbf{1}v + u_0 + v_0 = 1 \\
& u, v, u_0, v_0 \geq 0
\end{aligned}
\tag{58}
$$

where $\mathbf{1}$ is an m-dimensional vector of ones, and $\mathbf{1}u + \mathbf{1}v + u_0 + v_0$ is a normalization constraint. If the optimal objective is strictly positive, then the split cut separates $\bar{x}$ and is added to the relaxation. Else, a proof is given that $\bar{x}$ belongs to $\Pi_0 \cup \Pi_1$. In order to ensure a separation, typically one chooses $\overline{x}$ such that $\pi\overline{x} \in (\pi_0 - 1, \pi_0)$.

5.3.2 Split Cuts Generated from a Benders Disjunction

This section describes how split cuts can be generated from a disjunction defined from a Benders' feasibility cut. It is assumed that all variables from the RMP are integral except for continuous variables η present in optimality cuts. Because split cuts are derived from a disjunction that requires zero coefficients on all continuous variables, cut strengthening discussed in this section applies only to feasibility cuts.

Let $S(\widetilde{\Pi}, \widetilde{R})$ denote the mixed-integer feasible set of an RMP where $\widetilde{\Pi}$ and $\widetilde{R}$ denote the dual extreme points and extreme rays that are defined in the feasible set $S(\widetilde{\Pi}, \widetilde{R})$ for the RMP.

Given a dual extreme ray r consider a general Benders' feasibility cut of the form $rx \leq r_0$ and assume that $r \in \mathbb{Z}^n$. Then the optimal solution in the space of RMP variables must be contained in exactly one side of the following disjunction:

$$\{rx = r_0\} \quad \vee \quad \{rx \leq r_0 - 1\}. \tag{59}$$

Split cuts can be generated in a similar manner presented in Section 5.3.1 from the disjunction $\Pi_0 \vee \Pi_1$ where

$$\Pi_0 \equiv X(\widetilde{\Pi}, \widetilde{R}) \cap \{x : rx = r_0\} \quad \text{and}$$

$$\Pi_1 \equiv X(\widetilde{\Pi}, \widetilde{R}) \cap \{x : rx \leq r_0 - 1\}.$$

It is assumed without loss of generality that there is a single subproblem present in a MIP. The CGLP for an RMP whose feasible set $S\left(\widetilde{\Pi}, \widetilde{R}\right)$ is derived as follows.

$$S\left(\widetilde{\Pi}, \widetilde{R}\right) = \left\{ (x, \eta) : \begin{array}{rcll} Ax & \leq & b & \\ -r_q Bx & \leq & -r_q d & \forall q \in \widetilde{R} \\ -\pi_p Bx - \eta & \leq & -\pi_p d & \forall p \in \widetilde{\Pi} \\ (x, \eta) & \in & \mathbb{Z}_+^n \times \mathbb{R}_+ & \end{array} \right\}$$

where A is $m_1 \times n$ and B is $m_2 \times n$.

Let $X\left(\widetilde{\Pi}, \widetilde{R}\right)$ be the continuous relaxation of $S\left(\widetilde{\Pi}, \widetilde{R}\right)$ and consider the disjunction from (59) generated by a Benders' feasibility cut. Coefficients $(\alpha, \alpha_0) \in \mathbb{R}^n \times \mathbb{R}$ are defined for a split cut $\alpha x \leq \alpha_0$ that must be valid for $\Pi_0 \cup \Pi_1$. Note that

$$\alpha_0 \geq \max\{\alpha x : x \in \Pi_0\}$$

$$= \max \left\{ \alpha x : \begin{array}{rl} Ax & \leq b \\ -r_q Bx & \leq -r_q d \quad \forall r \in \widetilde{R} \\ -\pi_p Bx - \eta & \leq -\pi_p d \quad \forall p \in \widetilde{\Pi} \\ rx & = r_0 \\ (x, \eta) & \in \mathbb{R}^n_+ \times \mathbb{R}_+ \end{array} \right\}$$

$$= \begin{array}{rl} \min & u^1 b - \sum_{q \in \widetilde{R}} u^2_q r_q d - \sum_{q \in \widetilde{\Pi}} u^3_q \pi_p d + u_0 r_0 \\ \text{s.t.} & u^1 A - \sum_{q \in \widetilde{R}} u^2_q r_q B - \sum_{q \in \widetilde{\Pi}} u^3_q \pi_p B + u_0 r \geq \alpha \\ & \sum_{i=1}^{|\widetilde{\Pi}|} \sum_{j=1}^{m_2} u^3_{i,j} \leq 0 \\ & u^1, u^2, u^3 \geq 0 \\ & u_0 \in \mathbb{R}. \end{array}$$

The latter two constraints imply that $u^3 = 0$. Similarly

$$\alpha_0 \geq \max\{\alpha x : x \in \Pi_1\}$$

$$= \max \left\{ \alpha x : \begin{array}{rl} Ax & \leq b \\ -r_q Bx & \leq -r_q d \quad \forall r \in \widetilde{R} \\ -\pi_p Bx - \eta & \leq -\pi_p d \quad \forall p \in \widetilde{\Pi} \\ rx & \leq r_0 - 1 \\ (x, \eta) & \in \mathbb{R}^n_+ \times \mathbb{R}_+ \end{array} \right\}$$

$$= \begin{array}{rl} \min & v^1 b - \sum_{q \in \widetilde{R}} v^2_q r_q d - \sum_{q \in \widetilde{\Pi}} v^3_q \pi_p d + v_0 (r_0 - 1) \\ \text{s.t.} & v^1 A - \sum_{q \in \widetilde{R}} v^2_q r_q B - \sum_{q \in \widetilde{R}} v^3_q \pi_p B + v_0 r \geq \alpha \\ & \sum_{i=1}^{|\widetilde{\Pi}|} \sum_{j=1}^{m_2} v^3_{i,j} \leq 0 \\ & v^1, v^2, v^3 \geq 0 \\ & v_0 \in \mathbb{R}_+. \end{array}$$

Observing $u^3 = 0$ and $v^3 = 0$ the CGLP associated with the Benders' disjunction is

$$
\begin{aligned}
\max \quad & \alpha \overline{x} - \alpha_0 \\
\text{s.t.} \quad & u^1 b - \sum_{q \in \widetilde{R}} u_q^2 r_q d + u_0 r_0 \leq \alpha_0 \\
& v^1 b - \sum_{q \in \widetilde{R}} v_q^2 r_q d + v_0 \left(r_0 - 1 \right) \leq \alpha_0 \\
& u^1 A - \sum_{q \in \widetilde{R}} u_q^2 r_q B + u_0 r \geq \alpha \\
& v^1 A - \sum_{q \in \widetilde{R}} v_q^2 r_q B + v_0 r \geq \alpha \qquad\qquad (60) \\
& \sum_{i=1}^{m_1} \left(u_i^1 + v_i^1 \right) + \sum_{i=1}^{|\widetilde{\Pi}|} \sum_{j=1}^{m_2} \left(u_{i,j}^2 + v_{i,j}^2 \right) + u_0 + v_0 = 1 \\
& u^i \geq 0 \quad i = 1, 2 \\
& v^i \geq 0 \quad i = 1, 2 \\
& v_0 \geq 0.
\end{aligned}
$$

The original Benders' feasibility cut $rx \leq r_0$ is still added to the RMP.

The traditional use of split cuts are within the context of discrete optimization in which the CGLP (58) is evaluated at a fractional point $\overline{x}$. If the CGLP is evaluated at the integer solution x^* to the RMP, a trivial cut will be returned since $x^* \in \Pi_0$ or $x^* \in \Pi_1$. However, evaluating the CGLP from a point x^0 such that $\pi x^0 \in (\pi_0 - 1, \pi_0)$ may generate a nontrivial split cut $\alpha x \leq \alpha_0$ which cuts off the fractional point x^0. Figure 24 shows such an illustration.
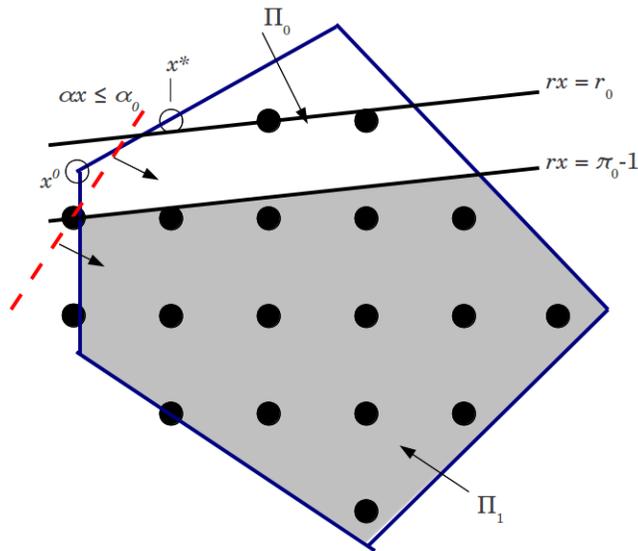
Figure 24: Benders-Induced Split Cuts

Different cuts may be generated depending upon the point $\bar{x}$ evaluated in the CGLP (60). The following result shows this approach generates split cuts that are not useful as they are dominated by the original Benders' cut. For a given it is desired to find a point $x^0 \in \{x : r_0 - 1 < r_0 x < r_0\}$ in order to generate a nontrivial split cut from the CGLP. The following results show how this can be done through following LP

$$
\begin{aligned}
\max \quad & \xi \\
\text{s.t.} \quad & rx \geq r_0 - 1 + \xi \\
& rx \leq r_0 - \xi \\
& (x, \eta) \in X\left(\widetilde{\Pi}, \widetilde{R}\right) \\
& \xi \geq 0.
\end{aligned}
\tag{61}
$$

Lemma 5.3.1. *If the LP (61) has a feasible solution it has a finite optimal objective value bounded by $\frac{1}{2}$.*

Proof. Note that the first two constraints suggest a valid solution must satisfy

$$
r_0 - 1 + \xi \leq r_0 - \xi \Leftrightarrow \xi \leq \frac{1}{2}.
$$

$\square$

Theorem 5.3.2. *Suppose the LP (61) has a feasible solution (x^0, ξ^*). Then $x^0 \in \{x : r_0 - 1 < rx < r_0\}$ if and only if $\xi^* \in \left(0, \frac{1}{2}\right]$.*

Proof. ($\Rightarrow$) If $x^0 \in \{x : r_0 - 1 < rx < r_0\}$, then $\xi^* > 0$ by the first two constraints present in (61). Moreover as x^0 is feasible, by Lemma 5.3.1, $\xi^* \leq \frac{1}{2}$.
($\Leftarrow$) If $\xi^* \leq \frac{1}{2}$, then $r_0 - 1 < rx < r_0$ for all $x \in X\left(\widetilde{\Pi}, \widetilde{R}\right)$, and thus for x^0. $\square$

The CGLP is likely to generate different cuts depending upon the point which is being evaluated in the objective. In practice, one can fix various values of $\xi \in \left(0, \frac{1}{2}\right]$ to find solutions from (61) that may generate different cuts to evaluate the Benders-induced CGLP (60). In general one may populate a set $\widehat{X}$ with different values in an attempt to generate different cuts. For instance, one may use a solution pool to populate points within a certain threshold of optimal values from (61). Another approach may be to define a set Ξ of different values ξ^* and populate $\widehat{X}$ with solutions satisfying different values of $\xi^* \in \Xi$.

If no feasible solution exists to (61), then no feasible solution exists to at least one set of the disjunction and no split cut is to be generated.

Example 5.3.3. *Recall Example 5.2.3 from Section 5.6.1. The associated Benders' disjunction from the cut (57) is*

$$\{2x_2 = 17\} \quad \vee \quad \{2x_2 \leq 16\}.$$

Let $\widehat{X} = \left\{\left(\frac{87}{8}, \frac{33}{4}\right)\right\}$ be a singleton solution to (61). The associated CGLP (58) then has a solution of $(\alpha_0^*, \alpha_1^*, \alpha_2^*) = \left(\frac{16}{10}, \frac{4}{10}, \frac{206}{10}\right)$ leading to the split cut $16x_1 + 4x_2 \leq 206$ which cuts off a region of X $(\emptyset, \emptyset)$, including the basic solution $\left(\frac{87}{8}, \frac{33}{4}\right)$.

The Benders' cut and its induced split cut are shown in Figure 25.

(a) Benders' feasibility cut

(b) Benders-induced disjunction

(c) Split cut

(d) Updated region $S(\emptyset, \{r\})$
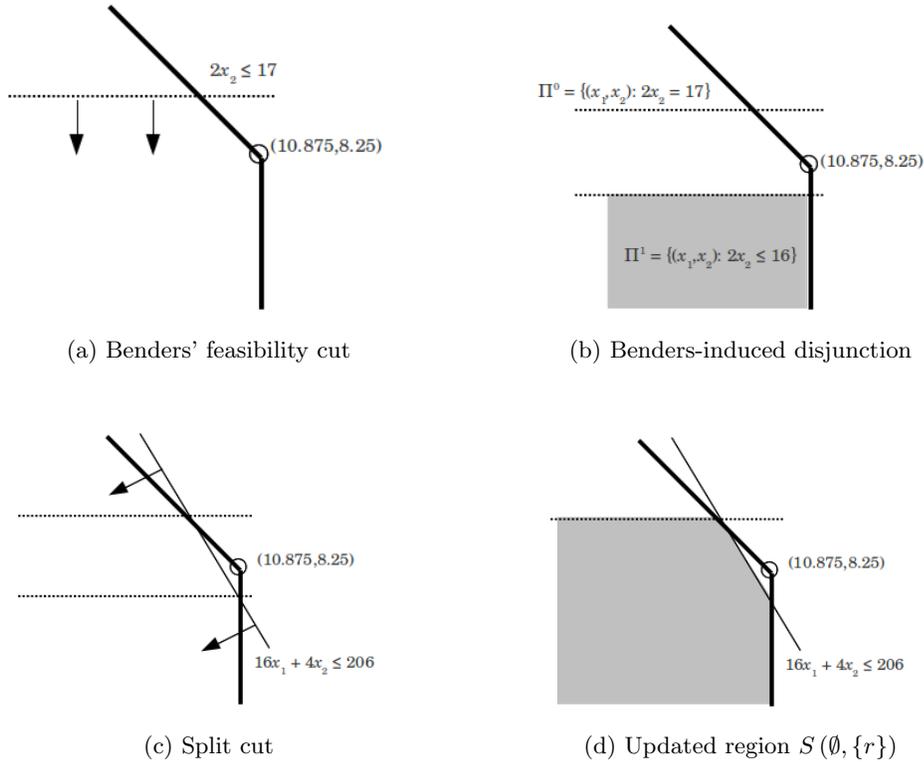
Figure 25: Example of Benders-Induced Split Cut

Algorithm 6 Benders-Induced Split Cut Algorithm

1: **Given:** Benders' feasibility cut $rx \leq r_0$
2: **if** $(r, r_0) \in \mathbb{Z}^n \times \mathbb{Z}$ **then**
3:     Consider the disjunction: $\{rx = r_0\} \vee \{rx \leq r_0 - 1\}$
4:     Let $\widehat{X}$ denote a set of points generated from the LP (61) where

$$rx \in (r_0 - 1, r_0) \quad \forall x \in \widehat{X}$$

5:     **while** $\widehat{X} \neq \emptyset$ **do**
6:         let $\overline{x} \in \widehat{X}$
7:         **solve** CGLP (60) evaluated at $\overline{x}$
8:         let (α, α_0) be a solution to (60)
9:         **if** $\alpha\overline{x} > \alpha_0$ **then**
10:             add split cut $\alpha x \leq \alpha_0$ to the RMP
11:         **end if**
12:         $\widehat{X} \leftarrow \widehat{X} \setminus \overline{x}$
13:     **end while**
14: **end if**
15: add Benders' feasibility cut $rx \leq r_0$ to RMP

Generating Benders-induced split cuts is summarized in Algorithm 6.

A hybrid version of the cut-pushing and split cut procedures may be used if the original Benders' cut is not binding at an integer solution. Therefore cut-pushing may be employed to ensure that some feasible solution is binding at the cut. This is shown in Algorithm 7.

Algorithm 7 A Modified Benders-Induced Split Cut Algorithm

1: **Given:** Benders' feasibility cut $rx \leq r_0$
2: **if** $(r, r_0) \in \mathbb{Z}^n \times \mathbb{Z}$ **then**
3:      Solve (54) for which Δ^* is the optimal objective value
4:      Consider the disjunction: $\{rx = r_0 - \Delta^*\} \vee \{rx \leq r_0 - (\Delta^* + 1)\}$
5:      Let $\widehat{X}$ denote a set of points generated from the LP (61) where

$$rx \in (r_0 - 1, r_0) \quad \forall x \in \widehat{X}$$

6:      **while** $\widehat{X} \neq \emptyset$ **do**
7:        let $\overline{x} \in \widehat{X}$
8:        **solve** CGLP (60) evaluated at $\overline{x}$
9:        let (α, α_0) be a solution to (60)
10:        **if** $\alpha \overline{x} > \alpha_0$ **then**
11:          add split cut $\alpha x \leq \alpha_0$ to the RMP
12:        **end if**
13:        $\widehat{X} \leftarrow \widehat{X} \setminus \overline{x}$
14:      **end while**
15: **end if**
16: add Benders' feasibility cut $rx \leq r_0$ to RMP

5.4 Strengthening by Lifting

Another standard technique that is used in conventional mixed-integer programming is to strengthen inequalities through *lifting*. This process takes as given a valid inequality and introduces binary variables that were not originally present by finding coefficients for which the augmented inequality remains valid. This chapter begins with a review of the concept of traditional lifting and discuss how lifting may be incorporated into standard Benders' decomposition.

Lifting was originally studied by Gomory [59] within the context of the group problem. It has initially been studied further by the likes of Padberg [90], Wolsey [128], among others. Further expositions are reviewed below.

5.4.1 A Review of Lifting in Mixed-Integer Programming

Consider a mixed $0 - 1$ optimization problem of the form

$$\min \left\{ \sum_{j \in N} c_j x_j : x \in S \right\}.$$

where

$$
S = \left\{ x \in \mathbb{R}_+^n : \begin{array}{ll} \sum_{j \in N} a_{ij} x_j \leq b_i, & i = 1, 2, \ldots, m \\ x_j \in \{0, 1\} & \forall j \in I \subseteq N \end{array} \right\} \tag{62}
$$

and I denotes the set of binary variables. Let $C_0 \subset N$ and $\bigcup_{k=1}^t C_k$ denote a disjoint partition of $I \setminus C_0$. It is also assumed that $b_i \geq 0$ for all $i = 1, 2, \ldots, m$. In similar studies it is assumed that $m = t + 1$ in which there is a single row where all variables are present, and t rows that form a block structure over the variables present in $C_1, \ldots, C_t$.

