Spring 3-14-2014

Runway Operations Management: Models, Enhancements, and Decomposition Techniques

Farbod Farhadi

farhadi.farbod@gmail.com

RUNWAY OPERATIONS MANAGEMENT: MODELS, ENHANCEMENTS, AND DECOMPOSITION TECHNIQUES

A Dissertation Presented

by

FARBOD FARHADI

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2014

Isenberg School of Management

RUNWAY OPERATIONS MANAGEMENT: MODELS, ENHANCEMENTS, AND DECOMPOSITION TECHNIQUES

A Dissertation Presented

by

FARBOD FARHADI

Approved as to style and content by:

_____

Ahmed Ghoniem, Chair

_____

Iqbal Agha, Member

_____

Robert A. Nakosteen, Member

_____

J. MacGregor Smith, Member

_____

George R. Milne, Director of PhD. Program
Isenberg School of Management

I dedicate this work to my father, mother, and brother. My most sincere gratitude to my loving parents, Farhad and Khadijeh, and my brother, Farzad, whose selfless love and support gives me power to live for today and hope for tomorrow.

ACKNOWLEDGMENTS

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

First and foremost, I would like to thank my advisor and mentor, Dr. Ahmed Ghoniem for his many years of thoughtful, patient guidance and support. This thesis would not have been possible without his constant trust, precision, and supervision. His guidance has made this a thoughtful and rewarding journey.

I would also like to extend my gratitude to the members of my committee, Dr. Iqbal Agha, Dr. Robert A. Nakosteen, and Dr. J. MacGregor Smith, for their helpful comments and suggestions on all stages of this project. Together, their valuable technical and practical advice to my thesis have been invaluable and will forever be appreciated.

A special thank you to all my friends whose presence, support and friendship helped me to stay focused on my goals and who have provided me with the encouragement to continue on this path.

ABSTRACT

RUNWAY OPERATIONS MANAGEMENT: MODELS, ENHANCEMENTS, AND DECOMPOSITION TECHNIQUES

MAY 2014

FARBOD FARHADI

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Ahmed Ghoniem

Air traffic loads have been on the rise over the last several decades and are expected to double, and possibly triple in some regions, over the coming decade. With the advent of larger aircraft and ever-increasing air traffic loads, aviation authorities are continually pressured to examine capacity expansions and to adopt better strategies for capacity utilization. However, this growth in air traffic volumes has not been accompanied by adequate capacity expansions in the air transport infrastructure. It is, therefore, predicted that flight delays costing multi-billion dollars will continue to negatively impact airline companies and consumers. In airport operations management, runways constitute a scarce resource and a key bottleneck that impacts system-wide capacity (Idris et al. 1999). Throughout the three essays that form this dissertation, enhanced optimization models and effective decomposition techniques are proposed for runway operations management, while taking into consideration safety and practical constraints that govern access to runways.

Essay One proposes a three-faceted approach for runway capacity management, based on the runway configuration, a chosen aircraft assignment/sequencing policy, and an aircraft separation standard as typically enforced by aviation authorities. With the objective of minimizing a fuel burn cost function, we propose optimization-based heuristics that are grounded in a classical mixed-integer programming formulation. By slightly altering the FCFS sequence, the proposed optimization-based heuristics not only preserve fairness among aircraft, but also consistently produce excellent (optimal or near optimal) solutions. Using real data and alternative runway settings, our computational study examines the transition from the (Old) Doha International Airport to the New Doha International Airport in light of our proposed optimization methodology.

Essay Two examines aircraft sequencing problems over multiple runways under mixed mode operations. To curtail the computational effort associated with classical mixed-integer formulations for aircraft sequencing problems, valid inequalities, preprocessing routines and symmetry-defeating hierarchical constraints are proposed. These enhancements yield computational savings over a base mixed-integer formulation when solved via branch-and-bound/cut techniques that are embedded in commercial optimization solvers such as CPLEX. To further enhance its computational tractability, the problem is alternatively reformulated as a set partitioning model (with a convexity constraint) that prompts the development of a specialized column generation approach. The latter is accelerated by incorporating several algorithmic features, including an interior point dual stabilization scheme (Rousseau et al. 2007), a complementary column generation routine (Ghoniem and Sherali, 2009), and a dynamic lower bounding feature. Empirical results using a set of computationally challenging simulated instances demonstrate the effectiveness and the relative merits of the strengthened mixed-integer formulation and the accelerated column generation approach.

Essay Three presents an effective dynamic programming algorithm for solving Elementary Shortest Path Problems with Resource Constraints (ESPPRC). This is particularly beneficial, because the ESPPRC structure arises in the column generation pricing subproblem which, in turn, causes computational challenges as noted in Essay Two. Extending the work by Feillet et al. (2004), the proposed algorithm dynamically constructs optimal aircraft schedules based on the shortest path between operations while enforcing time-window restrictions and consecutive as well as non-consecutive minimum separation times between aircraft. Using the aircraft separation standard by the Federal Aviation Administration (FAA), our computational study reports very promising results, whereby the proposed dynamic programming approach greatly outperforms the solution of the subproblem as a mixed-integer programming formulation using commercial solvers such as CPLEX and paves the way for developing effective branch-and-price algorithms for multiple-runway aircraft sequencing problems.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION AND MOTIVATION

Since the first powered airplane flight in 1903, aviation has been instrumental, at times of war or peace, to the growth of economies worldwide and to the creation of international markets supported by global supply chains. It has brought a revolution to the notion of travel, whether for business or leisure. Ever-increasing demand trends and the advent of new flight patterns due to the introduction of new and long-haul aircraft models require aviation authorities to permanently seek efficient procedures to better manage extant and newly built aviation infrastructures. In this chapter, we provide an overview of air transportation systems in general and runway operations management in particular. Section 1.1 briefly highlights key milestones in the history of modern aviation in the United States, the role of aviation in enabling economic growth and the challenges it faces with ever-increasing air traffic loads, and important components of air transport infrastructure and operations. In the latter, runways constitute a scarce resource that largely constrains airport capacity and operations. Section 1.2 provides a brief literature review on runway operations management, spanning modeling approaches, solution techniques, and runway performance metrics. Section 1.3 summarizes the organization of the proposal.

1.1. Air Transportation System

Civil aviation involves the management of the following activities and complex systems: Air transport including commercial carriage by air, non-commercial flights (e.g., private airplanes), commercial non-transport (e.g., aerial crop dusting and sur-

veying); air infrastructure (e.g., airports and air navigation facilities); and manufacturing (e.g., aircraft, engines, and avionics). Air transport is the most important constituent of civil aviation. In Subsection 1.1.1, we first briefly overview the key elements of the history of modern aviation in the United States. In Subsection 1.1.2, we discuss the role of air transport in enabling economies worldwide. In Subsection 1.1.3, we present air transport infrastructure and operations, and highlight the importance of the runway, a scarce resource that conditions the overall systemic performance and capacity of an airport.

1.1.1 Brief Historic Note

In the early years of the twentieth century, operating an airplane was considered risky, or at least adventurous, and aviation development was sparked in Europe by a race to acquire air weapons before and during World War I. It grew in the United States as well, but its use continued to be relatively limited until 1925 when the Air Mail Act allowed the Post Office to contract with private airlines to deliver mail. This act, in turn, encouraged the development of the airline industry, further accelerated by the Air Commerce Act in 1926 which authorized the Secretary of Commerce power to establish airways, certify aircraft, license pilots, and issue and enforce air traffic regulations. The first commercial airlines, including Pan American, Western Air Express and Ford Transport Service, began to operate and were joined in the following decades by many other airlines, such as United and American Airlines. In 1978, the U.S. Congress passed the Airline Deregulation Act which removed governmental control over commercial air fares, routes, and schedules, thereby enabling free market competition in commercial aviation. Air safety regulations continued, however, to be enforced by the Federal Aviation Administration. New airlines emerged into the market and many new routes connected cities directly. The number of costumers increased and fares dropped. Over the last decades, in the era of globalisation, air

traffic volumes have been steadily growing, prompted by new technologies, business opportunities, and prospects of prosperity in developing countries.

During the second half of the twentieth century, the advent of commercial aircraft and the development of local airports enabled not only industries and private businesses but also local communities to access air transportation services. At the turn of the millennium, the annual air traffic was in excess of 1.63 billion passengers who boarded scheduled flights on domestic and international routes (ICAO, Circular 292-AT/124). This was 181 times greater than the total number of passengers in 1945 (9 million passengers). In the United States alone, during peak air travel times, there are over 5,000 airplanes in the sky every hour (www.flightradar24.com). This translates to approximately 50,000 aircraft operating in our skies each day. According to Airports Council International (ACI), there were about 77 million aircraft movements worldwide in 2011. Airline scheduled services alone carried about 5,440 million passengers and moved over 90 million tons of freight and mail. Air traffic volumes are expecting to continue to rise: The total number of passengers is predicted to exceed 12 billion people in domestic and international flights and freight is estimated to reach 225 million tons by 2031 (ACI, Global Traffic Forecast, 2011).

1.1.2 Role in Global Economy and Future Challenges

Air transport can play a prominent role in the vitality of a region and its economy. It has been recognized as an important element of intermodal logistics in global supply chains and as a crucial service for individuals, corporations, and governments. Reflecting about the benefits of an air transport system, and how its unavailability can significantly inhibit a region's economic potential, Wells (1992) noted that:

> *A community's lack of an airport can be as detrimental to its development as being bypassed by the railroads a century ago, or left off the highway map 50 years ago. (Wells, 1992)*

3

Some benefits of air transport for a region include, but are not limited to, the following four aspects:

1. Air transport facilitates the integration of a region into a national, and possibly global, economy. It enables better access to fast growing markets and suppliers and facilitates interactions between industries and businesses.

2. Further, it can attract new businesses to a region, be it for tourism, manufacturing, or service activities. This, in turn, can stimulate a local economy and support its growth.

3. Air transport also enables the creation of a web of local, auxiliary business activities driven by the presence of an airport. For example, this can be beneficial for air cargo, airlines, ground transportation, hotels, or local restaurants.

4. The convenience of having an airport in relatively close proximity to a city typically has a positive impact on real state value.

The economic impacts of air transport can be divided into three categories. *Direct impacts* account for the effects on industries that directly depend on civil aviation, such as travel and tourism. *Indirect impacts* reflect the impacts on other related industries in the supply chain of civil aviation, such as aerospace manufacturers. *Induced impacts* refer to the overall benefit to an economy as the income generated by civil aviation gets re-invested in the growth and betterment of services for a local community. In 2000, the total impact of civil aviation in United States was over 900 billion dollars with an associated 11 million jobs, representing 9% of the United States gross domestic product (GDO). Of this economic impact, commercial aviation contributed 88%, whereas 12% are attributed to general aviation. In 2009, civil aviation contributed 1.3 trillion dollars to the economy of the United States which constituted 5.2% of the GDP.

In a study conducted by International Air Transport Association (IATA), the relationship between a country's level of connectivity to global air transport networks and its level of productivity and economic growth has been investigated. The study was performed across 48 countries, including both developed and developing economies, and over a ten-year period, from 1996 to 2005. One of the findings suggested that an investment of 1,805 million Canadian dollars at Vancouver airport was estimated to have led to a 5.4% increase in the overall connectivity of Canada, raising Canada's national productivity by 0.04%. It was also revealed that such improvements implied an annual increase in Canadian GDP by 348 million Canadian dollars (an annual economic rate of return of 19.3%) that can fully payback the investment within five to six years.

Air transport has grown over the last decades due to technological progress, capital intensive investments, and the increasing demand for air travel and freight services. In 2012, there were over forty-three thousand airports worldwide, with over fifteen thousand airports operating in the United States only. According to the Federal Aviation Administration (FAA), the total number of commercial airline fleet in the United States (including regional carriers) is estimated at over seven thousand aircraft at the end of 2011.

Despite all of the extant infrastructure worldwide, air transport systems are often facing congestion and some have reached their design capacity. For example, in 2011, the average delay per delayed flight is reported to be 29 minutes for arrival traffic and 28 minutes for departure traffic in European airports (EUROCONTROL, 2011). Moreover, air traffic is predicted to double, and even triple, in some areas of the world over the coming decades. In a 2009 survey conducted by the European Organization for the Safety of Air Navigation (EUROCONTROL), it has been suggested that to deal with the increasing scarcity of air transport system resources (e.g., slots, fre-

quencies), air transportation systems should adopt an integrated network congestion management approach.

Many investments and development programs are under investigation or implementation in order to improve and adapt aviation infrastructure in face of worldwide growing demands. Europe is implementing a new system for its Air Traffic Management (ATM) called Single European Sky ATM Research (SESAR). This collaborative project aims at entirely integrating the European airspace and its ATM by year 2020, and is intended to meet future airspace capacity and safety requirements. The United States has also supported a program called the Next Generation Air Transportation System (NextGen). This establishes a new National Airspace System that will be implemented throughout the United States in stages between 2012 and 2025. NextGen proposes to transform the United States ATC system from a ground-based system to a satellite-based system which would shorten air routes and reduce flight times and fuel burn, thereby reducing traffic delays and increasing airspace capacity. Such programs and future initiatives will continue to seek new strategies for capacity management and improvement in order to efficiently respond to rising air traffic volumes.