For $q \in \{0, 1, \ldots, t\}$ let

$$
S^q = \left\{ x \in \mathbb{R}_+^{\sum_{0 \leq k \leq q} |C_k|} : \begin{array}{ll} \sum_{0 \leq k \leq q} \sum_{j \in C_k} a_{ij} x_j \leq b_i, & i = 1, 2, \ldots, m \\ x_j \in \{0, 1\}, & \forall j \in I \bigcap \left(\cup_{k=0}^q C_k \right) \end{array} \right\} \tag{63}
$$

and suppose that

$$
\sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j \leq \alpha_0 \tag{64}
$$

is a valid inequality for $conv(S^q)$ (note that $S^t = S$). For $q \leq t - 1$ the process of *lifting* is to introduce coefficients $\{\alpha_j\}$ for $j \in C_{q+1}$ for which

$$
\sum_{0 \leq k \leq q+1} \sum_{j \in C_k} \alpha_j x_j \leq \alpha_0 \tag{65}
$$

is valid for $conv(S^{q+1})$. Clearly (64) and (65) are both valid for $conv(S)$, and (65) dominates (64) since every solution to the former is a solution to the latter by setting $x_k = 0$ for all $k \in C_{q+1}$.

Suppose (64) is valid for some S^q where $q \in \{0, 1, \ldots, t - 1\}$. The *lifting problem* is to find coefficients $\{\alpha_j\}_{j \in C_{q+1}}$ for which (65) is valid for $S^{q+1} \subseteq S$. When the largest such coefficients are found, the lifting is said to be *maximal*. Gu et al. [62] showed that if $conv(S^q)$ and $conv(S^{q+1})$ are full dimensional, $\alpha_0 \neq 0$, and (64) defines a facet of $conv(S^q)$, then (65) defines a facet of $conv(S^{q+1})$ if and only if the lifting is maximal.

The preceding illustrates lifting in binary variables that are originally not present in an inequality. Lifting is more general than this particular study, and more general studies of lifting can be seen in Richard et al. [95], [96] and Narisetty et al. [86].

For the remainder of this work, we will assume that only binary variables are lifted into a valid inequality so $\bigcup_{k=1}^{t} C_k$ form a partition of the set $I \setminus C_0$. In other words, fractional variables are present in the final inequality if and only if they belong to the original valid inequality.

For a given set $C_0 \subset N$ where $\sum_{j \in C_0} \alpha_j x_j \leq \alpha_0$ is valid for S^0, lifting was originally studied whereby binary variables are lifted in one at a time. That is, $m = |I \setminus C_0|$ lifting problems are solved and C_i denote singleton subsets for $i = 1, 2, \ldots, m$.

Let x_i denote some variable for which $i \in I \setminus C_0$ where $C_1 = \{i\}$. The lifting problem is to find a coefficient α_1 for which

$$\sum_{j \in C_0} \alpha_j x_j + \alpha_i x_i \leq \alpha_0$$

is valid for $conv(S^1)$. The inequality is trivially valid for all values of α_i when $x_i = 0$. If $x_i = 1$, then

$$\alpha_i \leq \alpha_0 - \sum_{j \in C_0} \alpha_j x_j$$

for all $x \in S^0$, and maximal for which the preceding relationship is binding for some x. A maximal lifting is then satisfied by solving the problem

$$\alpha_i^* = \max \left\{ \sum_{0 \leq k \leq 1} \sum_{j \in C_k} \alpha_j x_j : x \in S^1 \cap \{x : x_i = 1\} \right\}.$$

so that

$$\sum_{j \in C_0} \alpha_j x_j + \alpha_i^* x_i \leq \alpha_0.$$

is valid for $conv(S^1)$. This procedure is repeating for singleton sets $C_2, \ldots, C_t$, or terminated if $\alpha_i = 0$ since every successive iteration would yield the same value of the lifting problem.

In general for $q = 1, 2, \ldots, t$ let i be such that $C_q = \{i\}$. The qth lifting problem is of the form

$$\alpha_q^* = \max \left\{ \sum_{0 \leq k \leq q} \alpha_j x_j : x \in S^q \cap \{x : x_i = 1\} \right\}. \tag{66}$$

Lifted cover inequalities for a $0 - 1$ knapsack problem were among the first to illustrate the use of sequential lifting procedures.

5.4.2 Using Lifting in Benders' Decomposition

Consider a standard Benders' cut

$$\psi x \le \psi_0 + \mathbf{1}^{\text{opt}} \eta \tag{67}$$

Note that only those variables from the RMP present in the cut are those whose corresponding component from the dual extreme ray or dual extreme point are nonzero. Let $N = \{1, 2, \ldots, n\}$ denote all master variables and $I \subseteq N$ be those restricted to be binary. Let $C_0 \subseteq N$ be defined as

$$C_0 \equiv \{j \in N : \psi_j = 0\}.$$

The Benders' cut (67) can equivalently be expressed as

$$\sum_{j \in C_0} \psi_j x_j \le \psi_0 + \mathbf{1}^{\text{opt}} \eta. \tag{68}$$

This suggests that (67) may be strengthened by lifting in variables from a set $I' \equiv I \cap (C \setminus C_0)$ where it is assumed that $I' \ne \emptyset$. Let $C_1, C_2, \ldots, C_t$ denote a disjoint subsets of I'. The original inequality (67) is strengthened by successively solving the lifting problem for sets $C_1, \ldots, C_t$. Note that $C_1, \ldots, C_t$ need not partition I' as excessive calls to the lifting problem may more than offset the strengthened inequality.

Consider a problem solved by Benders' decomposition whose feasible set consists of original constraints of the form $Ax \le b$ along with all Benders' cuts. Let $\mathcal{B}^{\text{opt}}$ and $\mathcal{B}^{\text{feas}}$ denote the set of Benders' optimality and feasibility cuts, respectively. Given the original cut (68), the set analogous to (63) for $q \in \{0, 1, \ldots, t\}$ is given by

$$S^q = \left\{ x \in \mathbb{R}_+^{\sum_{0 \le k \le q} |C_k|} : \begin{array}{ll} \sum_{0 \le k \le q} \sum_{j \in C_k} a_{ij} x_j & \le b_i, \qquad i = 1, 2, \ldots, m \\ \sum_{0 \le k \le q} \sum_{j \in C_k} \pi_j^i x_j & \le \pi_0^i + \eta, \quad \forall i \in \mathcal{B}^{\text{opt}} \\ \sum_{0 \le k \le q} \sum_{j \in C_k} r_j^i x_j & \le r_0^i, \qquad \forall i \in \mathcal{B}^{\text{feas}} \\ x_j & \in \{0, 1\}, \quad \forall j \in I \cap \left(\cup_{k=0}^q C_k \right). \end{array} \right\} \tag{69}$$

Algorithm 8 summarizes how Benders' cuts may be strengthened through lifting procedure.

128

Algorithm 8 Benders' Lifting Algorithm

1: **Given:** Benders' cut

$$\sum_{j \in C_0} \psi_j x_j \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta$$

where $C_0 \equiv \{j \in N : \psi_j = 0\}$

2: let $I' \equiv I \setminus C_0$
3: **if** $I' \neq \emptyset$ **then**
4:     define a disjoint subsets such that $\bigcup_{k=1}^{t} C_k \subseteq I'$
5:     **for** $q = 1, 2, \ldots, t$ **do**
6:        let i be such that $C_q = \{i\}$
7:        define S^q as in (69)
8:        Solve the qth lifting problem:

$$\psi_i = \psi_0 - \max \left\{ \sum_{k=1}^{q} \sum_{j \in C_k} \psi_j x_j - \mathbf{1}^{\text{opt}} \eta : x \in S^q \cap \{x : x_i = 1\} \right\}$$

9:        **if** $\psi_i \neq 0$ **then**
10:          **continue;** $\sum_{0 \leq k \leq q} \sum_{j \in C_k} \psi_j x_j \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta$ is valid for $conv(S^q)$
11:        **else**
12:          Cut cannot be strengthened, set $\psi_j = 0$ for all $j \geq q$,
         $q = t$
13:        **end if**
14:     **end for**
15: **else**
16:     Set $\psi_j = 0$ for all $j \in N_0$
17: **end if**
18: **return** the strengthened Benders' cut

$$\sum_{0 \leq k \leq t} \sum_{j \in C_k} \psi_j x_j \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta$$

Example 5.4.1 (Christensen and Pedersen [35]). *Consider the uncapacitated facility location problem that seeks to satisfy demand from m clients from n potential facility locations that may be opened. Let x_j denote binary variables that dictate whether or not facility j are to be opened for all $j \in J = \{1, 2, \ldots, n\}$. Let y_{ij} denote the fraction of demand from client $i \in I = \{1, 2, \ldots, m\}$ that is satisfied from facility $j \in J$. The firm's problem is to minimize the total cost associated with opening plants and shipping the good from plants to clients given by*

$$\min \quad \sum_{j=1}^{n} f_j x_j + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} y_{ij} \tag{70}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} y_{ij} \geq 1, \quad i = 1, 2, \ldots, m \tag{71}$$

$$-x_j + y_{ij} \leq 0, \quad i = 1, 2, \ldots, m, \quad j = 1, 2, \ldots, n \tag{72}$$

$$(x, y) \in \mathbb{B}^{|J|} \times [0, 1]^{|I| \times |J|}. \tag{73}$$

Suppose $n = 3$ and $f_1 = 2, f_2 = 3$, and $f_3 = 3$. Suppose $m = 5$ and let the transportation costs c_{ij} be given in the following table.

Table 16: Transportation Costs c_{ij}

		client				
		1	**2**	**3**	**4**	**5**
factory	**1**	2	3	4	5	7
	2	4	3	1	2	6
	3	5	4	2	1	3

The problem may be solved by Benders' decomposition where x_j are the linking variables. Note that other than the binary restrictions on the x variables, only the Benders' cuts are present in the RMP. Let $x = (1, 0, 0)$ denote an initial feasible solution. This leads to the Benders' optimality cut $-2x_1 + 4x_2 + 7x_3 + \eta \geq 21$. No variables may be lifted since all binary variables are present in the cut. The RMP is then

$$\min \left\{ \eta : -2x_1 + 4x_2 + 7x_3 + \eta \geq 21, (x, \eta) \in \mathbb{B}^3 \times \mathbb{R}_+ \right\}$$

whose solution is $x^* = (0, 1, 1)$. The solution is suboptimal and generates the second feasibility cut

$$-3x_2 - 3x_3 + \eta \geq 12. \tag{74}$$

By rewriting the constraint by complementing x_2 and x_3 (since their coefficients are negative), the constraint is equivalent to $3x_2 + 3x_3 + \eta \geq 18$. The lifting problem seeks to find a coefficient α_1 such that $\alpha_1 x_1 + 3x_2 + 3x_3 + \eta \geq 18$ is valid for all x. The lifting problem is then to set α_1 such that

$$\alpha_1 = 18 - \min\left\{3x_2 + 3x_3 + \eta : 2x_1 + 4x_2 + 7x_3 + \eta \geq 23, x_1 = 1, (x, \eta) \in \mathbb{B}^3 \times \mathbb{R}_+\right\}$$

$$= 18 - 16 = 2$$

and thus $2x_1 + 3x_2 + 3x_3 + \eta \geq 12$, or by re-complementing x_2 and x_3, the strengthened inequality

$$2x_1 + 3x_2 + 3x_3 + 12 \leq \eta$$

is returned to the RMP.

5.5 Superaddititve Lifting

While lifting is a tractable mechanism for strengthening valid inequalities, one of its drawbacks is different sequences of sets $C_1, C_2, \ldots, C_t$ in general induce different inequalities, and therefore is said to be *sequence dependent*. Unless one has *a priori* knowledge on useful orderings of those subsets, strengthening the inequality depends on some degree of randomness with respect to sequence.

Gu et al. [62] show a sufficient condition for which lifting is independent of its sequence leading to more robust inequalities derived from lifting.

5.5.1 Review of Superaddititve Lifting

Consider a problem of the form $\min \left\{ \sum_{j \in N} c_j x_j : x \in S \right\}$ where S is described in (62). Let $Z \equiv [0, b_1] \times [0, b_2] \times \cdots \times [0, b_m]$ and $z = (z_1, z_2, \ldots, z_m) \in Z$ be arbitrary. For some $i \in \{1, 2, \ldots, t\}$ let $h_i(z)$ and $f_i(z)$ be two functions defined as follows:

$$
h_i(z) =
\begin{aligned}
\max \quad & \sum_{j \in C_i} \alpha_j x_j \\
\text{s.t.} \quad & \sum_{j \in C_i} a_{ij} x_j = z_i, \qquad i = 1, 2, \ldots, m \\
& \sum_{j \in C_i} w_j x_j \le r_i \\
& x_j \in \{0, 1\} \quad \forall j \in I \cap C_i \\
& x \in \mathbb{R}_+^{|C_i|}
\end{aligned}
\tag{75}
$$

and

$$
f_i(z) =
\begin{aligned}
\min \quad & \alpha_0 - \sum_{0 \le k \le i-1} \sum_{j \in C_k} \alpha_j x_j \\
\text{s.t.} \quad & \sum_{0 \le k \le i-1} \sum_{j \in C_k} a_{ij} x_j \le b_i - z_i && i = 1, 2, \ldots, m \\
& \sum_{j \in C_k} w_j x_j \le r_k && k = 0, 1, \ldots, i-1 \\
& x_j \in \{0, 1\} && \forall j \in I \cap \left(\cup_{k=0}^{i-1} C_k \right) \\
& x \in \mathbb{R}_+^{\sum_{0 \le k \le i-1} |C_k|}.
\end{aligned}
\tag{76}
$$

Using the convention that $h_i(z) = -\infty$ if $h_i(z)$ is infeasible for $z \in Z$, it follows that $h_i(z) \le f_i(z)$ for all $z \in Z$ as $f_i(z)$ is always feasible.

The central result uses the concept of a superaddititve function summarized in the following definition.

132

Definition 5.5.1. *A function $f : Z \to Z$ is said to be superadditive on Z if it is bounded, and if for all z_1, z_2, and $z_1 + z_2 \in Z$, $f(z_1) + f(z_2) \leq f(z_1 + z_2)$.*

The following result provides a sufficient condition for when lifting coefficients are independent of its sequence.

Theorem 5.5.2 (Gu et al. (2000)). *If the lifting function f is superadditive on Z, then lifting is sequence independent.*

Therefore under the conditions present in Theorem 5.5.2 all variables may be lifted concurrently.

While lifting functions generally are not superadditive, some important applications have been shown to be giving rise to sequence-independent lifting. Two important applications of superadditive lifting are reviewed that will be used in the subsequent results from superadditive lifting in a Benders' framework with multiple rows.

5.5.1.1   Review of Pure $0 - 1$ Knapsack

Consider the pure $0 - 1$ knapsack problem $\max\left\{\sum_{j \in N} c_j x_j : x \in Y\right\}$ where

$$Y = \left\{ x \in \mathbb{B}^n : \sum_{j \in N} a_j x_j \leq b \right\}$$

given $b \geq 0$ and $a_j \in [0, b]$ for all $j \in N$ (nonnegativity holds without loss of generality as binary variables may be complemented). A *cover* $C \subseteq N$ is a set such that $\sum_{j \in C} a_j > b$. The cover is said to be *minimal* if $\sum_{j \in C \setminus \{k\}} a_j \leq b$ for all $k \in C$. A cover inequality is $\sum_{j \in C} x_j \leq |C| - 1$ is valid for Y and facet-defining for $Y^C = \left\{ x \in \mathbb{B}^{|C|} : \sum_{j \in C} a_j x_j \leq b \right\}$. A lifted cover inequality seeks to find valid coefficients $\{\alpha_j\}_{j \in N \setminus C}$ such that $\sum_{j \in C} x_j + \sum_{j \in N \setminus C} \alpha_j x_j \leq |C| - 1$ is valid for $Y^N = Y$. The following pair of results from Gu et al. [62] show how superadditive lifting of a cover inequality can be used in the context of the pure $0 - 1$ knapsack problem.

Theorem 5.5.3 (Gu et al. [62]). *Given a cover inequality $\sum_{j \in C} x_j \leq |C| - 1$, the lifting*

function f is generally not superadditive over $Z = [0, b]$ *where*

$$f(z) = \min\left\{|C| - 1 - \sum_{j \in C} x_j : \sum_{j \in C} a_j x_j \leq b - z, x \in \mathbb{B}^{|C|}\right\}$$

$$= \max\left\{\sum_{j \in C} x_j : \sum_{j \in C} a_j x_j \leq b - z, x \in \mathbb{B}^{|C|}\right\}.$$

The preceding result shows that lifted cover inequalities are not superadditive. However, this can be generalized other other families of inequalities.

Lemma 5.5.4. *Given an inequality $\sum_{j \in M} \alpha_j x_j \leq \alpha_0$ that is valid for*
$X^0 = \left\{x \in \mathbb{B}^{|M|} : \sum_{j \in M} a_j x_j \leq b\right\}$ *for some $M \subset N$. The lifting function f is generally not superadditive over $Z = [0, b]$ where*

$$f(z) = \min\left\{\alpha_0 - \sum_{j \in M} \alpha_j x_j : \sum_{j \in M} a_j x_j \leq b - z, x \in \mathbb{B}^{|M|}\right\}$$

$$= \max\left\{\sum_{j \in M} \alpha_j x_j : \sum_{j \in M} a_j x_j \leq b - z, x \in \mathbb{B}^{|M|}\right\}.$$

Even when the pure lifting function f is in general not superadditive, superadditive lifting may still be employed by a valid lifting function g that approximates f such that $g(z) \leq f(z)$ for all $z \in Z$. Using this, the following approximate lifting function was shown to be valid for lifted cover inequalities. Assume without loss of generality that $a_1 \geq a_2 \geq \cdots \geq a_r$ where $C = \{1, 2, \ldots, r\}$.

Theorem 5.5.5 (Gu et al. [62]). *Given a cover inequality $\sum_{j \in C} x_j \leq |C| - 1$, the approximate lifting function g is superadditive over $Z = [0, b]$ where*

$$g(z) = \begin{cases} 0 & \text{for } z = 0 \\ h & \text{for } z \in (\mu_h - \lambda + \rho_h, \mu_{h+1} - \lambda], h = 0, 1, \ldots, r - 1 \text{,} \\ h - (\mu_h - \lambda + \rho_h - z) / \rho_1 & \text{for } z \in (\mu_h - \lambda, \mu_h - \lambda + \rho_h], h = 1, 2, \ldots, r - 1 \end{cases}$$

$$(77)$$

$\lambda \equiv \sum_{j \in C} a_j - b$, $\mu_0 \equiv 0$, and $\mu_i \equiv \sum_{1 \leq h \leq i} a_h$ for $i = 1, 2, \ldots, r$.