1.1.3 Infrastructure and Operations

Operations in the air transportation system can be categorized into *ground operations*, which typically take place in airports, and *airborne operations*, which occur in the airspace. Managing these operations require the collaboration of three entities, namely, airlines, airports, and air traffic controllers, depending on which phase of a flight an aircraft is. As depicted in Figure 1.1, ground operations include stationary, push back, taxiing and idling states, whereas airborne operations follow a take-off and comprise climb, cruise, descent, final approach, and landing states. Stationary and idling are two ground states where an aircraft does not move, with engines turned off or on, respectively. Push back is the phase where aircraft departs from its designated

6
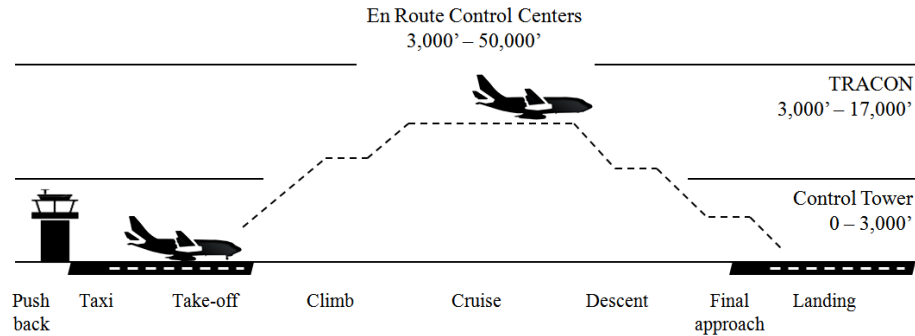
Figure 1.1: Air Traffic Controllers and Phases of a Flight

gate. Taxiing is the phase of the flight where an aircraft traverses taxiways to reach its assigned runway. In a takeoff phase, an aircraft accelerates and transitions from ground motion to flying over the runway. In the climb phase of a flight, an aircraft ascends to greater altitudes (typically 30,000 ft or 10 km), before it can travel in a safe and economic way. In the cruise phase of a flight, the aircraft travel at a nearly constant speed and under most fuel efficient settings. In the descent phase, an aircraft gradually decreases its altitude in preparation of landing, which brings an aircraft back to the ground by accessing an available runway.

Air traffic controllers are in charge of managing aircraft operations, be it for commercial or private flights. Air Traffic Control (ATC) is a system that monitors and coordinates aircraft's air and ground operations in order to direct aircraft departures and landings. It also ensures that air traffic flows smoothly with minimal delays. ATC is enforced by several ground-based controllers who direct aircraft on the ground and through controlled airspace. The primary purpose of ATC systems worldwide is to prevent collisions, organize and expedite the flow of traffic, and provide information and other support for pilots. In some countries, ATC also plays a security or defensive role, or is operated by the military.

ATC in the United States is run by the FAA. As depicted in Figure 1.2, the United States airspace is divided into 23 main zones (centers), each of which is further divided
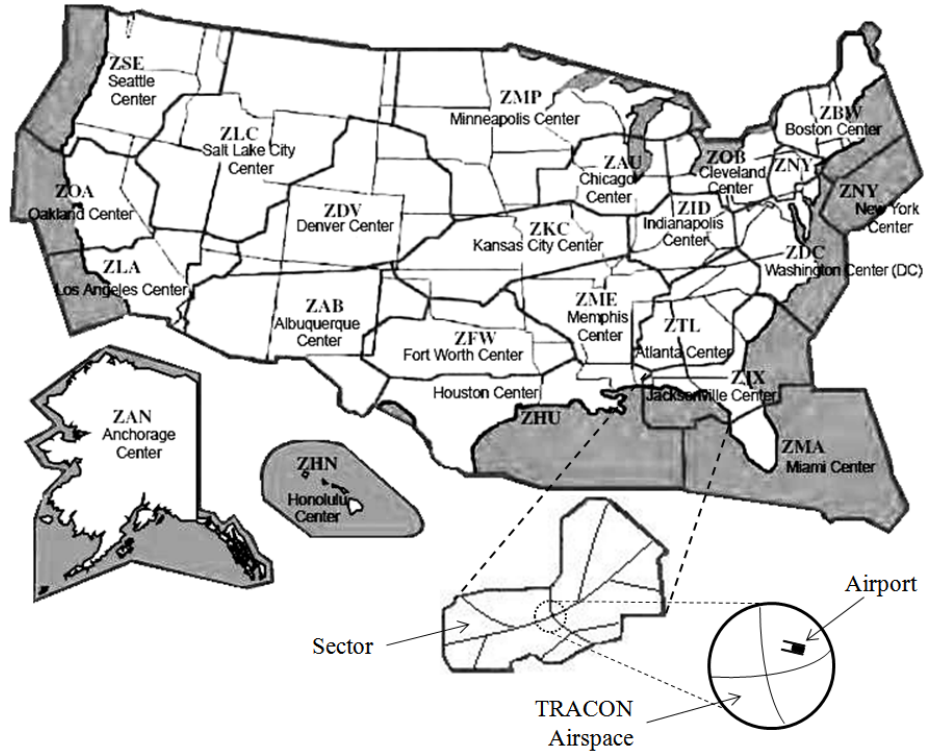
Figure 1.2: United States Air Traffic Control Zones

into sectors. Within each zone, certain areas of the airspace, of about 50 miles in diameter, are monitored by TRACON (Terminal Radar Approach CONtrol). Within each TRACON, there could be several airports, each controlling its airspace within a 5-mile radius. The TRACON controllers direct aircraft that are transitioning from their en-route phase to the approach phase into a destination airport located within the TRACON's airspace. An Air traffic control tower (ATCT) is located at every airport that handles all takeoff, landing, and ground traffic. The transition between ATCT, TRACON, and En-route Control Centers is also depicted in Figure 1.1.

To ensure air traffic safety and prevent collision, ATC enforces separation rules, which requires every aircraft to maintain a minimum volume of empty space around it at all times. At an airport, this translates as a minimum safety separation time that must separate two consecutive runway operations, which will depend on the operation type (departure/landing) and the respective sizes of the leading and fol-

lowing aircraft. Many aircraft also have collision avoidance systems, which provide additional safety support. In many countries, ATC provides services to all private, military, and commercial aircraft. The instructions provided by ATC to the pilots depend on the type of flight and the position of the aircraft in the airspace. Beyond instructions, it is incumbent upon the pilot in command to ultimately ensure safe operations of the aircraft and to react to emergencies or perceived dangers.

1.2. Literature on Runway Operations Management

There exist several research streams in the rich body of literature on air transportation, including: (i) ground operations problems related to gate assignment, aircraft sequencing over runways, or maintenance scheduling; and (ii) air-side operations problems pertaining to flight routing and scheduling, airspace planning, and final descent strategies. Prominent amongst ground operations, runway scheduling problems are particularly important, as runways constitute a key bottleneck that conditions downstream airport-wide operations (Idris et al., 1998). This dissertation focuses on investigating efficient strategies for improved runway capacity utilization. As a preamble, this section reviews the relevant literature on runway operations management. Subsection 1.2.1 reviews classical, static or dynamic, modeling approaches for runway operations. Subsection 1.2.2 summarizes exact and heuristic solution techniques that are commonly employed to address runway scheduling problems. In Subsection 1.2.3, we briefly discuss performance metrics pertinent to runway operations. At the beginning of each subsequent chapter, a relevant, more detailed review of the literature is presented.

1.2.1 Modeling Approaches

Several studies consider a static environment where for a given set of aircraft, with known information on each aircraft (e.g. operation type, its target time for runway

access, etc.), the decision-maker seeks to sequence aircraft in a fashion that yields a best schedule with respect to a chosen objective function. Luenberger (1998) offered a modeling approach for the static aircraft sequencing problem based on the classic traveling salesman problem (TSP). In this analogy, an aircraft plays the role of a city in the TSP problem and separation times between runway operations serve as inter-city distances. Beasley et al. (2000) introduced a mixed-integer program (MIP) with disjunctive constraints to model aircraft landings over a single or multiple runways. A decade later, and although the benchmark instances provided in Beasley et al. (2000) do not pose any computational difficulty to recent versions of commercial solvers and modern computers, the MIP in Beasley et al. (2000) is a classical benchmark model for aircraft sequencing problems. Brentnall (2006) proposed a machine scheduling model with sequence dependent setup times where each job corresponds to a landing aircraft and each machine with a limited capacity represents a runway. Bianco et al. (2006) also introduced a static model for scheduling landings and takeoffs in the terminal area.

Runway operations can be modeled under a dynamic environment such as a queuing system or a rolling horizon framework. In this setting, the set of aircraft is dynamically updated over time with new operations and more-up-to-date information on individual aircraft. Pujet et al. (1999) proposed a dynamic queuing model for aircraft departure problems, whereas Idris (2001) developed an analytical queuing framework for departure process dynamics. Hu and Chen (2005) also considered a dynamic aircraft landing problem and used an approach based on receding horizon control. Bauerle et al. (2007) modeled the landing problem as a special queuing system where costumers and service times represent incoming aircraft and aircraft separation times, respectively.

1.2.2 Solution Techniques

Bennell et al. (2011) provide an excellent survey of solution techniques for runway scheduling problems, spanning a broad spectrum of exact and heuristic methods. Exact solution techniques largely employ Branch-and-Bound (B&B) algorithms or dynamic programming (DP) approaches, whereas heuristic techniques comprise constructive type heuristics (e.g., tour construction and improvement schemes) and metaheuristics (e.g., genetic algorithms, simulated annealing, tabu search, etc.).

Exact solution approaches most employ a B&B algorithm and solve moderately-sized problem instances. Brinton (1992) proposed one of the early works that use B&B algorithms for aircraft arrival scheduling, followed by Abela et al. (1993) who devised a specialized B&B algorithm. Beasley et al. (2000) also relied on B&B algorithms to address a proposed MIP model. Wen et al. (2005) considered the MIP model by Beasley et al. (2000) for which they proposed a column generation approach. A wide range of other studies investigate the usefulness of exact algorithms for runway management problems. However, heuristic methods have received a great deal of attention due to the computational difficulty exact solution methods experience for large problem instances and, sometimes, even for certain moderately-sized instances.

Dear and Sherif (1991) present an enumerative heuristic for the static and dynamic case of aircraft landing problem. Anagnostakis and Clarke (2002) proposed a two-stage heuristic algorithm for solving a runway operation planning problem. Pinol and Beasley (2006) devised two population-based heuristics, namely, a scatter search and a bionomic algorithm, which are tested on publicly available benchmark instances involving up to five hundred aircraft and five runways. Atkin et al. (2004) also proposed a metaheuristic that addresses aircraft departure sequencing with practical physical constraints based on London Heathrow Airport. Many papers employed a specialized genetic algorithm, including Stevens (1995), Ciesielski and Scerri (1997), and Capri

and Ignaccolo (2004). Bencheikh et al. (2009) and Randall (2002) presented an ant colony optimization methodology for aircraft landing problem.

1.2.3 Performance Metrics

Many different parties are involved in an air transportation system, be it airlines, airports, governments, air traffic controllers, or aviation regulators. As noted in Bennell et al. (2011), depending on the specific interest of the decision-maker, alternative objectives may be considered. Therefore, the problem of managing runway operations may require the consideration of multiple, possibly conflicting, objectives which can help reveal attractive trade-offs for the decision-maker. Some of the key considerations related to runway management include the following aspects:

(a) At an Airport level:

- Workload of ground staff

- Aircraft maintenance schedule

- Gate utilization

(b) From an airline's viewpoint:

- Operating costs (mainly fuel burn and crew cost)

- Total passenger delays

- Tail assignment (assigning aircraft to flights)

- Flight routing schedules

(c) From an air traffic controller's viewpoint:

- Safety of operations

- Runway throughput

- Airspace capacity

- Fairness among operations

- Managing taxi routes

- Arrival/departure delay

(d) From a regulatory and governmental viewpoint:

- Safety of residential areas in the region

- Environmental effects (noise and air pollution).

A variety of objectives have been considered in the literature. Psaraftis (1978) examined runway throughput maximization; an objective function that continues to receive attention in the literature, as in the study by Anagnostakis and Clarke (2002). This objective may be detrimental to certain other considerations such as fairness amongst aircraft. Brentnall (2006) and Beasley et al. (2000) adopted a more airline-centric objective that minimizes a total weighted aircraft earliness and tardiness based on estimated aircraft target times. Likewise, Pinol and Beasley (2006) examined two earliness/tardiness-related objectives: (i) A nonlinear objective that maximizes the difference between the squared earliness and the squared tardiness of aircraft, thereby encouraging early landings; and (ii) Another (linearizable) objective that minimizes the total weighted earliness and tardiness, which penalizes any positive or negative deviation from aircraft target landing times. Another nonlinear objective is utilized by Atkin et al. (2007) for minimizing deviations from target departure times. Abela et al. (1993) employed an objective function that minimizes the cost associated with an aircraft speeding up or holding. In more recent studies more elaborate objectives have been utilized. Fuel costs are included in a study by Lee and Balakrishnan (2008) and are compared against aircraft delay and runway throughput considerations. Sölveling et al. (2011) proposed a multi-faceted objective that involves cost of fuel, passenger

and crew, emissions and noise. Boysen and Fliedner (2011) examined aircraft landing scheduling problems with a focus on balancing the workload of ground staff at an airport. In this proposal, multiple objectives are considered based on the problem statement and scope that is introduced in each of the subsequent chapters.