134

Consider now the previous problem with the addition of a single nonnegative variable to the right-hand side of the constraint leading to the mixed-integer $0-1$ knapsack problem $\max\left\{\sum_{j\in N} c_j x_j : x \in Y'\right\}$ where

$$Y' = \left\{(x, s) \in \mathbb{B}^n \times \mathbb{R}_+ : \sum_{j\in N} a_j x_j \leq b + s\right\}.$$

It is again assumed without loss of generality that $a_j > 0$ for all j as x_j may be complemented otherwise. Let $M \subset N$ and $\sum_{j\in M} \alpha_j x_j \leq \alpha_0$ be valid for $Y^0 = \left\{(x, s) \in \mathbb{B}^{|M|} \times \mathbb{R}_+ : \sum_{j\in M} a_j x_j \leq b + s\right\}$.

Theorem 5.5.6 (Marchand and Wolsey [81]). *Given an inequality $\sum_{j\in C} \alpha_j x_j \leq \alpha_0 + s$, the lifting function $f(z)$ is superadditive over $\mathbb{R}_+$ where*

$$f(z) = \min\left\{\alpha_0 - \sum_{j\in M} \alpha_j x_j + s : \sum_{j\in M} a_j x_j - s \leq b - z, (x, s) \in \mathbb{B}^n \times \mathbb{R}_+\right\}$$

$$= \max\left\{\sum_{j\in M} \alpha_j x_j - s : \sum_{j\in M} a_j x_j - s \leq b - z, (x, s) \in \mathbb{B}^n \times \mathbb{R}_+\right\}$$

Note that M need not be a cover, so the lifting holds with a general valid inequality for Y^0.

5.5.2 Superadditive Lifting for Relaxed Linear Systems

Most problems for which superadditive lifting is used are for those whose feasible set consists of a single row. Showing a lifting function is superadditive may be difficult for when $m = 1$, and approaching intractable for values of $m > 2$. This eliminates lifting for a large class of applications and likely explains why most studies have examined single-row systems. This section proposes one possible method to lift in variables of a valid inequality for which the lifting function has multiple rows. We examine relaxing the original feasible set S to a relaxed set $\overline{S}$ and using a lifting function that is superadditive for $\overline{S}$ to strengthen the original inequality.

As in Section 5.4 let $C_0, C_1, \ldots, C_t$ denote disjoint subsets of N. For all $q = 0, 1, \ldots, t$ let

$$S^q = \left\{ x \in \mathbb{R}_+^{\sum_{k=0}^{q} |C_k|} : \begin{array}{cc} \sum_{0 \le k \le q} \sum_{j \in C_k} a_{ij} x_j \le b_i, & i = 1, 2, \ldots, m \\ x_j \in \{0, 1\} & \forall j \in I \bigcap \left(\cup_{k=0}^{q} C_k \right) \end{array} \right\}$$

and note that $S^t = S$.

For some $C_0 \subset N$ consider a valid inequality for S^0 of the form

$$\sum_{j \in C_0} \alpha_j x_j \le \alpha_0. \tag{78}$$

Let $(z_1, z_2, \ldots, z_m) \in \mathcal{Z} = [0, b_1] \times [0, b_2] \times \cdots \times [0, b_m]$. The lifting function associated with (78) is

$$f(z_1, z_2, \ldots z_m) = \begin{array}{cl} \max & \sum_{j \in C_0} \alpha_j x_j \\ \text{s.t.} & \sum_{j \in C_0} a_{ij} x_j \le b_i - z_i, \quad i = 1, 2, \ldots, m \\ & x_j \in \{0, 1\}, \quad \forall j \in I \cap C_0. \end{array} \tag{79}$$

While showing (79) is superadditive is possible for $m = 1$ or 2, it is generally intractable for values of $m \ge 3$. Therefore in such circumstances we approximate (79) by relaxing S as follows. Let $\mu_1, \mu_2, \ldots, \mu_m$ denote nonnegative multipliers of the rows of the first set of constraints in (79). After aggregating the nonnegative linear combination, the *relaxed lifting function* is defined for all $z \in \overline{\mathcal{Z}} = [0, \sum_{i=1}^{m} \mu_i b_i]$:

$$f(z; \mu) = \begin{array}{cl} \max & \sum_{j \in C_0} \alpha_j x_j \\ \text{s.t.} & \sum_{j \in C_0} \widehat{a}_j x_j \le \widehat{b} - z \\ & x_j \in \{0, 1\}, \quad \forall j \in I \cap C_0 \end{array} \tag{80}$$

where $\widehat{a}_j = \sum_{i=1}^{m} \mu_i a_{ij}$ and $\widehat{b} = \sum_{i=1}^{m} \mu_i b_i$.

Given $\mu \in \mathbb{R}_+^m$ for all $q = 0, 1, \ldots, t$ let

$$S^q(\mu) = \left\{ x \in \mathbb{R}_+^{\sum_{k=0}^{q} |C_k|} : \begin{array}{cc} \sum_{0 \le k \le q} \sum_{j \in C_k} \widehat{a}_j x_j \le \widehat{b} \\ x_j \in \{0, 1\} & \forall j \in I \bigcap \left(\cup_{k=0}^{q} C_k \right) \end{array} \right\}$$

and note that $S^t(\mu) = \overline{S}(\mu)$.

The relaxed lifting function amounts to a knapsack problem which for which a superadditive function may be known depending upon whether or not $I = N$ (see Theorems 5.5.3

and 5.5.6). The following result shows that if the relaxed function (80) is superadditive over $\overline{S}^q(\mu)$ it remains superadditive over S^q for all $q = 0, 1, \ldots, t$.

Theorem 5.5.7. *Suppose $f(x)$ is a valid superadditive lifting function defined over a set X. Then $f(x)$ is superadditive over every subset Y of X.*

Proof. Let $y_1, y_2, y_1 + y_2 \in Y$. Since $Y \subseteq X$, $y_1, y_2, y_1 + y_2 \in X$. Thus $f(y_1 + y_2) \leq f(y_1) + f(y_2)$ since f is superadditive over X. □

Finally, when coefficients are determined over the relaxation that the lifted inequality remains valid for the original problem.

Theorem 5.5.8. *For any $q = 0, 1, \ldots, t - 1$ suppose*

$$\sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j \leq \alpha_0$$

is valid for S^q. Suppose coefficients $\{\alpha_j\}_{j \in C_{q+1}}$ are determined in accordance to the following relaxed lifting function where $z \in \overline{\mathcal{Z}}$ and $\mu \in \mathbb{R}_+^m$:

$$
\begin{aligned}
f(z) &= \min \left\{ \alpha_0 - \sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \leq k \leq q} \sum_{j \in C_k} \widehat{a}_j x_j \leq \widehat{b} - z, x \in \overline{S}^q(\mu) \right\} \\
&= \max \left\{ \sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \leq k \leq q} \sum_{j \in C_k} \widehat{a}_j x_j \leq \widehat{b} - z, x \in \overline{S}^q(\mu) \right\}.
\end{aligned}
$$

Then the lifted inequality

$$\sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j + \sum_{j \in C_{q+1}} \alpha_j x_j \leq \alpha_0$$

is valid for S^{q+1}.

Proof. For any $q = 0, 1, \ldots, t - 1$ suppose $\sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j \leq \alpha_0$ is valid for S^q. Given coefficients $\{\alpha_j\}_{j \in C_{q+1}}$ generated from the relaxed superadditive lifting function (80) $\sum_{0 \leq k \leq q} \sum_{j \in C_k} \alpha_j x_j + \sum_{j \in C_{q+1}} \alpha_j x_j \leq \alpha_0$ is valid for $\overline{S}^{q+1}(\mu)$ which remains valid for S^{q+1} since $S^{q+1} \subseteq \overline{S}^{q+1}(\mu)$ for all $\mu \in \mathbb{R}_+^m$. □

5.5.3 Superadditive Lifting for Relaxed Systems for Benders' Cuts

We now consider how the methods discussed in Section 5.5.2 are applicable to Benders' decomposition. For a given iteration of Benders' decomposition let $\mathcal{B} = \mathcal{B}^{\text{feas}} \bigcup \mathcal{B}^{\text{opt}}$ denote the set of all Benders' cuts that have been derived where $\mathcal{B}^{\text{feas}}$ and $\mathcal{B}^{\text{opt}}$ denote the sets of feasibility and optimality cuts, respectively. Let

$$S(\mathcal{B}) = \left\{ (x, \eta) \in \mathbb{B}^n \times \mathbb{R}_+^n : \begin{array}{ll} \sum_{j \in N} a_j x_j \leq b_i, & i = 1, 2, \ldots, m \\ \sum_{j \in N} r_j^i x_j \leq r_0^i, & \forall i \in \mathcal{B}^{\text{feas}} \\ \sum_{j \in N} \pi_j^i x_j \leq \pi_0^i + \eta, & \forall i \in \mathcal{B}^{\text{opt}} \end{array} \right\} \tag{81}$$

be the feasible set for the Restricted Master Problem that includes the sets of Benders' cuts. Note that we assume all integer variables are binary in the RMP. We relax the original feasible set (81) by aggregating nonnegative weights over the rows.

Let

$$\sum_{j \in C_0} \psi_j x_j \leq \psi_0 + \mathbf{1}^{\text{opt}} \eta$$

denote a generated Benders' cut where $C_0 \equiv \{j \in N : \psi_j \neq 0\}$. The cut is successively strengthened by taking the original inequality and lifting in variables from sets $C_1, C_2, \ldots, C_t$ whose coefficients are determined by solving a lifting problem over a set $\overline{S}(\mathcal{B}) \subseteq S(\mathcal{B})$ that is based on a relaxation corresponding to aggregation of the constraints in $S(\mathcal{B})$. The key is that the lifting function over $\overline{S}(\mathcal{B})$ is either itself superadditive, or can be approximated by a valid superadditive function, so the final inequality is invariant to the lifting sequence. Two cases are examined dependent upon whether or not an optimality cut is already present in the RMP.

5.5.3.1 Superadditive Lifting when $\mathcal{B}^{opt} = \emptyset$

In the case where no optimality cuts have been added let $(\mu, \lambda) \in \mathbb{R}_+^m \times \mathbb{R}_+^{|\mathcal{B}^{\text{feas}}|}$ denote the nonnegative weights used in the aggregation $\overline{S}(\mathcal{B}; \mu, \lambda)$ so that

$$\overline{S}(\mathcal{B}; \mu, \lambda) = \left\{ x \in \mathbb{B}^n : \sum_{j \in N} \widehat{a}_j x_j \leq \widehat{b} \right\}$$

where

$$\hat{a}_j \equiv \sum_{i=1}^m \mu_i a_{ij} + \sum_{i \in \mathcal{B}^{\text{feas}}} \lambda_i r_j^i \quad \text{and}$$

$$\hat{b} \equiv \sum_{i=1}^m \mu_i b_i + \sum_{i \in \mathcal{B}^{\text{feas}}} \lambda_i r_0^i.$$

The lifting problem associated with C_{q+1} is

$$f(z) = \min \left\{ \alpha_0 - \sum_{0 \le k \le q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \le k \le q} \sum_{j \in C_k} \hat{a}_j x_j \le \hat{b} - z, x \in \mathbb{B}^{\sum_{k=0}^q |C_k|} \right\}$$

$$= \max \left\{ \sum_{0 \le k \le q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \le k \le q} \sum_{j \in C_k} \hat{a}_j x_j \le \hat{b} - z, x \in \mathbb{B}^{\sum_{k=0}^q |C_k|} \right\}.$$

Note the lifting function amounts to a pure $0 - 1$ knapsack problem for which is in general not superadditive (Theorem 5.5.3). One can, however, use an approximate function which is superadditive similar to that shown in (77).

Given the approximate valid superadditive lifting function g, the lifting is done iteratively for $q = 1, 2, \ldots, t$ over the sets

$$\overline{S}^q(\mathcal{B}; \mu, \lambda) = \left\{ x \in \mathbb{R}_+^{\sum_{0 \le k \le q} |C_k|} : \begin{array}{c} \sum_{0 \le k \le q} \sum_{j \in C_k} \hat{a}_j x_k \le \hat{b} \\ x_j \in \{0, 1\}, \quad \forall j \in I \bigcap \left(\cup_{k=0}^q C_k \right) \end{array} \right\}. \tag{82}$$

5.5.3.2   Superadditive Lifting when $\mathcal{B}^{opt} \ne \emptyset$

Let $(\mu, \lambda, \rho) \in \mathbb{R}_+^m \times \mathbb{R}^{|\mathcal{B}^{\text{feas}}|} \times \mathbb{R}^{|\mathcal{B}^{\text{opt}}|}$ be given $(\sum_{i \in \mathcal{B}^{opt}} \rho_i \ne 0)$ where $\overline{S}(\mathcal{B}; \mu, \lambda, \rho)$ denotes the following relaxation over $S(\mathcal{B})$:

$$\overline{S}(\mathcal{B}; \mu, \lambda, \rho) = \left\{ (x, \eta) \in \mathbb{B}^n \times \mathbb{R}_+ : \sum_{j \in N} \hat{a}_j x_j \le \hat{b} + \eta \right\} \tag{83}$$

where

$$\hat{a}_j \equiv \frac{\sum_{i=1}^m \mu_i a_{ij} + \sum_{i \in \mathcal{B}^{\text{feas}}} \lambda_i r_j^i + \sum_{i \in \mathcal{B}^{\text{opt}}} \rho_i \pi_j^i}{\sum_{i \in \mathcal{B}^{\text{opt}}} \rho_i} \quad \text{and}$$

$$\hat{b} \equiv \frac{\sum_{i=1}^m \mu_i b_i + \sum_{i \in \mathcal{B}^{\text{feas}}} \lambda_i r_0^i + \sum_{i \in \mathcal{B}^{\text{opt}}} \rho_i \pi_0^i}{\sum_{i \in \mathcal{B}^{\text{opt}}} \rho_i}.$$

The lifting problem associated with C_{q+1} is

$$f(z) = \min \left\{ \alpha_0 - \sum_{0 \le k \le q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \le k \le q} \sum_{j \in C_k} \widehat{a}_j x_j \le \widehat{b} + \eta - z, x \in \mathbb{B}^{\sum_{k=0}^{q} |C_k|} \right\}$$

$$= \max \left\{ \sum_{0 \le k \le q} \sum_{j \in C_k} \alpha_j x_j : \sum_{0 \le k \le q} \sum_{j \in C_k} \widehat{a}_j x_j \le \widehat{b} + \eta - z, (x, \eta) \in \mathbb{B}^{\sum_{k=0}^{q} |C_k|} \times \mathbb{R}_+ \right\}.$$

The lifting function amounts to a $0-1$ knapsack problem with a single continuous variable for which is superadditive (Theorem 5.5.6).

The lifting is then done iteratively for $q = 1, 2, \ldots, t$ over the sets

$$\overline{S}^q(\mathcal{B}; \mu, \lambda, \rho) = \left\{ x \in \mathbb{R}^{\sum_{0 \le k \le q} |C_k|} : \begin{array}{c} \sum_{0 \le k \le q} \sum_{j \in C_k} \widehat{a}_j x_j \le \widehat{b} + \eta \\ x_j \in \{0, 1\}, \quad \forall j \in I \bigcap \left(\cup_{k=0}^{q} C_k \right) \end{array} \right\}. \quad (84)$$

There are four cases to consider for the superadditive lifting over the relaxation: whether or not there exists an optimality cuts, the following two may be employed:

1. lifting a feasibility cut for which no optimality cuts have been generated - lifting is over a 0-1 knapsack problem for which is not superadditive in general, but may be approximated by a valid superadditive function.

2. lifting an optimality cut for which there are other optimality cuts present in the RMP - lifting is over a 0-1 knapsack with a single continuous variable and is known to be superadditive.

While there are other cases (e.g. a feasibility cut for which there are optimality cuts generated in the feasible set of the RMP), they are not superadditive and no strong approximation to the superadditive function is known, similar to that shown in Gu et al. [62]).

Introducing sequence-independent lifting through a relaxed lifting function is summarized in Algorithm 9.

Algorithm 9 Relaxed Superadditive Lifting Algorithm

1: **Given:** Benders' cut
$$\sum_{j \in C_0} \psi_j x_j \le \psi_0 + \mathbf{1}^{\mathrm{opt}} \eta$$

where $C_0 \equiv \{j \in N : \psi_j = 0\}$

2: let $I' \equiv I \setminus C_0$

3: **if** $I' \ne \emptyset$ **then**

4:     define a disjoint subsets such that $\bigcup_{k=1}^{t} C_k \subseteq I'$

5:     **if** feasibility cut **then**

6:         **if** $\mathcal{B}^{\mathrm{opt}} = \emptyset$ **then**

7:             **for** $q = 1, 2, \ldots, t$ **do**

8:                 Define $(\mu, \lambda) \in \mathbb{R}_+^m \times \mathbb{R}_+^{|\mathcal{B}^{\mathrm{feas}}|}$

9:                 Construct relaxation $\overline{S}^q(\mathcal{B}; \mu, \lambda)$ as in (82)

10:                 Define valid superadditive lifting function g over $\overline{S}^q(\mathcal{B}; \mu, \lambda)$

11:                 For all $w \in C_q$ where $w = \{i\}$ solve lifting problem

$$\psi_i = \psi_0 - \max \left\{ g(x) : x \in \overline{S}^q(\mathcal{B}; \mu, \lambda) \cap \{x : x_i = 1\} \right\}$$

12:             **end for**

13:         **end if**

14:     **else**

15:         **if** $\mathcal{B}^{\mathrm{opt}} \ne \emptyset$ **then**

16:             **for** $q = 1, 2, \ldots, t$ **do**

17:                 Define $(\mu, \lambda, \rho) \in \mathbb{R}_+^m \times \mathbb{R}_+^{|\mathcal{B}^{\mathrm{feas}}|} \times \mathbb{R}_+^{|\mathcal{B}^{\mathrm{opt}}|}$

18:                 Construct relaxation $\overline{S}^q(\mathcal{B}; \mu, \lambda, \rho)$ as in (84)

19:                 Solve lifting problem

$$\psi_i = \psi_0 - \max \left\{ \sum_{0 \le k \le q} \sum_{j \in C_k} \psi_j x_j - \eta : (x, \eta) \in \overline{S}^q(\mathcal{B}; \mu, \lambda, \rho) \cap \{x : x_i = 1\} \right\}$$

20:             **end for**

21:         **end if**

22:     **end if**

23: **end if**

24: **return** the strengthened Benders' cut

$$\sum_{0 \le k \le t} \sum_{j \in C_k} \psi_j x_j \le \psi_0 + \mathbf{1}^{\mathrm{opt}} \eta$$

5.6 Computational Experiments from Airline Integrated Recovery

The cut strengthening methods proposed in this chapter are now performed on the Airline Integrated Recovery problem.

For a given iteration let $\mathcal{B}^{\text{feas}}$ denote the family of previously generated Benders' feasibility cuts for all subproblems. Let r^i denote the ith extreme ray $r^i = \left(r^{i,1}, r^{i,2}, r_0^i\right)$ used to generate a Benders' feasibility cut of the form $\sum_{e \in E} \sum_{s \in S} r_{e,s}^{i,1} x_{e,s} + \sum_{f \in F} r_f^{i,2} \kappa_f \leq r_0^i$ for some $i \in \mathcal{B}^{\text{feas}}$.

Let $\mathcal{B}_{\text{CRM}}^{\text{opt}}$ denote the set of Benders' optimality cuts generated from the CRM of the form $\sum_{f \in F} \pi_f^{i,2} \kappa_f \leq \pi_0^i + \eta^{\text{CRM}}$ where $\pi^i = \left(\pi^{i,2}, \pi_0\right)$ is a dual extreme point for the CRM that generates the ith feasibility cut for some $i \in \mathcal{B}_{\text{CRM}}^{\text{opt}}$. Finally, let $\mathcal{B}_{\text{IRM}}^{\text{opt}}$ denote the set of Benders' optimality cuts generated from the IRM of the form $\sum_{e \in E} \sum_{s \in S} \pi_{e,s}^{i,1} x_{e,s} + \sum_{f \in F} \pi_f^{i,2} \kappa_f \leq \pi_0^i + \eta^{\text{IRM}}$ where $\pi^i = \left(\pi^{i,1}, \pi^{i,2}, \pi_0\right)$ is a dual extreme point for the IRM that generates the ith IRM feasibility cut for some $i \in \mathcal{B}_{\text{IRM}}^{\text{opt}}$.

Let $\mathcal{B} = \mathcal{B}^{\text{feas}} \cup \mathcal{B}_{\text{CRM}}^{\text{opt}} \cup \mathcal{B}_{\text{IRM}}^{\text{opt}}$ denote all feasibility cuts that have been generated and $X\left(\mathcal{B}\right)$ represent the feasible set associated with the RMP given as follows:

$$X(\mathcal{B}) = \left\{ \begin{array}{l} \left(x, \kappa, \eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}}\right): \\[2mm] \begin{array}{rcll} \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} + \kappa_f & = 1 & \forall f \in F \\[2mm] \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} & = 1 & \forall f \in F^{\mathrm{strategic}} \\[2mm] \sum_{e \in E} \sum_{s \in I\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{arr}} & \forall \left(a, n_a^{\mathrm{arr}}, \underline{t}^a, \bar{t}^a\right) \in A^{\mathrm{arr}} \\[2mm] \sum_{e \in E} \sum_{s \in O\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{dep}} & \forall \left(a, n^{\mathrm{dep}}, \underline{t}^a, \bar{t}^a\right) \in A^{\mathrm{dep}} \\[2mm] \sum_{e \in E} \sum_{s \in W\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{gates}} & \forall \left(a, n_a^{\mathrm{gates}}, \underline{t}^a, \bar{t}^a\right) \in G \\[2mm] \sum_{e \in E} \sum_{s:s \ni f} CAP_e x_{e,s} & \geq n_f^{\mathrm{seats}} & \forall f \in F^{\mathrm{market}} \\[2mm] \sum_{e \in E} \sum_{s \in S} r_{e,s}^{i,1} x_{e,s} + \sum_{f \in F} r_f^{i,2} \kappa_f & \leq r_0^i & \forall i \in \mathcal{B}^{\mathrm{feas}} \\[2mm] \sum_{f \in F} \pi_f^{i,2} \kappa_f & \leq \pi_0^i + \eta^{\mathrm{CRM}} & \forall i \in \mathcal{B}_{\mathrm{CRM}}^{\mathrm{opt}} \\[2mm] \sum_{e \in E} \sum_{s \in S} \pi_{e,s}^{i,1} x_{e,s} + \sum_{f \in F} \pi_f^{i,2} \kappa_f & \leq \pi_0^i + \eta^{\mathrm{IRM}} & \forall i \in \mathcal{B}_{\mathrm{IRM}}^{\mathrm{opt}} \\[2mm] \eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}} & \geq 0 & \\[2mm] x_{e,s} & \{0,1\} & \forall e \in E, \forall s \in S \\[2mm] \kappa_f & \{0,1\} & \forall f \in F \end{array} \end{array} \right\}.$$

Consider a generic Benders' cut for the Airline Integrated Recovery problem of the form

$$\sum_{e \in E} \sum_{s \in S} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi_f^2 \kappa_f \leq \psi_0 + \mathbf{1}_{\mathrm{CRM}}^{\mathrm{opt}} \eta^{\mathrm{CRM}} + \mathbf{1}_{\mathrm{IRM}}^{\mathrm{opt}} \eta^{\mathrm{IRM}} \tag{85}$$

where $\mathbf{1}_{\mathrm{CRM}}^{\mathrm{opt}}$ and $\mathbf{1}_{\mathrm{IRM}}^{\mathrm{opt}}$ are indicator variables equal to unity if the inequality is a CRM or IRM optimality cut, respectively.

5.6.1 Cut Pushing

Using the framework introduced in Section the cut pushing subproblem is

$$\Delta^* = \min \left\{ \Delta : \sum_{e \in E} \sum_{s \in S} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi_f^2 \kappa_f + \Delta = \psi_0, \quad (x, \kappa, \eta) \in X(\mathcal{B}), \Delta \geq 0 \right\}. \tag{86}$$

In a traditional application of Benders' decomposition the RMP would be solved as a MIP. For any optimality cut $\Delta^* = 0$ since the cut is binding at the previous integer solution. However, variants of the algorithm allow for the RMP to be solved as an LP relaxation (this has been studied in McDaniel and Devine [83], among others). In these cases Δ^* may be

strictly positive inducing a stronger cut. For feasibility cuts the optimal value of (86) may be nonzero when the RMP is solved as a mixed integer or linear program.

After the subproblem (86) has been solved, the strengthend Benders' cut

$$\sum_{e \in E} \sum_{s \in S} \psi^1_{e,s} x_{e,s} + \sum_{f \in F} \psi^2_f \kappa_f \leq \psi_0 - \Delta^* + \mathbf{1}^{\mathrm{opt}}_{\mathrm{CRM}} \eta^{\mathrm{CRM}} + \mathbf{1}^{\mathrm{opt}}_{\mathrm{IRM}} \eta^{\mathrm{IRM}}$$

is added to the RMP which weakly dominate (85) if $\Delta^* = 0$ and strictly dominate if $\Delta^* > 0$.

5.6.2 Split Cuts

For a given iteration let $\mathcal{C}$ denote the family of previously generated split cuts. For the jth split cut let $\alpha^j = \left(\alpha^{j,1}, \alpha^{j,2}, \alpha^j_0 \right)$ denote the coefficients generated from the CGLP of the form $\sum_{e \in E} \sum_{s \in S} \alpha^{j,1}_{e,s} x_{e,s} + \sum_{f \in F} \alpha^{j,1}_f \kappa_f \leq \alpha^j_0$ where $j \in \mathcal{C}$.

This section will re-define the feasible set for the RMP from above that will include the split cuts and upper bound on all binary variables. For this section let

$$X'(\mathcal{B}, \mathcal{C}) = \left\{ \begin{array}{ll} \left(x, \kappa, \eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}} \right) : \\[2mm] \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} + \kappa_f = 1 & \forall f \in F \\[2mm] \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} = 1 & \forall f \in F^{\mathrm{strategic}} \\[2mm] \sum_{e \in E} \sum_{s \in I\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \leq n^{\mathrm{arr}}_a & \forall \left(a, n^{\mathrm{arr}}_a, \underline{t}^a, \bar{t}^a \right) \in A^{\mathrm{arr}} \\[2mm] \sum_{e \in E} \sum_{s \in O\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \leq n^{\mathrm{dep}}_a & \forall \left(a, n^{\mathrm{dep}}, \underline{t}^a, \bar{t}^a \right) \in A^{\mathrm{dep}} \\[2mm] \sum_{e \in E} \sum_{s \in W\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} \leq n^{\mathrm{gates}}_a & \forall \left(a, n^{\mathrm{gates}}_a, \underline{t}^a, \bar{t}^a \right) \in G \\[2mm] \sum_{e \in E} \sum_{s:s \ni f} CAP_e x_{e,s} \geq n^{\mathrm{seats}}_f & \forall f \in F^{\mathrm{market}} \\[2mm] \sum_{e \in E} \sum_{s \in S} r^{i,1}_{e,s} x_{e,s} + \sum_{f \in F} r^{i,2}_f \kappa_f \leq r^i_0 & \forall i \in \mathcal{B}^{\mathrm{feas}} \\[2mm] \sum_{e \in E} \sum_{s \in S} \alpha^1_{e,s} x_{e,s} + \sum_{f \in F} \alpha^2_f \kappa_f \leq \alpha^i_0 & \forall i \in \mathcal{C} \\[2mm] x_{e,s} \leq 1 & \forall e \in E, \forall s \in S \\[2mm] \kappa_f \leq 1 & \forall f \in F \\[2mm] \sum_{f \in F} \pi^{i,2}_f \kappa_f \leq \pi^i_0 + \eta^{\mathrm{CRM}} & \forall i \in \mathcal{B}^{\mathrm{opt}}_{\mathrm{CRM}} \\[2mm] \sum_{e \in E} \sum_{s \in S} \pi^{i,1}_{e,s} x_{e,s} + \sum_{f \in F} \pi^{i,2}_f \kappa_f \leq \pi^i_0 + \eta^{\mathrm{IRM}} & \forall i \in \mathcal{B}^{\mathrm{opt}}_{\mathrm{IRM}} \\[2mm] \eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}} \geq 0 \\[2mm] x_{e,s} \in \{0,1\} & \forall e \in E, \forall s \in S \\[2mm] \kappa_f \in \{0,1\} & \forall f \in F \end{array} \right\}.$$

An iteration of the Restricted Master Problem is an optimization problem whose feasible set consists of all SRM constraints (see Section 3.3.1) along with:

1. the set of all generated Benders cuts $\mathcal{B} = \mathcal{B}^{\text{feas}} \cup \mathcal{B}^{\text{opt}}_{\text{CRM}} \cup \mathcal{B}^{\text{opt}}_{\text{IRM}}$

2. the set of all generated split cuts $\mathcal{C}$.

Let $\overline{X}(\mathcal{B}, \mathcal{C})$ denote the continuous relaxation for $X'(\mathcal{B}, \mathcal{C})$ as defined above, and for a given iteration of the decomposition procedure let $Y(\mathcal{B}, \mathcal{C})$ denote the constraints from $\overline{X}(\mathcal{B}, \mathcal{C})$ together with the family of generated split cuts from $\mathcal{C}$. That is,

$$Y(\mathcal{B}, \mathcal{C}) = \overline{X}(\mathcal{B}, \mathcal{C}) \cap \left\{ (x, \kappa) : \sum_{e \in E} \sum_{s \in S} \alpha^{i,1}_{e,s} x_{e,s} + \sum_{f \in F} \alpha^{i,2}_f \kappa_f \leq \alpha^i_0, \quad \forall i \in \mathcal{C} \right\}.$$

For a given iteration of Benders' decomposition suppose the solution to the RMP induces an infeasible solution for a subproblem. Let $r = \left(q^1, q^2, q_0 \right)$ denote an extreme ray whose feasibility cut is of the form

$$\sum_{e \in E} \sum_{s \in S} q^1_{e,s} x_{e,s} + \sum_{f \in F} q^2_f \kappa_f \leq q_0.$$

Recall that only feasibility cuts are applicable to induce a split cut since no continuous variables are present. Moreover, the coefficients r are required to be integral. From Section 5.3, we wish to induce a split cut of the form

$$\sum_{e \in E} \sum_{s \in S} \alpha^1_{e,s} x_{e,s} + \sum_{f \in F} \alpha^2_f \kappa_f \leq \alpha_0 \tag{87}$$

from the disjunction

$$\left\{ \sum_{e \in E} \sum_{s \in S} q^1_{e,s} x_{e,s} + \sum_{f \in F} q^2_f \kappa_f = q_0 \right\} \quad \bigvee \quad \left\{ \sum_{e \in E} \sum_{s \in S} q^1_{e,s} x_{e,s} + \sum_{f \in F} q^2_f \kappa_f \leq q_0 - 1 \right\}.$$

Let

$$\Pi^0 = Y(\mathcal{B}, \mathcal{C}) \cap \left\{ (x, \kappa) : \sum_{e \in E} \sum_{s \in S} q^1_{e,s} x_{e,s} + \sum_{f \in F} q^2_f \kappa_f = q_0 \right\} \quad \text{and}$$

$$\Pi^1 = Y(\mathcal{B}, \mathcal{C}) \cap \left\{ (x, \kappa) : \sum_{e \in E} \sum_{s \in S} q^1_{e,s} x_{e,s} + \sum_{f \in F} q^2_f \kappa_f \leq q_0 - 1 \right\}.$$

Coefficients of the split cut $\alpha = \left(\alpha^1, \alpha^2, \alpha_0\right)$ from (87) are generated from the CGLP using sets Π^0 and Π^1. Let $u^i, v^i, i = 1, 2, \ldots, 12$ denote multipliers from the first twelve constraints from $Y\left(\mathcal{B}, \mathcal{C}\right)$ (or equivalently, the first twelve rows of X') and let u_0, v_0 denote scalars corresponding to multipliers from the respective disjunctions.

Note that $u^i = v^i = 0$ for $i = 11, 12$ since the multipliers are associated with Benders' optimality cuts whose relation is implied (see Section 5.3.2 for derivation).

In order to formulate the CGLP associated with the RMP for the Airline Integrated Recovery problem, the following notation is introduced:

$$
\Lambda^0(u) = \begin{aligned}
& \sum_{f \in F} u_f^1 + \sum_{f \in F^{\text{strategic}}} u_f^2 + \sum_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a) \in A^{\text{arr}}} n_a^{\text{arr}} u_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a)}^3 \\
& + \sum_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a) \in A^{\text{dep}}} n_a^{\text{dep}} u_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a)}^4 + \sum_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a) \in G} n_a^{\text{gates}} u_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a)}^5 \\
& + \sum_{f \in F^{\text{mrkt}}} n_f^{\text{seats}} u_f^6 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_0^{i,1} u_i^7 + \sum_{i \in \mathcal{C}} \alpha_0^i u_i^8 \\
& + \sum_{e \in E} \sum_{s \in S} u_{e,s}^9 + \sum_{f \in F} u_f^{10} + q_0 u_0
\end{aligned}
$$

$$
\Psi_{e,s}^0(u) = \begin{aligned}
& \sum_{f \in s} u_f^1 + \sum_{\substack{s: s \ni f \\ f \in F^{\text{strategic}}}} u_f^2 + \sum_{s: s \in I(a, \underline{t}^a, \bar{t}^a)} u_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a)}^3 + \sum_{s: s \in O(a, \underline{t}^a, \bar{t}^a)} \\
& u_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a)}^4 + \sum_{s: s \in W(a, \underline{t}^a, \bar{t}^a)} u_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a)}^5 \\
& + \sum_{\substack{s \ni f: \\ f \in F^{\text{mrkt}}}} CAP_e u_f^6 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_{e,s}^{i,1} u_i^7 \\
& + \sum_{i \in \mathcal{C}} \alpha_{e,s}^{i,1} u_i^8 + u_{e,s}^9 + q_{e,s}^1 u_0
\end{aligned}
$$

$$
\Omega_f^0(u) = u_f^1 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_f^{i,2} u_i^7 + \sum_{i \in \mathcal{C}} \alpha_f^{i,2} u_i^8 + u_f^{10} + q_f^2 u_0.
$$

For a Benders-induced split cut of the form

$$
\alpha_0 \geq \max \left\{ \sum_{e \in E} \sum_{s \in S} \alpha_{e,s}^1 x_{e,s} + \sum_{f \in F} \alpha_f^2 \kappa_f : (x, \kappa) \in \Pi^0 \right\}
$$

the following relation is used in the CGLP:

$$\alpha_0 \geq \min \Lambda^0(u)$$

$$\text{s.t.} \quad \Psi^0_{e,s}(u) \geq \alpha^1_{e,s} \quad \forall e \in E, \forall s \in S$$

$$\Omega^0_f(u) \geq \alpha^2_f \quad \forall f \in F \tag{88}$$

$$u^i \geq 0 \quad i = 1, 2, \dots, 10$$

$$u_0 \in \mathbb{R}.$$

The analogous condition with respect to Π^1 is derived in the same manner by introducing the following notation:

$$\Lambda^1(v) = \begin{aligned} &\textstyle\sum_{f \in F} v^1_f + \sum_{f \in F^{\text{strategic}}} v^2_f + \sum_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a) \in A^{\text{arr}}} n_a^{\text{arr}} v^3_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a)} \\ &+ \textstyle\sum_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a) \in A^{\text{dep}}} n_a^{\text{dep}} v^4_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a)} + \sum_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a) \in G} n_a^{\text{gates}} v^5_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a)} \\ &+ \textstyle\sum_{f \in F^{\text{mrkt}}} n_f^{\text{seats}} v^6_f + \sum_{i \in \mathcal{B}^{\text{feas}}} r^{i,1}_0 v^7_i + \sum_{i \in \mathcal{C}} \alpha^i_0 v^8_i + \sum_{e \in E} \sum_{s \in S} v^9_{e,s} \\ &+ \textstyle\sum_{f \in F} v^{10}_f + (q_0 - 1) v_0 \end{aligned}$$

$$\Psi^1_{e,s}(v) = \begin{aligned} &\textstyle\sum_{f \in s} v^1_f + \sum_{\substack{s: s \ni f \\ f \in F^{\text{strategic}}}} v^2_f + \sum_{s: s \in I(a, \underline{t}^a, \bar{t}^a)} v^3_{(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a)} \\ &+ \textstyle\sum_{s: s \in O(a, \underline{t}^a, \bar{t}^a)} v^4_{(a, n_a^{\text{dep}}, \underline{t}^a, \bar{t}^a)} + \sum_{s: s \in W(a, \underline{t}^a, \bar{t}^a)} v^5_{(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a)} \\ &+ \textstyle\sum_{\substack{s \ni f: \\ f \in F^{\text{mrkt}}}} CAP_e v^6_f + \sum_{i \in \mathcal{B}^{\text{feas}}} r^{i,1}_{e,s} v^7_i + \sum_{i \in \mathcal{C}} \alpha^{i,1}_{e,s} v^8_i + v^9_{e,s} + q^1_{e,s} v_0 \end{aligned}$$

$$\Omega^1_f(v) = v^1_f + \sum_{i \in \mathcal{B}^{\text{feas}}} r^{i,2}_f v^7_i + \sum_{i \in \mathcal{C}} \alpha^{i,2}_f v^8_i + v^{10}_f + q^2_f v_0.$$

For a Benders-induced split cut of the form

$$\alpha_0 \geq \max \left\{ \sum_{e \in E} \sum_{s \in S} \alpha^1_{e,s} x_{e,s} + \sum_{f \in F} \alpha^2_f \kappa_f : (x, \kappa) \in \Pi^1 \right\}$$

the following relation is used in the CGLP:

$$\alpha_0 \geq \min \Lambda^1(v)$$

$$\text{s.t.} \quad \Psi^1_{e,s}(v) \geq \alpha^1_{e,s} \quad \forall e \in E, \forall s \in S$$

$$\Omega^1_f(v) \geq \alpha^2_f \quad \forall f \in F \tag{89}$$

$$v^i \geq 0 \quad i = 1, 2, \ldots, 10$$

$$v_0 \in \mathbb{R}_+.$$

For a given point $\widehat{x}$, the associated CGLP associated with the RMP of the Airline Integrated Recovery problem is given as follows, as derived from (88) and (89):

$$\max \quad \sum_{e \in E} \sum_{s \in S} \alpha^1_{e,s} \overline{x}_{e,s} + \sum_{f \in F} \alpha^2_f \overline{\kappa}_f - \alpha_0$$

$$\text{s.t.} \quad \alpha_0 \geq \Lambda^0(u)$$

$$\alpha_0 \geq \Lambda^1(v)$$

$$\Psi^0_{e,s}(u) \geq \alpha^1_{e,s} \quad \forall e \in E, \forall s \in S$$

$$\Psi^1_{e,s}(v) \geq \alpha^1_{e,s} \quad \forall e \in E, \forall s \in S \qquad \text{(AIR-CGLP)}$$

$$\Omega^0_f(u) \geq \alpha^2_f \quad \forall f \in F$$

$$\Omega^1_f(v) \geq \alpha^2_f \quad \forall f \in F$$

$$\sum_{i=1}^{10} \sum_j u^i_j + \sum_{i=1}^{10} v^i_j + u_0 + v_0 = 1$$

$$u^i, v^i \geq 0 \quad i = 1, 2, \ldots, 10$$

$$(u_0, v_0) \in \mathbb{R} \times \mathbb{R}_+.$$

5.6.3 Standard Lifting

Strengthening a standard Benders' cut of the form (85) through sequential lifting is now examined. Let $S^1 = \left\{ s \in S : \exists e \in E \text{ s.t. } \psi^1_{e,s} \neq 0 \right\}$. Similarly let $F^1 = \left\{ f \in F : \psi^2_f \neq 0 \right\}$. One can lift in strings or flight cancellation variables. We will let $S^0 = S \backslash S^1$ and $F^0 = F \backslash F^1$ denote the strings or flight cancellation variables, respectively, that will be introduced in (85).