1.3. Organization of Dissertation

This dissertation is organized as follows. Chapter 2 presents an empirical study on runway capacity management strategies. Departing from a large body of literature that focuses – at an operational level – on building runway schedules, this chapter proposes a three-faceted approach to analyze runway capacity management. In particular, consideration is given to the physical configuration of runways, runway scheduling strategies (i.e. runway assignment and aircraft sequencing schemes), and runway safety regulations. The relative merits and the impacts of alternative settings on aircraft fuel burn cost and average delay are examined using real data acquired from Doha International Airport in Qatar.

Chapter 3 examines aircraft sequencing problems over multiple runways using optimization models that are enhanced via valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints. To further enhance the computational tractability of this class of problems, this work proposes a set partitioning model reformulation of the problem that prompts the development of a column generation algorithm. This decomposition technique is further accelerated by incorporating an interior point dual stabilization scheme, a complementary column generation routine, and a dynamic lower bounding feature.

Chapter 4 presents an effective dynamic programming algorithm for solving Elementary Shortest Path Problems with Resource Constraints (ESPPRC), a structure that arises in the column generation pricing subproblem discussed in Chapter 3. Extending the work by Feillet et al. (2004), the proposed algorithm dynamically

constructs optimal aircraft schedules based on the shortest path between operations while enforcing time-window restrictions and consecutive as well as non-consecutive minimum separation times between aircraft. Using the aircraft separation standard by the Federal Aviation Administration (FAA), our computational study reports very promising results, whereby the proposed dynamic programming approach greatly out-performs the solution of the subproblem as a mixed-integer programming formulation. This paves the way for developing effective branch-and-price algorithms for this class of problems.

Chapter 5 concludes the dissertation by summarizing our findings and directions for future research.

CHAPTER 2

RUNWAY CAPACITY MANAGEMENT – AN EMPIRICAL STUDY WITH APPLICATION TO DOHA INTERNATIONAL AIRPORT

This chapter examines a three-faceted approach for runway capacity management, based on the runway configuration, a chosen sequencing policy, and an aircraft separation standard. In this context, we propose optimization-based heuristics that yield optimal or near-optimal schedules and assess their benefits under alternative runway settings. This integrated approach is applied, in collaboration with *Qatar Civil Aviation Authority*, to investigating the transition from the (Old) Doha International Airport to the New Doha International Airport. Our computational study of alternative runway settings uses optimization methodology along with tailored preprocessing routines.

2.1. Introduction

In recent years, new flight patterns – facilitated by the advent of larger aircraft – and ever-increasing air traffic loads have required airlines and airports to seek new frontiers in operations efficiency. In 2012, Airports Council International (ACI) reported over 6 billion passengers in domestic and international flights worldwide. By 2025, it is anticipated that this figure will increase by at least 50%, with over 9 billion passengers in global air traffic. The growing air traffic trends necessitate the construction of new airports, major capacity expansions at busy airports, a commensurate adjustment of aviation infrastructure, and the identification of operational

Year	Passenger	% Increase	Cargo (Kg)	% Increase	Aircraft movement	% Increase
2007	9,459,812	-	247,163,753	-	65,373	-
2008	12,272,505	29.7	409,462,811	65.7	90,713	38.8
2009	13,113,224	6.9	522,920,986	27.7	101,941	12.4
2010	15,724,027	19.9	699,941,401	33.9	118,751	16.5
2011	18,108,521	15.2	795,558,797	13.7	136,768	15.2
2012	21,163,382	16.9	826,669,094	3.9	155,671	13.8

Table 2.1: Air Traffic Volumes at Doha International Airport (www.dohaairport.com)

policies and managerial directives that best avail of existing capacity. In particular, airports are faced with persistent challenges related to runway scheduling, a key bottlneck in the air transport system.

The Middle East is serving as a hub for global trade and transport and has witnessed rapid air traffic growth over the last years. According to the *International Civil Aviation Organization* (ICAO), international air traffic amounts to nearly 60% of the total passenger traffic, 10% of which occurs in the Middle East. In this context, the United Arab Emirates and Qatar are making large investments in aviation infrastructure and host two major airlines, *Emirates* and *Qatar Airways*. In 2012, Doha International Airport (DOH) ranked 25th in international passenger traffic and experienced the second largest growth of 19%, after Istanbul with 25%, over the previous year. Table 2.1 further summarizes the 2007-2012 traffic at DOH, reflecting sustained growth rates in passenger, cargo, and aircraft movements over the last few years.

DOH currently operates with a single runway (see Figure 2.1a), one of the longest at civil airports with a length of 4,570 meters. It employs an FCFS policy for aircraft sequencing and the ICAO aircraft safety separation standard. The main terminal at DOH has been expanded several times over the last years in order to accommodate sharply increasing air traffic loads (see Table 2.1). In 2008, the airport witnessed a 38.8% growth in aircraft movement and ranked amongst the 100 busiest airports worldwide. Further, DOH was the world's 27th busiest airport by cargo traffic in

(a) (Old) Doha International Airport (b) New Doha International Airport
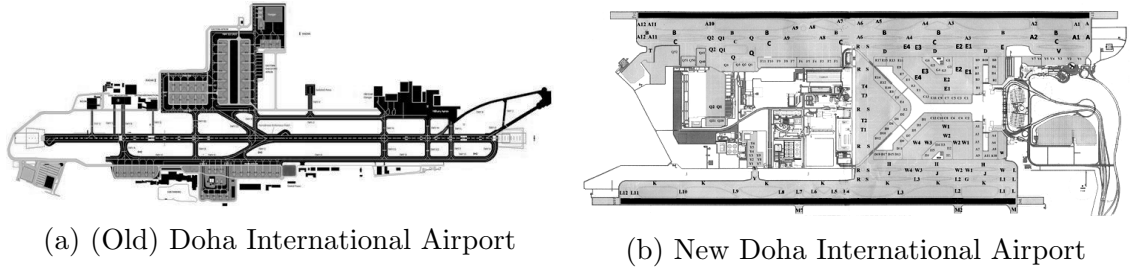
Figure 2.1: Layout of the "Old" and the New Doha International Airports

2010, with over 15 million passengers. To manage the growing air traffic and to better prepare the country for hosting the Qatar 2022 FIFA World Cup and the Qatar 2030 Strategic Vision, the *New Doha International Airport*, to be officially called *Hamad International Airport* (HIA), was constructed as a distinct, new facility with two parallel independent runways. It is expected to replace the single-runway DOH in a near future. The first phase of HIA is planned for inauguration with one runway offering a capacity of 29 million passengers. It is designed to ultimately operate with two parallel independent runways, as depicted in Figure 2.1b, and a capacity of up to 50 million passengers, two million tons of cargo, and 320,000 aircraft landings/take-offs per year upon its completion in 2015.