Lifting Flight String Variables Let $C_0 = S^1$ and $C_1, C_2, \ldots, C_t$ be disjoint subsets of S^0. For $q \in \{0, 1, \ldots, t\}$ let $X^q(\mathcal{B})$ denote the set analogous to $X(\mathcal{B})$ (so $X^t(\mathcal{B}) = X(\mathcal{B})$) defined only over $\bigcup_{0 \leq k \leq q} C_k$. That is,

$$
X^q(\mathcal{B}) = \left\{
\begin{array}{rll}
\left(x, \kappa, \eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}}\right) : & & \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq C_q} \sum_{s \in C_k : s \ni f} x_{e,s} + \kappa_f & = 1 & \forall f \in F \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k : s \ni f} x_{e,s} & = 1 & \forall f \in F^{\mathrm{strategic}} \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k \cap I\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{arr}} & \forall \left(a, n_a^{\mathrm{arr}}, \underline{t}^a, \bar{t}^a\right) \in A^{\mathrm{arr}} \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k \cap O\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{dep}} & \forall \left(a, n^{\mathrm{dep}}, \underline{t}^a, \bar{t}^a\right) \in A^{\mathrm{dep}} \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k \cap W\left(a, \underline{t}^a, \bar{t}^a\right)} x_{e,s} & \leq n_a^{\mathrm{gates}} & \forall \left(a, n_a^{\mathrm{gates}}, \underline{t}^a, \bar{t}^a\right) \in G \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k : s \ni f} CAP_e x_{e,s} & \geq n_f^{\mathrm{seats}} & \forall f \in F^{\mathrm{market}} \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k} r_{e,s}^{i,1} x_{e,s} & & \\[4pt]
+ \sum_{f \in F} r_f^{i,2} \kappa_f & \leq r_0^i & \forall i \in \mathcal{B}^{\mathrm{feas}} \\[4pt]
\sum_{f \in F} \pi_f^{i,2} \kappa_f & \leq \pi_0^i + \eta^{\mathrm{CRM}} & \forall i \in \mathcal{B}_{\mathrm{CRM}}^{\mathrm{opt}} \\[4pt]
\sum_{e \in E} \sum_{0 \leq k \leq q} \sum_{s \in C_k} \pi_{e,s}^{i,1} x_{e,s} & & \\[4pt]
+ \sum_{f \in F} \pi_f^{i,2} \kappa_f & \leq \pi_0^i + \eta^{\mathrm{IRM}} & \forall i \in \mathcal{B}_{\mathrm{IRM}}^{\mathrm{opt}} \\[4pt]
\eta^{\mathrm{CRM}}, \eta^{\mathrm{IRM}} & \geq 0 & \\[4pt]
x_{e,s} & \{0,1\} & \forall e \in E, \forall s \in \bigcup_{0 \leq k \leq q} C_k \\[4pt]
\kappa_f & \{0,1\} & \forall f \in F
\end{array}
\right\}.
$$

The first term in the Benders' cut (85) may then be expressed as $\sum_{e \in E} \sum_{s \in C_0} \psi_{e,s}^1 x_{e,s}$ and is valid for $X^0(\mathcal{B})$. Given some $\widetilde{s} \in C_1$ for which a valid equipment is eligible, the initial lifting problem is to determine a value of $\psi_{e,\widetilde{s}}^1$ such that

$$
\psi_{e,\widetilde{s}}^1 x_{e,\widetilde{s}} + \sum_{e \in E} \sum_{s \in C_0} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi_f^2 \kappa_f \leq \psi_0 + \mathbf{1}_{\mathrm{CRM}}^{\mathrm{opt}} \eta^{\mathrm{CRM}} + \mathbf{1}_{\mathrm{IRM}}^{\mathrm{opt}} \eta^{\mathrm{IRM}}
$$

is valid for $X^1(\mathcal{B})$.

In general suppose that from an original Benders' inequality (85), $q - 1$ variables have been lifted in so that

$$
\sum_{e \in E} \sum_{0 \leq k \leq q-1} \sum_{s \in C_k} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi_f^2 \kappa_f \leq \psi_0 + \mathbf{1}_{\mathrm{CRM}}^{\mathrm{opt}} \eta^{\mathrm{CRM}} + \mathbf{1}_{\mathrm{IRM}}^{\mathrm{opt}} \eta^{\mathrm{IRM}}
$$

is valid for $conv(X^{q-1}(\mathcal{B}))$. For some $\widetilde{e}, \widetilde{s} \in C_q$ the qth lifting problem is then given by

$$\psi^1_{\widetilde{e},\widetilde{s}} = \max \quad \sum_{e \in E} \sum_{k=1}^{q-1} \sum_{s \in C_k} \psi^1_{e,s} x_{e,s} + \sum_{f \in F} \psi^2_f \kappa_f - \mathbf{1}^{\text{opt}}_{\text{CRM}} \eta^{\text{CRM}} - \mathbf{1}^{\text{opt}}_{\text{IRM}} \eta^{\text{IRM}}$$

$$\text{s.t.} \quad \left(x, \kappa, \eta^{\text{CRM}}, \eta^{\text{IRM}}\right) \in X^q(\mathcal{B}) \cap \left\{x : x_{\widetilde{e},\widetilde{s}} = 1\right\}.$$

Lifting Flight Cancellation Variables Let $C_0 = S^1$ and $C_1, C_2, \ldots, C_t$ be disjoint subsets of F^0. For $q \in \{0, 1, \ldots, t\}$ let $X^q(\mathcal{B})$ denote the analogous set as the preceding case, but for the appropriate subset of flight cancellation variables that have been introduced, or

$$X^q(\mathcal{B}) = \left\{ \begin{array}{l} \left(x, \kappa, \eta^{\text{CRM}}, \eta^{\text{IRM}}\right): \\[6pt] \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} + \kappa_f = 1 \quad \forall f \in \bigcup_{0 \le k \le q} C_k \\[6pt] \sum_{e \in E} \sum_{s:s \ni f} x_{e,s} = 1 \quad \forall f \in F^{\text{strategic}} \bigcap \left(\cup_{0 \le k \le q} C_k\right) \\[6pt] \sum_{e \in E} \sum_{s \in I\left(a,\underline{t}^a,\bar{t}^a\right)} x_{e,s} \le n_a^{\text{arr}} \quad \forall \left(a, n_a^{\text{arr}}, \underline{t}^a, \bar{t}^a\right) \in A^{\text{arr}} \\[6pt] \sum_{e \in E} \sum_{s \in O\left(a,\underline{t}^a,\bar{t}^a\right)} x_{e,s} \le n_a^{\text{dep}} \quad \forall \left(a, n^{\text{dep}}, \underline{t}^a, \bar{t}^a\right) \in A^{\text{dep}} \\[6pt] \sum_{e \in E} \sum_{s \in W\left(a,\underline{t}^a,\bar{t}^a\right)} x_{e,s} \le n_a^{\text{gates}} \quad \forall \left(a, n_a^{\text{gates}}, \underline{t}^a, \bar{t}^a\right) \in G \\[6pt] \sum_{e \in E} \sum_{s:s \ni f} CAP_e x_{e,s} \ge n_f^{\text{seats}} \quad \forall f \in F^{\text{market}} \bigcap \left(\cup_{0 \le k \le q} C_k\right) \\[6pt] \sum_{e \in E} \sum_{s \in S} r_{e,s}^{i,1} x_{e,s} \\ + \sum_{0 \le k \le q} \sum_{f \in C_k} r_f^{i,2} \kappa_f \le r_0^i \quad \forall i \in \mathcal{B}^{\text{feas}} \\[6pt] \sum_{0 \le k \le q} \sum_{f \in C_k} \pi_f^{i,2} \kappa_f \le \pi_0^i + \eta^{\text{CRM}} \quad \forall i \in \mathcal{B}^{\text{opt}}_{\text{CRM}} \\[6pt] \sum_{e \in E} \sum_{s \in S} \pi_{e,s}^{i,1} x_{e,s} \\ + \sum_{0 \le k \le q} \sum_{f \in C_k} \pi_f^{i,2} \kappa_f \le \pi_0^i + \eta^{\text{IRM}} \quad \forall i \in \mathcal{B}^{\text{opt}}_{\text{IRM}} \\[6pt] \eta^{\text{CRM}}, \eta^{\text{IRM}} \ge 0 \\[6pt] x_{e,s} \in \{0,1\} \, \forall e \in E, \forall s \in S \\[6pt] \kappa_f \in \{0,1\} \, \forall f \in \bigcup_{0 \le k \le q} C_k \end{array} \right\}.$$

The second term in (85) may be expressed as $\sum_{f \in C_0} \psi^2_f \kappa_f$ and is valid for $X^0(\mathcal{B})$.

For some $\widetilde{f} \in C_1$ the lifting problem is to determine a value of $\psi^2_{\widetilde{f}}$ such that

$$\psi^2_{\widetilde{f}} \kappa_{\widetilde{f}} + \sum_{e \in E} \sum_{s \in S} \psi^1_{e,s} x_{e,s} + \sum_{f \in C_0} \psi^2_f \kappa_f \le \psi_0 + \mathbf{1}^{\text{opt}}_{\text{CRM}} \eta^{\text{CRM}} + \mathbf{1}^{\text{opt}}_{\text{IRM}} \eta^{\text{IRM}}$$

is valid for $X^1(\mathcal{B})$.

In general suppose that from an original Benders' inequality (85), $q-1$ variables have been lifted in so that

$$\sum_{e \in E} \sum_{s \in S} \psi_{e,s}^1 x_{e,s} + \sum_{0 \leq k \leq q-1} \sum_{f \in C_k} \psi_f^2 \kappa_f \leq \psi_0 + \mathbf{1}_{\text{CRM}}^{\text{opt}} \eta^{\text{CRM}} + \mathbf{1}_{\text{IRM}}^{\text{opt}} \eta^{\text{IRM}}$$

is valid for $conv(X^{q-1}(\mathcal{B}))$. For some $\widetilde{f} \in C_q$ the qth lifting problem is then given by

$$\psi_{\widetilde{f}}^2 = \max \quad \sum_{e \in E} \sum_{s \in S} \sum_{s \in C_k} \psi_{e,s}^1 x_{e,s} + \sum_{0 \leq k \leq q} \sum_{f \in C_k} \psi_f^2 \kappa_f - \mathbf{1}_{\text{CRM}}^{\text{opt}} \eta^{\text{CRM}} - \mathbf{1}_{\text{IRM}}^{\text{opt}} \eta^{\text{IRM}}$$

$$\text{s.t.} \quad \left(x, \kappa, \eta^{\text{CRM}}, \eta^{\text{IRM}}\right) \in X^q(\mathcal{B}) \cap \left\{\kappa : \kappa_{\widetilde{f}} = 1\right\}.$$

'

5.6.4 Superadditive Lifting from Relaxation

This section considers lifting in flight strings from the superadditive lifting over the relaxed feasible set (lifting cancellation variables are not considered in this section). While Marchand and Wolsey [81] show that lifting over a $0-1$ knapsack problem with a single continuous variable is superadditive, it is not necessarily the case for two continuous variables. Therefore superadditive lifting is done only when $\mathcal{B}_{\text{CRM}}^{\text{opt}} = \emptyset$. Recall the CRM is solved after several iterations of the SRM, IRM, and ARM (see Figure 18 from Chapter 3) so the strengthening will still be valid for most Benders' cuts applied.

Given the set $X(\mathcal{B})$ as described above, let $\mu = \left(\mu^1, \mu^2, \ldots, \mu^6\right)$ denote weights for all rows from the original set of constraints that exclude all Benders' cuts, where each subvector μ^j is nonnegative for $j = 1, 2, \ldots, 6$. Moreover let $\lambda \in \mathbb{R}_+^{|\mathcal{B}^{\text{feas}}|}$ denote the weight vector for the feasibility cuts, and $\rho = \rho^{\text{IRM}}$ denote the nonnegative weight vector corresponding to the optimality cuts from the IRM. If $\mathcal{B}_{\text{IRM}}^{\text{opt}} \neq \emptyset$ the weights must satisfy $\sum_{i \in \mathcal{B}_{\text{IRM}}^{\text{opt}}} \rho_i^{\text{IRM}} > 0$.

Following the notation above let

$$\sum_{e \in E} \sum_{s \in C_0} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi^2 \kappa_f \leq \psi_0 + \mathbf{1}_{\text{IRM}}^{\text{opt}} \eta^{\text{IRM}}$$

denote a Benders' cut where

$$C_0 = \{s \in S : \psi_{e,s} \neq 0 \quad \text{for some } e \in E\}.$$

There are two cases considered which lifting may be utilized to strengthen the Benders' cut.

Case 1: lifting feasibility cut and $\mathcal{B}_{\text{IRM}}^{\text{opt}} = \emptyset$ The relaxation to the system $X(\mathcal{B})$ is

$$\overline{X}(\mathcal{B}) = \left\{ x \in \mathbb{R}_+^{|E| \times |S|} : \begin{array}{c} \sum_{e \in E} \sum_{s \in S} \widehat{a}_{e,s}^1 x_{e,s} + \sum_{f \in F} \widehat{a}_f^2 \kappa_f \leq \widehat{b} \\ x_{e,s} \in \{0,1\} \quad \forall e \in E, \forall s \in S \\ \kappa_f \in \{0,1\} \quad \forall f \in F \end{array} \right\} \tag{90}$$

where for all $e \in E$ and $s \in S$

$$\widehat{a}_{e,s}^1 = \sum_{f \in s} \mu_f^1 + \sum_{\substack{s:s \ni f \\ f \in F^{\text{strategic}}}} \mu_f^2 + \sum_{s:s \in I(a,\underline{t}^a,\overline{t}^a)} \mu_{(a,n_a^{\text{arr}},\underline{t}^a,\overline{t}^a)}^3$$

$$+ \sum_{s:s \in O(a,\underline{t}^a,\overline{t}^a)} \mu_{(a,n_a^{\text{dep}},\underline{t}^a,\overline{t}^a)}^4 + \sum_{s:s \in W(a,\underline{t}^a,\overline{t}^a)} \mu_{(a,n_a^{\text{gates}},\underline{t}^a,\overline{t}^a)}^5 \tag{91}$$

$$+ \sum_{\substack{s \ni f: \\ f \in F^{\text{mrkt}}}} CAP_e \mu_f^6 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_{e,s}^{i,1} \lambda_i,$$

for all $f \in F$

$$\widehat{a}_f^2 = \mu_f^1 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_f^{i,2} \lambda_i, \tag{92}$$

and

$$\widehat{b} = \sum_{f \in F} \mu_f^1 + \sum_{f \in F^{\text{strategic}}} \mu_f^2 + \sum_{(a,n_a^{\text{arr}},\underline{t}^a,\overline{t}^a) \in A^{\text{arr}}} n_a^{\text{arr}} \mu_{(a,n_a^{\text{arr}},\underline{t}^a,\overline{t}^a)}^3$$

$$+ \sum_{(a,n_a^{\text{dep}},\underline{t}^a,\overline{t}^a) \in A^{\text{dep}}} n_a^{\text{dep}} \mu_{(a,n_a^{\text{dep}},\underline{t}^a,\overline{t}^a)}^4 + \sum_{(a,n_a^{\text{gates}},\underline{t}^a,\overline{t}^a) \in G} n_a^{\text{gates}} \mu_{(a,n_a^{\text{gates}},\underline{t}^a,\overline{t}^a)}^5 \tag{93}$$

$$+ \sum_{f \in F^{\text{mrkt}}} n_f^{\text{seats}} \mu_f^6 + \sum_{i \in \mathcal{B}^{\text{feas}}} r_0^{i,1} \lambda_i.$$

For $C_1, C_2, \ldots C_t$ that are disjoint subsets for $S' = S \setminus C_0$, the lifting is done iteratively over sets $\overline{X}^q(\mathcal{B}; \mu, \lambda)$ where

$$\overline{X}^q(\mathcal{B}; \mu, \lambda) = \left\{ x \in \mathbb{R}^{\sum_{0 \leq k \leq q} |C_k|} : \begin{array}{c} \sum_{0 \leq k \leq q} \sum_{j \in C_k} \widehat{a}_{e,s}^1 x_{e,s} + \sum_{f \in F} \widehat{a}_f^2 \kappa_f \leq \widehat{b} \\ x_{e,s} \in \{0,1\} \quad \forall e \in E, \forall s \in \bigcup_{k=0}^q C_k \\ \kappa_f \in \{0,1\} \quad \forall f \in F \end{array} \right\}.$$

Given a valid superadditive lifting function $g(x, \kappa)$ that is to approximate the true lifting function, for the qth lifting problem where $\widetilde{s} \in C_q$ is given by

$$\psi_{e,\widetilde{s}} = \psi_0 - \max\left\{ g(x, \kappa) : (x, \kappa) \in \overline{X}^q(\mathcal{B}; \mu, \lambda) \cap \{x : x_{e,\widetilde{s}} = 1\} \right\}.$$

Case 2: lifting optimality cut and $\mathcal{B}^{\text{opt}}_{\text{IRM}} \neq \emptyset$ The relaxation to the system $X(\mathcal{B})$ is now given by

$$\overline{X}(\mathcal{B}) = \left\{ x \in \mathbb{R}^{|E| \times |S|}_+ : \begin{array}{c} \sum_{e \in E} \sum_{s \in S} \widehat{a}^1_{e,s} x_{e,s} + \sum_{f \in F} \widehat{a}^2_f \kappa_f \leq \widehat{b} + \eta^{\text{IRM}} \\ x_{e,s} \in \{0,1\} \quad \forall e \in E, \forall s \in S \\ \kappa_f \in \{0,1\} \quad \forall f \in F \end{array} \right\}$$

where for all $e \in E$ and $s \in S$

$$\widehat{a}^1_{e,s} = \frac{(a^1_{e,s})'}{\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \rho^{\text{IRM}}_i},$$

for all $f \in F$

$$\widehat{a}^2_f = \frac{(a^2_f)'}{\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \rho^{\text{IRM}}_i},$$

and

$$\widehat{b} = \frac{b'}{\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \rho^{\text{IRM}}_i}$$

where:

- $(a^1_{e,s})'$ is described in (91) with the addition of the term $\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \pi^{i,1}_{e,s} \rho^{\text{IRM}}_i$ to the right-hand side

- $(a^2_f)'$ is described in (92) with the addition of the term $\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \pi^{i,2}_{e,s} \rho^{\text{IRM}}_i$ to the right-hand side

- b' is described in (93) with the addition of the term $\sum_{i \in \mathcal{B}^{\text{opt}}_{\text{IRM}}} \pi^i_0$ to the right-hand side.

For $C_1, C_2, \ldots C_t$ that are disjoint subsets for $S' = S \setminus C_0$, the lifting is done iteratively over sets $\overline{X}^q(\mathcal{B}; \mu, \lambda, \rho)$ where

$$\overline{X}^q(\mathcal{B}; \mu, \lambda, \rho) = \left\{ x \in \mathbb{R}^{\sum_{0 \leq k \leq q} |C_k|} : \begin{array}{c} \sum_{0 \leq k \leq q} \sum_{j \in C_k} \widehat{a}^1_{e,s} x_{e,s} + \sum_{f \in F} \widehat{a}^2_f \kappa_f \leq \widehat{b} + \eta \\ x_{e,s} \in \{0,1\} \quad \forall e \in E, \forall s \in \bigcup_{k=0}^q C_k \\ \kappa_f \in \{0,1\} \quad \forall f \in F \end{array} \right\}.$$

For the qth lifting problem where $\widetilde{s} \in C_q$ is given by

$$\psi_{e,\widetilde{s}} = \psi_0 - \max \left\{ \begin{array}{ll} & \sum_{e \in E} \sum_{k=0}^{q} \sum_{j \in C_k} \psi_{e,s}^1 x_{e,s} + \sum_{f \in F} \psi_f^2 \kappa_f - \eta^{\mathrm{IRM}} \\ \text{s.t.} & (x, \kappa, \eta) \in \overline{X}^q(\mathcal{B}; \mu, \lambda, \rho) \cap \left\{ x : x_{e,\widetilde{s}} = 1 \right\} \end{array} \right\}$$

which is superadditive by Marchand and Wolsey [81].

5.6.5 Computational Results from AIR Model

The preceding four proposals to accelerate Benders' decomposition by cut strengthening are now evaluated.

5.6.5.1 Benchmark Results

Table 17 summarizes the performance of each method relative to the standard approach used from Chapter 3 represented by the column labeled 'benchmark'. The subsequent columns represent each of the four cut-strengthening methods which are, respectively, cut-pushing (Section 5.6.1), Benders-induced split cuts (Section 5.6.2), sequential lifting (Section 5.6.3), and relaxed superadditive lifting (Section 5.6.4).

The following notes are important considerations for the initial results presented:

- Concerning the cut-pushing procedure, only feasibility cuts are considered as all iterations of the RMP are solved as integer programs.

- For the sequential lifting results presented, only flight string variables $x_{e,s}$ are being lifted, and the lifting order is determined by the number of flights in each string. That is, a pool is populated until either all eligible strings (those who are not present in the initial inequality) have been added, or until the pool size meets a predetermined limit. For this implementation, strings with the highest number of flights are added first in descending order (other lifting strategies are tested in Section 5.6.5.2).

- The weights for all rows are 1 for all superadditive lifting scenarios.

The table shows three salient results. First, the Benders-induced split cuts perform the best overall, reducing runtimes by 3.1% over all scenarios - which is 1.1% better than the second best method (cut-pushing). Second, the acceleration methods perform better on

Table 17: Acceleration Methods on AIR Model: Runtime Performance (CPU seconds)

flow rate restr.	disruption length (min.)	benchmark	cut-pushing (% diff.)	induced split cuts (% diff.)	sequential lifting (% diff.)	relaxed SA lifting (% diff.)
50%	**30 (hub)**	62	65 (+4.8%)	63 (+1.6%)	68 (+9.7%)	69 (+11.3%)
	60 (hub)	1481	1511 (+2.0%)	1408 (-4.9%)	1502 (+1.4%)	1528 (+3.2%)
	75 (hub)	1948	1900 (-3.9%)	1860 (-2.5%)	1924 (-1.2%)	2019 (+3.6%)
	60 (hub) 300 (spoke)	2194	2102 (-4.2%)	2032 (-7.4%)	2149 (-2.1%)	2108 (-3.9%)
75%	**30 (hub)**	64	64 (0.0%)	66 (+3.1%)	68 (+6.3%)	69 (+7.8%)
	60 (hub)	1400	1381 (-1.4%)	1379 (-1.5%)	1445 (+3.2%)	1449 (+3.5%)
	75 (hub)	1856	1810 (-2.5%)	1833 (-1.2%)	1791 (-3.5%)	1897 (+2.2%)
	60 (hub) 300 (spoke)	1947	1891 (-2.9%)	1811 (-7.0%)	1930 (-0.1%)	1942 (0.0%)
100%	**30 (hub)**	106	99 (-6.6%)	100 (-5.7%)	106 (0.0%)	105 (-0.1%)
	60 (hub)	1449	1383 (-4.6%)	1399 (-3.5%)	1508 (+4.1%)	1518 (+4.8%)
	75 (hub)	1860	1828 (-1.7%)	1804 (-3.0%)	1793 (-3.6%)	2007 (+7.9%)
	60 (hub) 300 (spoke)	1462	1415 (-3.2%)	1390 (-4.9%)	1434 (-1.9%)	1396 (-4.5%)
mean difference: (all scenarios)			-2.0%	-3.1%	+1.0%	+3.0%
mean difference: (excluding 30-min scenarios)			-2.5%	-4.0%	-0.4%	+1.9%

more complicated problems in general. By eliminating the smallest 30-minute scenarios, split cuts save 4.0% of total time which may be significant in settings such as an OCC. Third, both sequential and relaxed sequence-independent lifting do not help very much on average. For the case of sequential lifting the reduced time spent as a result of the strengthened inequality is, on average, more than offset in the time associated with solving the lifting problem. For the case of relaxed superadditive lifting, the lifting problem becomes almost trivial to solve; however, it is believed that the coefficients are weak and do not help the net computational effort.

5.6.5.2 Sensitivity Analysis for Lifting Implementations

While Table 17 presents initial results of the methods to strengthen a Benders' cut, some of the implementations are nontrivial as there are certain degrees of freedom in the experiments. This section examines different implementations that seek to enhance the peformance.

Cut-Pushing As previously mentioned, the cut-pushing results from Table 17 are only used for feasibility cuts (typically deriving from the ARM subproblem). However, the method may be applicable to optimality cuts when the RMP is solved as an LP-relaxation, and the associated feasible solution is fractional.

Table 18 shows the results from where an optimality cut is generated after k fractional solutions are found for $k = 5, 10, 20$. The results indicate that adding optimality cuts from the relaxations are, on average, still faster than adding no strengthening, but slower than just considering cut-pushing from feasibility cuts associated with integer solutions to the RMP.

Sequential Lifting Sequential lifting is nontrivial to implement considering it is governed by a lifting rule (for sequential lifting) or aggregation parameters (for relaxed superadditive lifting). Sensitivity is conducted for different rules in the sequential lifting case where it is shown that performance may be improved upon from the initial results presented above.

Because the large number of flight strings, not all strings are required to be present

Table 18: Generating Optimality Cuts from RMP Relaxation: Runtime Performance (CPU seconds)

flow rate restr.	disruption length (min.)	benchmark	optimality cut only	optimality and feasibility cuts		
				$k = 5$	$k = 10$	$k = 20$
			(% diff.)	(% diff.)	(% diff.)	(% diff.)
50%	30 (hub)	62	65 (+4.8%)	65 (+4.8%)	63 (+1.6%)	63 (+1.6%)
	60 (hub)	1481	1511 (+2.0%)	1520 (+2.6%)	1511 (+2.0%)	1508 (+1.8%)
	75 (hub)	1948	1900 (-3.9%)	1941 (-0.4%)	1938 (-0.5%)	1945 (-0.2%)
	60 (hub) 300 (spoke)	2194	2102 (-4.2%)	2186 (-3.6%)	2145 (-2.2%)	2169 (-1.1%)
75%	30 (hub)	64	64 (0.0%)	65 (+1.6%)	66 (+3.1%)	68 (+6.3%)
	60 (hub)	1400	1381 (-1.4%)	1372 (-2.0%)	1403 (+0.2%)	1410 (+0.7%)
	75 (hub)	1856	1810 (-2.5%)	1818 (-2.0%)	1843 (-0.7%)	1839 (-0.9%)
	60 (hub) 300 (spoke)	1947	1891 (-2.9%)	1896 (-2.6%)	1902 (-2.3%)	1926 (-1.1%)
100%	30 (hub)	106	99 (-6.6%)	104 (-1.9%)	106 (0.0%)	103 (-2.8%)
	60 (hub)	1449	1383 (-4.6%)	1390 (-4.1%)	1389 (-4.1%)	1451 (+0.1%)
	75 (hub)	1860	1828 (-1.7%)	1806 (-2.9%)	1800 (-3.2%)	1784 (-4.1%)
	60 (hub) 300 (spoke)	1462	1415 (-3.2%)	1432 (-2.1%)	1444 (-1.2%)	1458 (-0.3%)
	mean difference: (all scenarios)		-2.0%	-1.1%	-0.6%	0.0%
	mean difference: (excluding 30-min scenarios)		-2.5%	-1.9%	-1.3%	-0.6%

in the final inequality. Given a set C_0 of strings whose coefficients are nonzero and a given parameter $p^{\max}$ that specifies the maximum number of lifting problems to be solved, $m = \min\{|I \setminus C_0|, p^{\max}\}$ lifting problems are solved (for all computations $p^{\max} = 500$). The following four strategies were used to determining the sequence of lifting variables:

1. *Maximum Flights per String (default method shown in Table 17)*: As mentioned this method seeks to first lift in string variables with the highest number of flights.

2. *Maintenance Strings*: For all aircraft requiring maintenance within the time window, those strings not already present in the original cut will be lifted first.

3. *Adjacency*: Given the incumbent solution $(x^*_{e,s}, \kappa^*_f)$ that generates the Benders' cut, adjacent solutions are lifted in first that correspond to strings that have the most flights in common with that from those strings $s : x^*_{e,s} = 1$.

4. *Random Lifting*: strings are be lifted in randomly for the m variables lifted.

Table 19 shows the results for the four different policies and shows the result against the benchmark (no strengthening).

The number of flight cancellation variables are far fewer than that of flight strings. Therefore all flights whose coefficient is zero in the original cut will be lifted until a zero coefficient is found. The following three lifting strategies are employed for lifting flight cancellation variables.

1. *Identical Segments*: For all flights f such that $\kappa^*_f = 1$, priority is given to flights with the same origin and departure as f.

2. *Follow-On Segments*: For all flights f such that $\kappa^*_f = 1$, priority is given to all flights scheduled to follow f for same scheduled aircraft through the end of the recovery window.

3. *Random Lifting*: Lift in flight cancellation variables in a random order.

Table 20 shows how the different lifting strategies perform for flight cancellation variables against both the benchmark solution (with no lifting) and the three lifting strategies. The

158

Table 19: Sensitivity for Sequential Lifting of Flight Strings: Runtime Performance (CPU seconds)

flow rate restr.	disruption length (min.)	benchmark	default (max flts) (% diff.)	maint. strings (% diff.)	adjacency (% diff.)	random (% diff.)
50%	30 (hub)	62	68 (+9.7%)	66 (+6.5%)	62 (0.0%)	66 (+6.5%)
	60 (hub)	1481	1502 (+1.4%)	1513 (+2.2%)	1493 (+0.1%)	1524 (+2.9%)
	75 (hub)	1948	1924 (-1.2%)	1960 (+0.1%)	1923 (-1.3%)	2033 (+4.4%)
	60 (hub) 300 (spoke)	2194	2149 (-2.1%)	2137 (-2.6%)	2088 (-4.8%)	2167 (-0.1%)
75%	30 (hub)	64	68 (+6.3%)	60 (-6.3%)	66 (+3.1%)	68 (+6.3%)
	60 (hub)	1400	1445 (+3.2%)	1440 (+2.9%)	1377 (-1.6%)	1461 (+4.4%)
	75 (hub)	1856	1791 (-3.5%)	1819 (-2.0%)	1801 (-3.0%)	1830 (-1.4%)
	60 (hub) 300 (spoke)	1947	1930 (-0.1%)	1911 (-1.8%)	1862 (-4.4%)	1909 (-2.0%)
100%	30 (hub)	106	106 (0.0%)	103 (-2.8%)	107 (0.0%)	109 (+2.8%)
	60 (hub)	1449	1508 (+4.1%)	1486 (+2.6%)	1499 (+3.5%)	1516 (+4.6%)
	75 (hub)	1860	1793 (-3.6%)	1864 (0.0%)	1783 (0.0%)	1870 (+0.5%)
	60 (hub) 300 (spoke)	1462	1434 (-1.9%)	1514 (+3.6%)	1415 (-3.2%)	1403 (-4.0%)
mean difference: (all scenarios)			+1.0%	+0.2%	-1.0%	+2.1%
mean difference: (excluding 30-min scenarios)			-0.4%	+0.6%	-1.6%	+1.0%

159

table shows that for the cases of a single disruption, there is little difference from the benchmark. However, when multiple disruptions exist, then lifting is considerably more valuable in expediting Benders' decomposition.

Relaxed Superadditive Lifting   Aggregation is necessary in order to transform a multirow system into a single constraint in order to utilize sequence independent lifting over the relaxation. A weight vector $\mu \in \mathbb{R}_+^m$ is given as input for which the initial implementation uses a weight of one for all rows. Other weights were used, but the results were nearly identical to that of the benchmark scenario displayed in Table 17.

Table 20: Sensitivity for Sequential Lifting of Flight Cancellations: Runtime Performance (CPU seconds)

flow rate restr.	disruption length (min.)	benchmark	*lifting strings* default (max flts) (% diff.)	*lifting cancellation variables* ident. segs. (% diff.)	follow-on segs. (% diff.)	random (% diff.)
50%	**30 (hub)**	62	68 (+9.7%)	63 (+1.6%)	64 (+3.2%)	67 (+8.1%)
	60 (hub)	1481	1502 (+1.4%)	1485 (+0.3%)	1489 (+0.5%)	1505 (+1.6%)
	75 (hub)	1948	1924 (-1.2%)	1957 (+0.5%)	1940 (-0.4%)	1956 (+0.4%)
	60 (hub) 300 (spoke)	2194	2149 (-2.1%)	2104 (-4.1%)	2122 (-3.3%)	2142 (-2.4%)
75%	**30 (hub)**	64	68 (+6.3%)	64 (0.0%)	67 (+4.7%)	66 (+3.1%)
	60 (hub)	1400	1445 (+3.2%)	1413 (+0.9%)	1449 (+3.5%)	1428 (+2.0%)
	75 (hub)	1856	1791 (-3.5%)	1853 (-0.2%)	1861 (+0.3%)	1852 (-0.2%)
	60 (hub) 300 (spoke)	1947	1930 (-0.1%)	1870 (-4.0%)	1883 (-3.3%)	1910 (-1.9%)
100%	**30 (hub)**	106	106 (0.0%)	109 (+2.8%)	112 (+5.7%)	108 (+1.9%)
	60 (hub)	1449	1508 (+4.1%)	1461 (+0.8%)	1450 (0.0%)	1462 (+0.9%)
	75 (hub)	1860	1793 (-3.6%)	1813 (-2.5%)	1829 (-1.5%)	1839 (-1.1%)
	60 (hub) 300 (spoke)	1462	1434 (-1.9%)	1403 (-4.0%)	1411 (-3.5%)	1424 (-2.6%)
mean difference: (all scenarios)			+1.0%	-0.7%	+0.5%	+0.8%
mean difference: (excluding 30-min scenarios)			-0.4%	-1.4%	-0.9%	-0.4%

161

CHAPTER VI

CONCLUSIONS AND FUTURE RESEARCH

6.1 Conclusions

There is perhaps no single industry that has benefited more from advancements made in operations research than aviation. From an airline's perspective, these tools have been successfully employed to help airlines to design their flight network, assign their fleets, schedule their aircraft and crew resources, price their fares, utilize their airport resources, and a host of other well-known paradigms. One common characteristic these paradigms share is they take place well in advance of operations, sometimes over a year in advance relative to the departure of a given flight. Such problems present in the *planning* phase of airline operations are assumed to be frictionless. However, as discussed in Chapter 2, disruptions are common that render the necessity of re-scheduling various resources. As the growth of the demand of air transportation is expected to outgrow that of airspace and airport capacities worldwide, delays and cancellations are anticipated to induce further difficulties for airlines.

The recovery process involves four primary classes of re-scheduling decisions pertaining to the flight schedule, aircraft routings, crew schedules, and passenger itineraries. Such scheduling decisions, made in the *operational* environment of an airline's Operational Control Center (OCC), are faced with greater complexities relative to its planning counterpart. First, decisions are required to be delivered in as close to real-time as possible. Second, operational regulations stemming from regulations mandated by the FAA, unions, and airline agreements, often make finding a feasible solution alone an arduous process.

While there has been a relatively small body of research aimed at studying mathematical programming techniques to solve the airline recovery problem, most studies have proposed doing so by a sequential approach. While this may be computationally tractable, its practical use within an OCC is spurious. Integrating the four classes of re-scheduling decisions

overcomes most barriers, but thought to be too computationally burdensome for its own practical use. In order to incorporate integrated recovery into the constraints imposed by the real-time decision making environment of an OCC, some blend of optimization and heuristic methods are necessary. Chapter 3 has studied one possible mechanism that optimizes the four class of problems together via Benders' decomposition after the problem size has been reduced by heuristic methods. Results verify that the integrated solution delivers (sometimes substantial) improvements over the traditional sequential approach. Moreover, it has been shown that the runtimes associated with solving a one-day problem may be viable for an OCC.