This study is motivated by our collaboration with the *Qatar Civil Aviation Authority* and the transition from the single-runway DOH to the HIA with two parallel runways in 2013. In particular, our work is predicated on the notion that runway capacity should be analyzed in light of three primary factors: (i) The runway physical configuration and operating mode (segregated vs. mixed); (ii) The adopted aircraft scheduling policy which spans heuristics, metaheuristics, and optimization approaches; and (iii) The specific standard adopted for aircraft separation standards. Two of the main commonly used standards that we examine in this chapter are stipulated by the *Federal Aviation Administration* (FAA) and the *International Civil Aviation Organization* (ICAO).

The remainder of the chapter is organized as follows. Section 2.2 positions the present work in the context of the extensive literature on aircraft sequencing problems. Section 2.3 presents an optimization model for runway scheduling, which is enhanced via preprocessing routines. We also propose heuristics that are grounded in the optimization model and the FCFS sequencing policy. In Section 2.4, we discuss data related to runway operations at in Doha and present our computational results for alternative runway settings using the proposed solution methodology and heuristic approaches. Section 2.5 concludes the chapter with a summary of our findings and directions for future research.

2.2. Literature Review

At an operational level, runway scheduling problems seek to determine effective aircraft schedules over one or multiple runways using pertinent cost objectives or performance criteria. There exists a large body of literature on aircraft sequencing approaches that is grounded in seminal works on machine scheduling. Bennell et al. (2011) offer an excellent survey of runway scheduling problems, covering modeling approaches, solution techniques, and performance criteria. Popular solution techniques for runway scheduling problems include dynamic programming, branch-and-bound/cut algorithms, and a broad spectrum of constructive/greedy heuristics and metaheuristics. Most studies tend to focus on either departure or arrival aircraft sequencing, in isolation, with a few exceptions that consider mixed-mode operations.

Noting the similarity between aircraft sequencing problems and machine scheduling problems with sequence-dependent set up times and time-windows for the completion of jobs, Ernst et al. (1999) proposed an optimization model that is tackled using a heuristic based on branch-and-bound algorithms. In a similar spirit, Beasley et al. (2000) proposed a disjunctive mixed-integer program (MIP) for single and multiple-runway aircraft sequencing problems which is widely used in the literature. Further,

Ghoniem et al. (2013) presented an asymmetric traveling salesman problem-based (ATSP) model for combined arrival-departure aircraft sequencing problems over a single runway. The computational tractability of this formulation was significantly enhanced using valid inequalities and preprocessing routines.

A few studies in the literature address runway operations management with application to specific airports. Using landing time intervals at Logan Airport, Venkatakrishnan et al. (1993) demonstrated that aircraft sequences that outperform those identified by controllers could be constructed, thereby reducing flight delays by up to 30%. Idris et al. (1999) examined the interaction between key elements of an airport system, including runways, taxiways, ramps, and gates. Focusing on aircraft departures at Logan Airport, the authors concluded that runways constitute the principal bottleneck in the flow of airport operations and their management significantly impacts system-wide efficiency. Also, motivated by an application to London Heathrow Airport, Beasley et al. (2001) proposed a metaheuristic to improve the sequencing of landing aircraft. Atkin et al. (2008) developed a metaheuristic approach for the sequencing of departing aircraft as a decision support tool for runway controllers at Heathrow airport.

Key stakeholders in aircraft operations management include the airport, airlines, and governmental authorities (Bennell et al., 2011). Depending on the planner's interest, different performance criteria and objective functions can be considered for runway scheduling. For instance, minimizing the makespan, or equivalently maximizing the runway throughput, optimizes the start-time of the last aircraft to access the runway and is viewed as an airport-driven target. This performance criterion can, however, be detrimental to the mean aircraft delay (Lee and Balakrishnan, 2008), an objective that is more important to airlines and passengers. Beasley et al. (2000) and Ernst et al. (1999) employ an objective function that minimizes the total aircraft earliness and tardiness, measured as the weighted deviation from target landing/departure
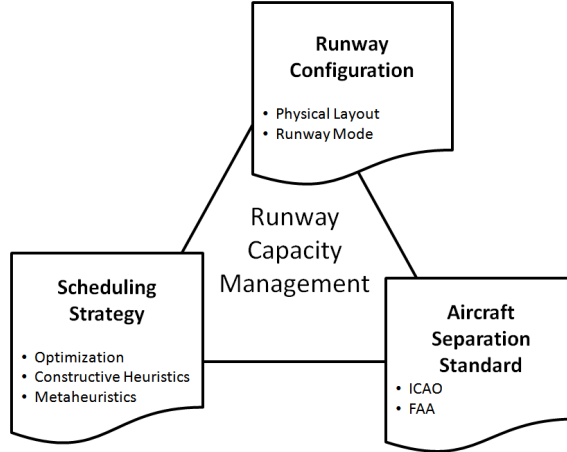
Figure 2.2: Key Factors Related to Runway Capacity

times. Such objective functions are advantageous to airlines and passengers, but also contribute to smoothing airport-wide operations. Recent studies increasingly use direct monetary costs related to fuel burn (Lee and Balakrishnan, 2008; Sölveling et al., 2011), passenger delays, or crew costs (Sölveling et al., 2011).

Whereas most studies in the literature focus on a specific exact or heuristic solution approach to the aircraft sequencing problem, the present chapter adopts a more integrated approach. As depicted in Figure 2.2, we examine the combined effect of the following three factors on runway capacity utilization and operations management: (i) a specific runway configuration, including the physical layout of runways and their operation mode (mixed or segregated); (ii) a heuristic or optimized aircraft scheduling policy; and (iii) an aircraft separation standard. The runway performance under alternative settings is assessed using optimization methodology, MIP-based heuristics, and data from DOH.

2.3. Optimization Model and Heuristic Approaches

Central to our evaluation of alternative runway settings is the use of an MIP model that is introduced in Subsection 2.3.1. In Subsection 2.3.2, we discuss tailored

21

valid inequalities and preprocessing routines for the MIP model which are developed to enhance the computational tractability of the model. In Subsection 2.3.3, optimization-based heuristics, grounded in the MIP model, are proposed with the objective of producing high quality schedules that are close in structure to the FCFS sequence that is widely used at airports.

2.3.1 Mixed-Integer Program

We consider a set of J aircraft arrivals and departures to be scheduled over a set of N parallel independent runways during a particular planning horizon. Each aircraft $j \in J$ is characterized by the following attributes: (i) its operation type $\mathcal{O}_j$ (Departure/Arrival); (ii) its weight class (Heavy, Large, or Small); (iii) a ready-time r_j and a due date d_j which enforce a time-window over which aircraft j should access a runway and start its operation; and (iv) a fuel burn cost, w_j, which depends on its operation type and weight class. We denote by $p_{j_1 j_2}$ the minimum separation time between a leading aircraft j_1 and a following aircraft j_2, which depends on their operation types and weight classes and is numerically specified by a chosen standard (ICAO or FAA) as discussed in Section 2.4.1.