Computational effort is central to the efficacy of real-time decision making environments like that of an OCC, so improving the performance of traditional algorithm have important consequences. Chapters 4 and 5 introduce two mechanisms that are used to solve the larger AIR problem, and may be applied to other problems exhibiting a similar structure. The AIR model relies on both column generation and constraint generation. While each of these classical techniques to solve large-scale optimization problems are well-known in practice, little has been studied in using these two methods concurrently. Chapter 4 examines one such possibility that relies on an the Theorem of the Alternative to determine whether or not the Benders' cut remains valid over those columns that have not priced out. By saving much effort associated with generating paths through the crew duty network, results have shown that the certificate-based method reduces computational effort from the standard approach that makes redundant calls to the pricing out phase. A discrete version was also discussed whose certificate is based on the integer analog to the Theorem of the Alternative.

The core algorithm employed in the solution presented in Chapter 3 is Benders' decomposition, which is a pragmatic approach to solve large-scale optimization problems whose constraint matrix is represented by a block diagonal structure. While this method has been successfully employed to solving ubiquitous problems arising in mixed-integer programming, stochastic programming, and transportation, it has been seen to exhibit slow convergence in some settings. A body of literature exists to circumvent such shortcomings. However, these studies often assume a certain structure for the underlying problems. Chapter 5 has

163

explored ways in which the well-known Benders' decomposition algorithm may be expedited by strengthening a standard Benders' feasibility or optimality cut. Four methods were proposed. The first considered pushing a non-binding cut into the interior of the feasible set associated with the RMP until it was binding with a feasible point. The second considered a split disjunction from a Benders' cut in which a split cut could be generated from the Cut Generating Linear Program (CGLP). The third suggested lifting in variables whose component in the dual extreme point or extreme ray was zero in a standard, sequential manner. To overcome the dependency from the lifting sequence, the RMP was relaxed by a method of row aggregation. Lifting is then either superadditive, or may be approximated by a valid superadditive function. By testing the approaches within the context of the AIR problem, the first two methods were of particular use in reducing the computational effort.

6.2   Future Work

A number of alternative approaches and extensions arise from the conclusions of this work. The following list some of the more salient features that may lead to promising future research within this domain.

6.2.1   Extensions and Alternative Approaches to AIR

This thesis presents a methodology to solve a problem for integrated airline recovery. Problems of such size and complexity often involve as much inexact procedures as more scientifically rigorous ones. Consequentially, the approach undertaken by this body of work represents only one of a multitude of possibilities. The following possible extensions may expedite the optimization process making an optimization-based approach to AIR more tractable.

Recovering only duties in crew recovery   The Crew Recovery Model (CRM) presented in Chapter 3 has sought to assign crew members to modified crew pairings that delivers each disrupted crew member to the station in which they are required to be present at by the end of the time window. While this replicates the practice of airline operations, it may be more beneficial to assign crew members only to duties that are contained within

the time window and concatenate the duties to pairings in a postprocessing stage. This would eliminate the need to generate pairings that correspond to paths over a crew duty network that may be dense, particularly for regional carriers.

Incorporating uncertain Event times   While the practice of disruption management differs across carriers, one trait in common to all OCCs is the uncertainty of event times. For recovery scenarios that are caused by weather disruptions, the times associated with the impact on operations may change on a frequent basis. For airlines that use recovery pro-actively well into the future, an optimized solution to be be invalid at the beginning of its operational irregularity. Incorporating uncertainty through a stochastic process or scenario tree analysis may improve on the applicability of the proposed model.

Endogenous Time Window   Because disruptable flights, aircraft, crews, and passengers are a function of time, the time window plays an important role in the computational effort to deliver an sequential or integrated solution. This window is exogenous to the optimization model and is defined by the user. However, this may be spurious considering it may limit the feasibility of a solution (if the window is relatively short), or induce longer than needed runtimes (if the window is relatively long). Having an exact or inexact method to where the time window is endogenous so as to mitigate such errors may improve the quality of the system.

Updates of Crew Duty Network   Building, managing, and generating paths through the crew duty network is likely to be a tedious effort for most large disruptions. Therefore initial efforts to solve the AIR model have been over a single-day horizon where it is plausible to rebuild the network over each iteration. However, rebuilding the network for a time window that spans multiple days will preclude its use in practice. Alternate methods are therefore required for larger scenarios to be employed in an OCC environment.

One possibility is to generate the crew duty network once before the iterative optimization process based off the original schedule, and update the network before the CRM is solved as changes to duties and eligible connections between them arise from the new

scheduling decisions. Algorithm 10 demonstrates such an idea. The key is determining how the network is to update, which is nontrivial and would require additional analysis.

Algorithm 10 Integrating SRM and CRM through Approximations to Crew Duty Network

1: **given:** original flight schedule and all data
2: generate all duties $\mathcal{D}$ based off the original operating schedule
3: generate arc set $\mathcal{A}$ which defines all legal connections between duties
4: generate the crew duty network $\mathcal{G} = (\mathcal{D}, \mathcal{A})$
5: initialize $i = 1$
6: **while** optimal AIR solution has not been found **do**
7:     let $\left(x_{e,s}^*, \kappa_f^*, \eta_{\mathrm{CRM}}^*, \eta_{\mathrm{IRM}}^*\right)$ denote a solution to the SRM (Benders' RMP)
8:     let $\mathcal{D}^{\mathrm{new}}$ denote newly generated duties from $x_{e,s}^*$
9:     let $\mathcal{D}^{\mathrm{illegal}}$ denote the duties that are no longer legal given $(x_{e,s}^*, \kappa_f^*)$
10:     let $\mathcal{A}_1^{\mathrm{illegal}} = \left\{a_{ij} \in \mathcal{A} : i \in \mathcal{D}^{\mathrm{illegal}} \text{ or } j \in \mathcal{D}^{\mathrm{illegal}}\right\}$
11:     let $\mathcal{A}_2^{\mathrm{illegal}} = \left\{a_{ij} \in \mathcal{A} : \text{ duty } i \text{ and duty } j \text{ are no longer legal given } x_{e,s}^*\right\}$
12:     let $\mathcal{A}^{\mathrm{illegal}} = \mathcal{A}_1^{\mathrm{illegal}} \cup \mathcal{A}_2^{\mathrm{illegal}}$ denote illegal arcs for new solution
13:     let $\mathcal{A}^{\mathrm{new}}$ denote the set of new legal arcs given $x_{e,s}^*$
14:     approximate new crew duty network by $\mathcal{G}^i = \left(\mathcal{D}^i, \mathcal{A}^i\right)$ where:

$$\mathcal{D}^i = \mathcal{D} \setminus \mathcal{D}^{\mathrm{illegal}} \cup \mathcal{D}^{\mathrm{new}} \quad \text{and}$$
$$\mathcal{A}^i = \mathcal{A} \setminus \mathcal{A}^{\mathrm{illegal}} \cup \mathcal{A}^{\mathrm{new}}$$

15:     **for** all crew $k \in K$ **do**
16:         generate $s - t$ paths over $\mathcal{G}^i$ and let P_k' denote the set of pairings
17:     **end for**
18:     solve CRM given $\cup_{k \in K} P_k'$
19:     **if** solution is not optimal **then**
20:         price out new pairings $P_k^{\mathrm{new}} \subset P_k \setminus P_k'$
21:         generate candidate Benders' cut
22:         **if** cut shown to be valid over all P_k **then**
23:             add cut to RMP and update new columns
24:         **else**
25:             do not add cut, add new columns and re-solve CRM
26:         **end if**
27:     **end if**
28:     $i \leftarrow i + 1$
29: **end while**

6.2.2 Computational Enhancements

In addition to analyzing and solving the Airline Integrated Recovery problem, another contribution of this thesis has been examining ways in which computational advancements may accelerate some classes of optimization problems. Given this exposition, a number of

166

open questions are raised for future research.

6.2.2.1   Simultaneous Cut and Column Generation

Chapter 4 has studied an alternative method where column generation and cut generation may be applied in a simultaneous manner. There are two areas of interest that may be useful to such large-scale optimization problems whose problem structure is amenable to *both* constraint and column generation.

Comparison with Existing Methods   The proposed approach relied on using information obtained from the pricing problem and determining whether or not a solution could be obtained for the *alternate* polyhedron which is obtained by invoking the Theorem of the Alternative. If a subproblem is infeasible or suboptimal over a proper subset of local variables and certain conditions are met in the pricing problem, then a certificate of infeasibility or suboptimality is found over all subproblem variables and the candidate cut may be added. If the conditions are not met, then columns continue to price out until either the certificate is found, or no other columns price out.

While relatively little work has been done in this general area, there are two connections of interest. One is recent work by Fischetti [52] in which it is shown that the method above has a strong connection with (Theorem 4.4.5). However, there are other studies in which illustrate a different idea that first relies on a reformulation of a master problem such that the pricing problem becomes independent of the original master problem in a way the pricing problem and cuts to the new problem are independent. Further analysis and computational experiments would provide valuable insight into the strengths and weaknesses of each approach.

Generalization to Mixed-Integer Programming   The proposed method was shown to work well on a class of problems that are solved by Benders' decomposition where the Restricted Master Problem (RMP) is solved as an MIP problem, all subproblems were continuous, and columns are being generated from the variables of one or more subproblem. Another method was proposed to a more generic combinatorial optimization problem

167

involving edge coloring (Section 4.4.2). An analogous version of the Theorem of the Alternative, applied to discrete systems (Theorem 4.4.10), was used in a manner similar to the continuous version. However, the methodology relied on pricing out *all* columns whose reduced cost was negative, which caused the runtimes to increase dramatically (although fewer iterations were required). It would be of interest to either devise a better algorithm to construct a solution, or to tighten the discrete alternative system in an equivalent manner.

Perhaps of even greater interest would be to lay a theoretical basis for the connection to the above method and that of a generic branch-and-bound-and-price framework from Barnhart et al. [21]. If the traditional methodology could be strengthened from the certificate-based approach, there could be great implications to a wide range of problem employing such methods in practice.

6.2.2.2 *Further Strengthening Techniques to Benders' Cuts*

While Chapter 5 sought to accelerate Benders' decomposition by strengthening the standard Benders' inequality, three classes of further analysis may be of interest.

Other Cut Strengthening Procedures The methods proposed to accelerate Benders' decomposition in Chapter 5 are just four possibilities of other methods thought to accelerate Benders' decomposition through cut strengthening. Other polyhedral methods may be employed, such as examining lift-and-project cuts for an RMP whose integer variables are binary (see Balas [12], Balas et al. [14], and Balas and Perregaard [15]), or some possible work in two-row cuts.

Optimizing over the Split Closure The results from the AIR problem show that split cuts generated from a Benders' disjunction performed the best on average. However the Benders' disjunction is just one class of cuts that may be used. Some analysis may be of interest on why the Benders-induced split is strong relative to other classes of split cuts. Perhaps of greater interest would be to optimize over the split closure of the RMP, which optimizes over *all* possible split disjunctions. While the separation problem amounts to a nonlinear programming problem, a reformulation into a parametric mixed integer linear

programming problem is possible (see Balas and Saxena [16]). Because of the computational burden of optimizing over all possible split disjunction, this was not considered in this thesis, but testing the method might show the procedure is tractable.

Tighter Relaxations to Superadditive Lifting Approximations   The superadditive lifting over the row aggregation was not shown to be effective in the context of the AIR model as the generated coefficients were typically weak. Whether the poor results were attributable to the structure of the AIR mode (perhaps due to too many rows of the RMP), the aggregation used (poor choice of weights for the aggregation), or the overall method are unclear. Further analysis may show the method to dominate the sequential lifting, and may be of use for other applications.

REFERENCES

[1] "Airbus global market forecast, 2010–2029," December 2010. www.airbus.com/company/market/gmf2010/.

[2] "Boeing current market outlook, 2010–2029," 2010. www.boeing.com/cmo.

[3] ABARA, J., "Applying integer linear programming to the fleet assignment problem," *Interfaces*, vol. 19, no. 4, pp. 20–28, 1989.

[4] ACUNA-AGOST, R., FEILLET, D., MICHELON, P., and GUEYE, S., "Rescheduling flights, aircraft, and passengers simultaneously under disrupted operations - a mathematical programming approach based on statistical analysis." submitted to AGIFORS Anna Valicek Medal, 2009.

[5] AGEEVA, Y., "Approaches to incorporating robustness into airline scheduling," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, August 2000.

[6] AHMADBEYGI, S., COHN, A., and LAPP, M., "Decreasing airline delay propagation by re-allocating scheduled slack," *IIE Transactions*, vol. 42, pp. 478–489, 2010.

[7] AHUJA, R., MAGNANTI, T., and ORLIN, J., *Network Flows: Theory, Algorithms, and Applications*. Upper Saddle River: Prentice Hall, 1993.

[8] ALVES, C. and VALÉRIO DE CARVALHO, J., "A stabilized branch-and-price-and-cut algorithm for the multiple length cutting stock problem," *Computers and Operations Research*, vol. 35, pp. 1315–1328, 2008.

[9] ANBIL, R., TANGA, R., and JOHNSON, E., "A global approach to crew pairing optimization," *IBM Systems Journal*, vol. 31, no. 1, pp. 71–78, 1992.

[10] ARGÜELLO, M., BARD, J., and YU, G., "A GRASP for aircraft routing in response to groundings and delays," *Journal of Combinatorial Optimization*, vol. 5, no. 1, pp. 211–228, 1997.

[11] ATAMTÜRK, A., "Polyhedral methods in discrete optimization," in *Proceedings of Symposia in Applied Mathematics* (HOSTEN, S., LEE, J., and THOMAS, R., eds.), vol. 61, pp. 21–37, American Mathematical Society, 2004.

[12] BALAS, E., "Disjunctive programming and a hierarchy of relaxations for discrete optimization problems," *SIAM Journal on Algebraic and Discrete Methods*, vol. 6, pp. 466–486, 1985.

[13] BALAS, E., CERIA, S., and CORNUÉJOLS, G., "Mixed 0–1 programming by lift-and-project in a branch-and-cut framework," *Management Science*, vol. 42, pp. 1229–1246, 1996.

[14] BALAS, E., CERIA, S., and CORNUÉJOLS, G., "A lift-and-project cutting plane algorithm for mixed 0-1 programs," *Mathematical Programming*, vol. 58, pp. 295–324, 1993.

[15] BALAS, E. and PERREGAARD, M., "A precise correspondence between lift-and-project cuts, simple disjunctive cuts and mixed integer Gomory cuts for 0-1 programming," *Mathematical Programming B*, vol. 94, pp. 221–245, 2003.

[16] BALAS, E. and SAXENA, A., "Optimizing over the split closure," *Mathematical Programming A*, 2008.

[17] BALL, M., BARNHART, C., NEMHAUSER, G., and ODONI, A., "Air transportation: Irregular operations and control," in *Transportation Volume 14: Handbook of Operations Research and Management Science* (BARNHART, C. and LAPORTE, G., eds.), Amsterdam: Elsevier, 2007.

[18] BALL, M., BARNHART, C., DRESNER, M., HANSEN, M., NEELS, K., ODONI, A., PETERSON, E., SHERRY, L., TRANI, A., and ZOU, B., "Total delay impact study: A comprehensive assessment of the costs and impacts of flight delay in the united states," tech. rep., Federal Aviation Administration, November 2010. revised release.

[19] BARNHART, C., BOLAND, N., CLARKE, L., JOHNSON, E., and NEMHAUSER, G., "Flight string models for aircraft fleeting and routing," *Transportation Science*, vol. 32, no. 3, pp. 208–220, 1998.

[20] BARNHART, C., BOLAND, N., CLARKE, L. W., JOHNSON, E. L., and NEMHAUSER, G. L., "Flight string models for aircraft fleeting and routing," *Transportation Science*, vol. 32, pp. 208–220, 1998.

[21] BARNHART, C., HANE, C. A., and VANCE, P. H., "Robust branch-and-cut-and-price for the capacitated vehicle routing problem," *Operations Research*, vol. 48, no. 2, pp. 318–326, 2000.

[22] BARNHART, C., JOHNSON, E. L., NEMHAUSER, G. L., SAVELSBERGH, M. W., and VANCE, P. H., "Branch-and-price: Column generation for solving huge integer programs," *Operations Research*, vol. 46, no. 3, pp. 316–329, 1998.

[23] BARNHART, C., KNIKER, T. S., and LOHATEPANONT, M., "Itinerary-based airline fleet assignment," *Transportation Science*, vol. 36, no. 2, pp. 199–217, 2002.

[24] BARNHART, C., LU, F., and SHENOI, R., "Integrated airline schedule planning," in *Operations Research in the Airline Industry* (YU, G., ed.), Kluwer's International Series, ch. 13, pp. 384–403, Austin, TX: Kluwer Academic Publisher, 1998.

[25] BELOBABA, P. P., *Air Travel Demand and Airline Seat Inventory Management*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1987.

[26] BENDERS, J. F., "Partitioning procedures for solving mixed-variable programming problems," *Numerische Mathematik*, vol. 4, no. 1, pp. 238–252, 1962.

[27] BISAILLON, S., CORDEAU, J.-F., LAPORTE, G., and PASIN, F., "A large neighbourhood search heuristic for the aircraft and passenger recovery problem." http://neumann.hec.ca/chairelogistique/jfc.shtml, April 2010.

[28] BONNEFOY, P., *Development of Multi-Airport Systems: A Worldwide Approach*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2008.

[29] BOWMAN, V. and NEMHAUSER, G. L., "Deep cuts in integer programming," *Opsearch*, vol. 8, pp. 89–111, 1981.

[30] BRATU, S. and BARNHART, C., "Flight operations recovery: New approaches considering passenger recovery," *Journal of Scheduling*, vol. 9, pp. 279–298, 2006.

[31] BRATU, S. and BARNHART, C., "An analysis of passenger delays using flight operations and passenger booking data," *Journal of Transportation and Statistics*, vol. 13, no. 1, pp. 1–27.

[32] BURKE, E. K., DE CAUSMAECKER, P., DE MAERE, G., MULDER, J., PAELINCK, M., and VANDEN BERGHE, G., "A multi-objective approach for robust airline scheduling," *Computers and Operations Research*, vol. 37, no. 5, pp. 822–832, 2010.

[33] CAO, J. and KANAFANI, A., "Real-time decision support for integration of airline flight cancellations and delays, part I: Mathematical formulation," *Transportation Planning and Technology*, vol. 20, pp. 189–199, 1997.

[34] CAO, J. and KANAFANI, A., "Real-time decision support for integration of airline flight cancellations and delays, part II: Algorithm and computational experiments," *Transportation Planning and Technology*, vol. 20, pp. 201–217, 1997.

[35] CHRISTENSEN, U. G. and PEDERSEN, A. B., "Lecture note on Benders decomposition." `http://www.andersbp.dk/studier/dat/OPPP/benders.pdf`, October 2008. accessed January 2012.

[36] CLARKE, L. W., HANE, C. A., JOHNSON, E. L., and NEMHAUSER, G. L., "Maintenance and crew consideration in fleet assignment," *Transportation Science*, vol. 30, no. 3, pp. 249–260, 1996.

[37] CLARKE, M. D., "Irregular airline operations: A review of the state-of-the-art practice in airline operations control center," *Journal of Air Transport Management*, vol. 4, no. 2, pp. 67–76, 1998.

[38] CLARKE, M. D., *Development of Heuristic Procedures for Flight Rescheduling in the Aftermath of Irregular Airline Operations*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, February 1998.

[39] CLAUSEN, J., LARSEN, A., LARSEN, J., and REZANOVA, N., "Disruption management in the airline industry - concepts, models and methods," *Computers and Operations Research*, vol. 37, pp. 809–821, 2010.

[40] CODATO, G. and FISCHETTI, M., "Combinatorial Benders' cuts for mixed-integer linear programming," *Operations Research*, vol. 54, no. 4, pp. 756–766, 2006.

[41] COHN, A. and BARNHART, C., "Improving crew scheduling by incorporating key maintenance routing decisions," *Operations Research*, vol. 51, no. 3, pp. 387–396, 2003.

[42] CONFORTI, M., CORNUÉJOLS, G., and ZAMBELLI, G., "Polyhedral approaches to mixed integer linear programming," in *50 Years of Integer Programming: 1958–2008* (JÜNGER, M., LIEBLING, T. M., NADDEF, D., NEMHAUSER, G., PULLEYBLANK, W. R., REINELT, G., RINALDI, G., and WOLSEY, L., eds.), ch. 11, pp. 343–386, Springer, 2010.

[43] COOK, A., TANNER, G., WILLIAMS, V., and MEISE, G., "Dynamic cost indexing - managing airline delay costs," *Journal of Air Transport Management*, vol. 15, pp. 26–35, 2009.

[44] COOK, W., KANNAN, R., and SHRIJVER, A., "Chvátal closures for mixed integer programming problems," *Mathematical Programming*, vol. 47, pp. 155–174, 1990.

[45] CORDEAU, J.-F., STOJKOVIĆ, G., SOUMIS, F., and DEROSIERS, J., "Benders decomposition for simultaneous aircraft routing and crew scheduling," *Transportation Science*, vol. 35, no. 4, pp. 375–388, 2001.

[46] CÔTÉ, G. and LAUGHTON, A., "Large-scale mixed integer programming: Benders-type heuristics," *European Journal of Operational Research*, vol. 16, no. 3, pp. 327–333, 1984.

[47] DESAULNIERS, G., DESROSIERS, J., DUMAS, Y., SOLOMAN, M. M., and SOUMIS, F., "Daily aircraft routing and scheduling," *Management Science*, vol. 43, no. 6, pp. 847–855, 1997.

[48] EGGENBERG, N., BIERLAIRE, M., and SALANI, M., "A column generation algorithm for disrupted airline schedules," tech. rep., École Polytechnique Federale de Lausanne, 2007.

[49] FEILLET, D., GENDREAU, M., MEDAGLIA, A., and WALTEROS, J., "A note on branch-and-cut-and-price," *Operations Research Letters*, vol. 38, pp. 346–353, 2010.

[50] FILAR, J., MANYEM, P., and WHITE, K., "How airlines and airports recover from schedule perturbations: A survey," *Annals of Operations Ressearch*, vol. 108, pp. 315–333, 2001.

[51] FISCHETTI, M. and LODI, A., "Local branching," *Mathematical Programming*, vol. 98, no. 1–3, pp. 23–47, 2003.

[52] FISCHETTI, M., SALVAGNIN, D., and ZANETTE, A., "A note on the selection of Benders' cuts," *Mathematical Programming series B*, vol. 124, pp. 175–182, 2010.

[53] FUKASAWA, R., LONGO, H., LYSGAARD, J., DE ARAGAO, M., REIS, M., UCHOA, E., and WERNECK, R., "Robust branch-and-cut-and-price for the capacitated vehicle routing problem," *Mathematical Programming*, vol. 106, pp. 497–511, 2006.

[54] GABTENI, S., "Roadef competition winners methods review," May 2009. Agifors Airline Operations Meeting, Montreal.

[55] GAO, C., *Airline Integrated Planning and Operations*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, August 2007.

[56] GAO, C., JOHNSON, E. L., and SMITH, B. C., "Integrated airline fleet assignment and robust crew planning," *Transportation Science*, 2009.

[57] GEOFFRION, A., "Generalized Benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, no. 4, pp. 237–260, 1972.

[58] GLEESON, J. and RYAN, J., "Identifying minimally infeasible subsytems of inequalities," *ORSA Journal on Computing*, vol. 2, no. 1, pp. 61–63, 1990.

[59] GOMORY, R. E., "Some polyhedra related to combinatorial problems," *Linear Algebra and Its Applications*, vol. 2, pp. 451–558, 1969.

[60] GOYCOOLEA, M., "A heuristic for generating violated GMI cuts from non-optimal bases." INFORMS Annual Meeting, Austin, TX, 2010.

[61] GRANDEAU, S. C., CLARKE, M. D., and MATHAISEL, D. F., "The processes of airline system operations control," in *Operations Research in the Airline Industry* (YU, G., ed.), ch. 11, pp. 312–369, Austin, TX: Kluwer Academic Publishers, 1998.

[62] GU, Z., NEMHAUSER, G. L., and SAVELSBERGH, M. W., "Sequence independent lifting in mixed integer programming," *Journal of Combinatorial Optimization*, vol. 4, no. 1, pp. 109–129, 2000.

[63] HANE, C. A., BARNHART, C., JOHNSON, E. L., MARSTEN, R. E., NEMHAUSER, G. L., and SIGISMONDI, G., "The fleet assignment problem: Solving a large-scale integer program," *Mathematical Programming*, vol. 70, pp. 211–232, 1995.

[64] HOFFMAN, K. L. and PADBERG, M., "Solving airline crew scheduling problems by branch-and-cut," *Management Science*, vol. 39, no. 6, pp. 657–682, 1993.

[65] HOLMBERG, K., "On using approximations of the Benders master problem," *European Journal of Operational Research*, vol. 77, no. 1, pp. 111–125, 1994.

[66] HOLYER, I., "The NP-completeness of edge coloring," *SIAM Journal of Computing*, vol. 10, pp. 718–770, 1981.

[67] HOOKER, J. N., *Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction*. New York: John Wiley and Sons, 2000.

[68] JACOBS, T. L., SMITH, B. C., and JOHNSON, E. L., "Incorporating network flow effects into the airline fleet assignment process," *Transportation Science*, vol. 42, no. 4, pp. 514–529, 2008.

[69] JARRAH, A., YU, G., KRISHNAMURTHY, N., and RAKSHIT, A., "A decision support framework for airline flight cancellations and delays," *Transportation Science*, vol. 27, no. 3, pp. 266–280, 1993.

[70] JENKINS, D. and MARKS, J., "The tarmac rule: Adjustments needed," March 2011.

[71] JOINT ECONOMIC COMMITTEE, "Your flight has been delayed again: Flight delays cost passengers, airlines, and the u.s. economy billions," May 2008.

[72] K.F. ABDELGHANY, A. A. and EKOLLU, G., "An integrated decision support tool for airlines schedule recovery during irregular operations," *European Journal of Operational Research*, vol. 185, no. 2, pp. 825–848, 2008.

[73] KLABJAN, D., JOHNSON, E. L., NEMHAUSER, G. L., GELMAN, E., and RAMASWAMY, S., "Airline crew scheduling with time windows and and plane-count constraints," *Transportation Science*, vol. 36, no. 3, pp. 337–348, 2002.

[74] KOHL, N., LARSEN, A., LARSEN, J., ROSS, A., and TIOURINE, S., "Airline disruption management - perspectives, experiences and outlook," *Journal of Air Transport Management*, vol. 13, no. 3, pp. 149–162, 2007.

[75] LAN, S., CLARKE, J.-P., and BARNHART, C., "Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions," *Transportation Science*, vol. 40, pp. 15–28, February 2006.

[76] LEE, J. and LEUNG, J., "A comparison of two edge-coloring formulations," *Operations Research Letters*, vol. 13, pp. 215–223, 1993.

[77] LETTOVSKY, L., *Airline Operations Recover: An Optimization Approach*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 1997.

[78] LETTOVSKY, L., JOHNSON, E., and NEMHAUSER, G., "Airline crew recovery," *Transportation Science*, vol. 34, no. 4, pp. 337–348, 2000.

[79] LOHATEPANONT, M., *Airline Fleet Assignment and Schedule Design: Integrated Models and Algorithms*. PhD thesis, Massachusetts Institute of Technology, 2001.

[80] MAGNANTI, T. L. and WONG, R., "Accelerating benders decomposition: Algorithmic enhancements and model selection criteria," *Operations Research*, vol. 29, no. 3, pp. 464–484, 1981.

[81] MARCHAND, H. and WOLSEY, L. A., "The 0-1 knapsack problem with a single continuous variable," *Mathematical Programming*, vol. 85, pp. 15–33, 1999.

[82] MARKS, J., "Operational implications of expanded U.S. passenger protections." AGIFORS 2011 Operations Conference, London, May 2011.

[83] McDANIEL, D. and DEVINE, M., "A modified Benders' partitioning algorithm for mixed integer programming," *Management Science*, vol. 24, no. 3, pp. 312–319, 1977.

[84] McGILL, J. I. and VAN RYZIN, G. J., "Revenue management: Research overview and prospects," *Transportation Science*, vol. 33, no. 2, pp. 233–256, 1999.

[85] MERCIER, A., CORDEAU, J.-F., and SOUMIS, F., "A computational study of benders decomposition for the integrated aircraft routing and crew scheduling problem," *Computers and Operations Research*, vol. 32, no. 6, pp. 1451–1476, 2005.

[86] NARISETTY, A. K., NEMHAUSER, G. L., and RICHARD, J.-P., "Lifted tableaux inequalities for 01 mixed-integer programs: A computational study," *INFORMS Journal on Computing*, vol. 23, no. 3, pp. 416–424, 2011.

[87] NEMHAUSER, G. and PARK, S., "A polyhedral approach to edge coloring," *Operations Research Letters*, vol. 10, pp. 315–322, 1991.

[88] NEMHAUSER, G. and WOLSEY, L., *Integer and Combinatorial Optimization.* New York: John Wiley & Sons, 1999.

[89] NISEEN, R. and HAASE, K., "Duty-period-based network model for crew rescheduling in European airlines," *Journal of Scheduling*, vol. 9, no. 3, pp. 255–278, 2006.

[90] PADBERG, M. W., "On the facial structure of set packing polyhedra," *Mathematical Programming*, vol. 5, pp. 199–215, 1973.

[91] PALPANT, M., BOUDIA, M., ROBELIN, C.-A., GABTENI, S., and LABURTHE, F., "Roadef 2009 challenge: Disruption management for commercial aviation," 2009. http://challenge.roadef.org/2009.

[92] POGGI DE ARAGÃO, M. and UCHOA, E., "Integer program reformulation for robust branch-and-cut-and-price," in *Annals of Mathematical Programming in Rio de Janerio*, (Búzios, Brazil), pp. 56–61, 2003.

[93] PUJET, N. and FERON, E., "Modeling and airline operations control." Second USA/Europe Air Traffic Management R&D Seminar, December 1998. Orlando, FL.

[94] REI, W., CORDEAU, J.-F., GENDREAU, M., and SORIANO, P., "Accelerating Benders decomposition by local branching," *INFORMS Journal on Computing*, vol. 21, no. 2, pp. 333–345, 2009.

[95] RICHARD, J.-P., DE FARIAS, I., and NEMHAUSER, G. L., "Lifted inequalities for 0-1 mixed integer programming: Basic theory and algorithms," *Mathematical Programming*, vol. 98, pp. 89–113, 2003.

[96] RICHARD, J.-P., DE FARIAS, I., and NEMHAUSER, G. L., "Lifted inequalities for 0-1 mixed integer programming: Superlinear lifting," *Mathematical Programming*, vol. 98, pp. 115–143, 2003.

[97] ROSENBERGER, J., JOHNSON, E., and NEMHAUSER, G., "Rerouting aircraft for airline recovery," *Transportation Science*, vol. 37, no. 4, pp. 408–421, 2003.

[98] ROSENBERGER, J., JOHNSON, E., and NEMHAUSER, G., "A robust fleet-assignment model with hub isolation and short cycles," *Transportation Science*, vol. 38, no. 3, pp. 357–368, 2004.

[99] RYAN, D. and FALKNER, J., "A bus crew scheduling system using a set partitioning model," *Asia Pacific Journal of Operational Research*, vol. 4, pp. 39–56, 1997.

[100] SANDHU, R. and KLABJAN, D., "Integrated airline fleeting and crew-pairing decisions," *Operations Research*, vol. 55, no. 3, pp. 439–456, 2007.

[101] SCHELLEKENS, B. and OMONDI, T., "Drivers behind on-time departure performance: A case for Kenya Airways." AGIFORS 2011 Operations Conference, London, May 2011.

[102] SCHRIJVER, A., *Theory of Linear and Integer Programming.* John Wiley and Sons, 1986.

[103] SEYMOUR, P., "On multicoloring of cubic graphs and conjectures of fulkerson and tutte," *Proc. London Math. Soc.*, vol. 38, pp. 423–460, 1979.

[104] SHAW, T. L., *Hybrid Column Generation for Large Network Routing Problems: with Implementations in Airline Crew Scheduling.* PhD thesis, Georgia Institute of Technology, Atlanta, GA, April 2003.

[105] SHEBALOV, S. and KLABJAN, D., "Robust airline crew pairing: Move-up crews," *Transportation Science*, vol. 40, no. 3, pp. 300–312, 2006.

[106] SMITH, B., *Robust Airline Fleet Assignment.* PhD thesis, Georgia Institute of Technology, Atlanta, GA, 2004.

[107] SMITH, B. and JOHNSON, E., "Robust airline fleet assignment: Imposting station purity using station decomposition," *Transportation Science*, vol. 40, no. 4, pp. 497–516, 2006.

[108] SONG, M., WEI, G., and YU, G., "A decision support framework for crew management during airline irregular operations," in *Operations Research in the Airline Industry* (YU, G., ed.), ch. 1, pp. 68–100, Austin, TX: Kluwer Academic Publishers, 1998.

[109] SOUMIS, F., FERLAND, J. A., and ROUSSEAU, J.-M., "A model for large-scale aircraft routing and scheduling problems," *Transportation Research Part B: Methodological*, 1980.

[110] STAHL, S., "Fractional edge coloring," *Cashiers CEntre Etudes Rech. Oper.*, vol. 21, pp. 127–131, 1979.

[111] STOJKOVIĆ, G., SOUMIS, F., DESROSIERS, J., and SOLOMON, M., "An optimization model for a real-time flight scheduling problem," *Transportation Research A: Policy and Practice*, vol. 36, pp. 779–788, 2002.

[112] STOJKOVIĆ, M. and SOUMIS, F., "An optimization model for the simultaneous operational flight and pilot scheduling problem," *Management Science*, vol. 47, no. 9, pp. 1290–1305, 2001.

[113] STOJKOVIĆ, M., SOUMIS, F., and DESROSIERS, J., "The operational airline crew scheduling problem," *Transportation Science*, vol. 32, no. 3, pp. 232–245, 1998.

[114] TALLURI, K., "The four-day aircraft maintenance routing problem," *Transportation Science*, 1998.

[115] TALLURI, K. T. and VAN RYZIN, G. J., *The Theory and Practice of Revenue Management.* Norwell, MA: Kluwer Academic Publishers, 2004.

[116] TEODOROVIC, D. and GUBERINIC, G., "Optimal dispatching strategy on an airline network after a schedule perturbation," *European Jouranl of Operational Research*, vol. 15, pp. 178–182, 1984.

[117] TEODOROVIC, D. and STOJKOVIĆ, G., "Model for operational daily airline scheduling," *Transportation Planning and Technology*, vol. 14, no. 4, pp. 273–285, 1990.

[118] TEODOROVIC, D. and STOJKOVIĆ, G., "Model to reduce airline schedule disturbances," *Journal of Transportation Engineering*, vol. 121, no. 4, pp. 324–331, 1995.

[119] Thengvall, B., Gang, Y., and Bard, J., "Multiple fleet aircraft schedule recovery following hub closures," *Transportation Research A: Policy and Practice*, vol. 35, pp. 289–308, 2001.

[120] U.S. Department of Transportation Bureau of Transportation Statistics, "Airline on-time data." www.bts.gov.

[121] Valério de Carvalho, J. M., "Using extra dual cuts to accelerate column generation," *INFORMS Journal on Computing*, vol. 17, pp. 175–182, 2005.

[122] Van Roy, T., "Cross decomposition for mixed integer programming," *Mathematical Programming*, vol. 25, pp. 46–63, 1983.

[123] Van Roy, T. J., "Cross decomposition for mixed integer programming," *Mathematical Programming*, vol. 25, pp. 46–63, 1983.

[124] Vizing, V., "On an estimate of the chromatic class of a p-graph," *Diskrete Analiz*, vol. 3, pp. 25–30, 1964.

[125] Weatherford, L. R., "A tutorial on optimization in the context of perishable-asset revenue management problems for the airline industry," in *Operations Research in the Airline Industry* (Yu, G., ed.), ch. 3, pp. 68–100, Austin, TX: Kluwer Academic Publishers, 1998.

[126] Wei, G., Yu, G., and Song, M., "Optimization model and algorithm for crew management during airline irregular operations," *Journal of Combinatorial Optimization*, vol. 1, pp. 305–321, 1997.

[127] Weide, O., Ryan, D., and Ehrgott, M., "An iterative approach to robust and integrated aircraft routing and crew scheduling," *Computers and Operations Research*, vol. 37, no. 5, pp. 833–844, 2010.

[128] Wolsey, L. A., "Facets and strong valid inequalities for integer programs," *Operations Research*, vol. 24, pp. 367–372, 1976.

[129] Yan, S. and Lin, C.-G., "Airline scheduling for the temporary closure of airports," *Transportation Science*, vol. 31, no. 1, pp. 72–82, 1997.

[130] Yan, S. and Tu, Y.-P., "Multifleet routing and multistop flight scheduling for schedule perturbation," *European Journal of Operational Research*, vol. 103, no. 1, pp. 155–169, 1997.

[131] Yan, S. and Yang, D.-H., "A decision support framework for handling schedule perturbation," *Transportation Research B: Methodological*, vol. 30, no. 6, pp. 405–419, 1996.

[132] Yen, J. and Birge, J. R., "A stochastic programming approach to the airline crew scheduling problem," *Transportation Science*, vol. 40, no. 1, pp. 3–14, 2006.

[133] Zhang, Y. and Hansen, M., "Real-time intermodal substitution: Strategy for airline recovery from schedule perturbation and for mitigation of airport congestion," *Transport Resarch Record*, vol. 2052, pp. 90–99, 2008.

[134] ZHU, Y., "Evaluating airline delays: The role of airline networks, schedules, and passenger demands," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, February 2009.