An assignment binary variable z_{ij} is introduced; it equals 1 if and only if aircraft $j \in J$ is assigned to runway $i \in N$. We also introduce a sequencing binary variable $y_{j_1 j_2}$ to determine the relative order of a pair of aircraft j_1 and j_2 if they are assigned to the same runway. The continuous decision variable, t_j, establishes the time at which aircraft j accesses its assigned runway.

Given specific input parameters as described above, the Runway Capacity Management problem is formulated as the following 0-1 MIP, which we refer to as **RCM**:

RCM: Minimize $\displaystyle\sum_{j \in J} w_j(t_j - r_j)$ $\hfill$ (2.1a)

$$\sum_{i \in N} z_{ij} = 1, \quad \forall j \in J \hfill (2.1b)$$

$$r_j \leq t_j \leq d_j, \quad \forall j \in J \hfill (2.1c)$$

$$t_{j_2} \geq t_{j_1} + p_{j_1 j_2} - M(1 - y_{j_1 j_2}), \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2 \hfill (2.1d)$$

$$y_{j_1 j_2} + y_{j_2 j_1} \geq z_{i j_1} + z_{i j_2} - 1, \quad \forall i \in N, j_1 \in J, j_2 \in J, j_1 < j_2 \hfill (2.1e)$$

$$y, z \text{ binary.} \hfill (2.1f)$$

The objective function (2.1a), where the term $\sum_{j \in J} w_j r_j$ is a constant, minimizes the total fuel cost resulting from the deviation of aircraft start-times from their respective ready-times. We refer to this metric in the objective function as the total *excess fuel cost*; if all start-times equal their associated ready-times in a given schedule, then no excess fuel cost is incurred. Constraint (2.1b) assigns every aircraft to exactly one runway. Ready-time and due date restrictions are enforced in Constraint (2.1c). The disjunctive constraint (2.1d) introduces a minimum separation time between any pair of aircraft, whether consecutive or not, that are assigned to the same runway. It involves a sufficiently large scalar M, which we validly set to $M \equiv d_{j_1} - r_{j_2} + p_{j_1 j_2}$. Constraint (2.1e) guarantees that precedence between any pair of aircraft must be established if they are assigned to the same runway. Constraint (2.1f) specifies binary restrictions on decision variables.

Model RCM, in the spirit of the model in Beasley et al. (2000), can be used to evaluate the potential of alternative runway configurations, sequencing policies, and aircraft separation standards. The key modeling distinction is that we eliminate an auxiliary variable that is explicitly introduced in Beasley et al. (2000) to establish whether or not a pair of aircraft is assigned to the same runway, which results in a model size reduction in RCM. Although Model RCM is stated for a multiple-runway

configuration, it can be adjusted for a single-runway configuration by relaxing Constraints (2.1b) and (2.1e) and eliminating the z-variables.

2.3.2 Preprocessing Routines

We develop preprocessing routines with the objective of fixing the relative order of certain aircraft, without loss of optimality, and, therefore, enhancing the computational tractability of Model RCM. Such preprocessing routines can be identified by analyzing input parameters related to aircraft and separation times. For example, Constraint (2.2) states that if the preceding of aircraft j_2 to aircraft j_1 would cause the latter to violate its due date, then this relative order should be precluded to ensure feasibility:

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2, r_{j_2} + p_{j_2 j_1} > d_{j_1}. \tag{2.2}$$

Constraint (2.3) considers two equivalent aircraft that have the same fuel cost $w_{j_1} = w_{j_2}$, which implies that they have the same operation type (both are arrivals or departures) and weight class (both are Heavy, Large or Small), and where one of the aircraft has an earlier time-window. From an aircraft separation point of view, both aircraft in Constraint (2.3) are equivalent and, therefore, the earlier aircraft can be required not to follow the later one, without loss of optimality.

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2, r_{j_1} < r_{j_2}, d_{j_1} \leq d_{j_2}, w_{j_1} = w_{j_2}. \tag{2.3}$$

Constraint (2.4) considers a similar situation, but caters for the special case where the two aircraft have identical time windows. It is conceivable to require the lower-indexed aircraft not to follow the higher-indexed one, without loss of optimality:

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 < j_2, r_{j_1} = r_{j_2}, d_{j_1} = d_{j_2}, w_{j_1} = w_{j_2}. \tag{2.4}$$

There could be additional cases where a pair of aircraft j_1 and j_2 are not equivalent, but they introduce the same separation times (i.e. $p_{j_1 k} = p_{j_2 k}$ and $p_{k j_1} = p_{k j_2}$, for any aircraft k, which we simply represent as $p_{j_1,*} = p_{j_2,*}$ and $p_{*,j_1} = p_{*,j_2}$). Constraint (2.5) identifies aircraft with such symmetric separation times and requires, without loss of optimality, the aircraft with an earlier time-window and heavier fuel cost not to follow the other aircraft:

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2, r_{j_1} \leq r_{j_2}, d_{j_1} \leq d_{j_2}, p_{j_1,*} = p_{j_2,*}, p_{*,j_1} = p_{*,j_2}, w_{j_1} > w_{j_2}.$$

(2.5)

2.3.3 Optimization-based Heuristics

We propose in this section two optimization-based heuristics that are grounded in the use of the MIP model RCM and the FCFS sequencing policy. The overarching objective here is to develop heuristics that yield optimal or near-optimal solutions, while largely preserving the structure of the FCFS sequence (for practical reasons and in order to maintain fairness among aircraft). We shall, therefore, use the global optimal schedule produced by Model RCM and the FCFS schedule as two benchmarks for comparison with the proposed MIP-based heuristics, as delineated next.

a) FCFS sequencing policy with segregated-mode runways (**FCFS-SEG**). This base policy reflects the sequencing strategy that the New Doha International Airport *a priori* plans to use. Considering segregated runways, either dedicated to departures or arrivals, aircraft on a given runway are sequenced in the non-decreasing order of their ready-times. If the problem involves two runways, one is dedicated to the arrivals and the other to departures. If multiple runways are devoted to the same operation type, e.g. departures, there is a need to both assign aircraft to suitable runways and to sequence them using FCFS over the same runway. FCFS-SEG can be implemented using Model RCM. To this end,

we consider N_d and N_a, the subsets of runways dedicated exclusively for departures and arrivals, respectively, along with J_d and J_a, the subset of aircraft departures and arrivals, respectively. We then enforce the following restrictions in Model RCM:

$$z_{ij} = 0, \quad \forall i \in N_d, j \in J_a \tag{2.6}$$

$$z_{ij} = 0, \quad \forall i \in N_a, j \in J_d \tag{2.7}$$

$$t_{j_1} \leq t_{j_2}, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2 | r_{j_1} < r_{j_2} \text{ and } \mathcal{O}_{j_1} = \mathcal{O}_{j_2}. \tag{2.8}$$

b) Heuristic 1 – FCFS sequencing policy with mixed-mode runways (**FCFS-MIX**). In contrast with FCFS-SEG, runways operate in a mixed mode, allowing arrivals and departures to share runways. This proposed heuristic ranks aircraft based on their ready-times and iteratively assigns aircraft to the first available runway. Under this strategy, all aircraft assignments follow the FCFS order and no aircraft is allowed to overtake an earlier aircraft in the sequence. FCFS-MIX can be implemented by appending the following restrictions to Model RCM:

$$t_{j_1} \leq t_{j_2}, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2 | r_{j_1} < r_{j_2}. \tag{2.9}$$

c) Heuristic 2 – FCFS sequencing policy with optimized assignment (**FCFS-OPT**). Under this proposed heuristic, the assignment of aircraft to runways is optimized with the restriction that no aircraft can overtake another aircraft in its queue (i.e. aircraft of the same operation type, whether arrival or departure). However, an aircraft is allowed to overtake other aircraft of the opposite operation type if deemed pertinent from a cost reduction point of view. For example, an arriving aircraft can overtake a departure aircraft with an earlier time-window. Consequently, the FCFS order applies only within each queue of
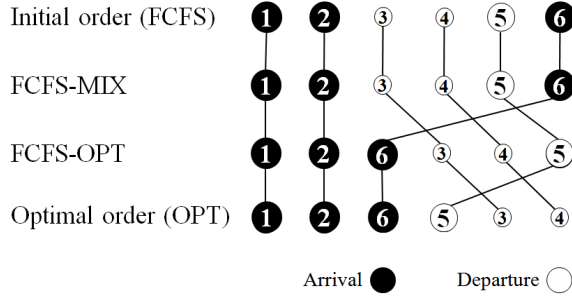
Figure 2.3: Example on scheduling settings

arrival and departure aircraft, but not across the two queues. The heuristic en-
ables an optimized interweaving of both queues and can be implemented using
Model RCM by enforcing the following constraint:

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2 | r_{j_1} < r_{j_2} \text{ and } \mathcal{O}_{j_1} = \mathcal{O}_{j_2}. \tag{2.10}$$

d) Optimal Schedule (**OPT**): We also consider the setting where the assignment
and sequencing of aircraft are optimized, independently from any FCFS con-
siderations, using Model RCM. Although this setting does not make provision
for fairness amongst aircraft, we use it as a benchmark for the best possible
performance under a given runway/data input setting.

Figure 2.3 provides an illustrative example with six aircraft sequenced over a single
runway and highlights aircraft position shifts from one sequencing policy to another.
The FCFS policy provides a base sequence for a combination of arrivals and departures
and different aircraft weight classes. In this single-runway example, the FCFS-MIX
heuristic does not alter the FCFS sequence. Under the FCFS-OPT heuristic, the
FCFS order is preserved within the arrival and departure queues but not across the
two queues. For example, arriving aircraft 6 is moved to the third position, overtaking
departing aircraft 3, 4, and 5, with the implication that this decision produces a
better solution. Under the optimal schedule, the FCFS order may be violated within

27

and across arrival and departure queues. For example, departing aircraft 5 precedes departing aircraft 3 and 4, although the latter have earlier ready-times.

2.4. Computational Study and Key Findings

Our study is anchored in the analysis of data on Doha International Airport that we obtained from *Qatar Civil Aviation Authority* in 2011. We analyzed aircraft movement patterns using SAS 9.3 and implemented all heuristic and optimization approaches using AMPL/CPLEX 12.4 on a desktop with Windows 7 professional 64-bit operating system, an Intel Core i7-2600 CPU with 3.40 GHz, and 12 GB RAM. Subsection 2.4.1 discusses the data and the aircraft separation standards we considered in the study. Subsection 2.4.2 reports our results and key findings with regard to the proposed runway capacity management model and heuristic methods.

2.4.1 Data Analysis

Our study is grounded in air traffic and aircraft movement projections in anticipation of increasing loads that HIA would have to handle. There are about 685 operations per day in typical data instances which are examined using our proposed solution approaches. This corresponding to nearly 80% of the HIA expected nominal capacity after the completion of its final construction phase, i.e. about 857 operations/day or 320,000 operations/year. The alternative runway settings (runway configuration, scheduling policy, and separation times) are encapsulated in Model RCM with the objective of minimizing the total excess fuel cost. Aircraft fuel consumption (see Appendix A) is adapted from fuel burn data in Cook et al. (2004). For each aircraft model, we employ an average fuel burn (gal/min) associated with its ground or final approach operations. We used jet fuel costs based on recent IATA data on fuel prices (3.132 USD/gal in the Middle East and Africa on March 1, 2013). In
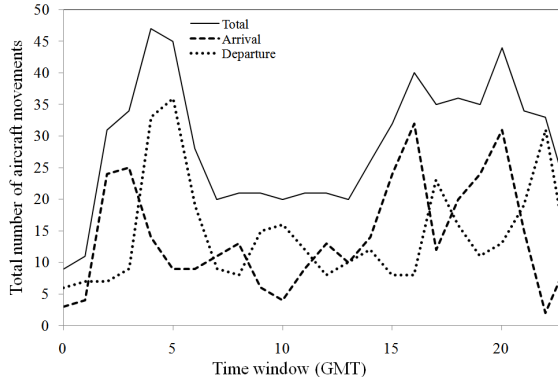
Figure 2.4: Projected Aircraft Movements in Doha

our post-solution analysis, we also record the total delay incurred under each runway setting.

Figure 2.4 depicts typical aircraft movements in our Doha-based dataset, where aircraft operations are categorized into Arrivals and Departures and where the total aircraft movements are represented. In about 50% of the day, 30 operations or more take place in a time-window of one hour. The combined number of departures and landings peaks to over 45 operations, potentially causing delays and requiring careful planning. Figures 2.5a and 2.5b provide a higher level of detail by depicting the number of aircraft arrivals and departures, separately, while categorizing aircraft by their weight classes (Heavy, Large, and Small).

In our Doha dataset, aircraft are predominantly heavy and large (39% H, 55% L and 6% S). Our analysis indicates that the inter-operation time (time lapse between the occurrence of two operations on the runway) ranges from 80 seconds to 6 minutes during different hours of the day, with a two-minute inter-operation time at an average. This is indicative of non-uniform air traffic operations throughout the day at DOH, as is typical of international airports. Doha faces heavier air traffic activity during three main time-windows of the day. As far as arrivals are concerned, busier activity takes place around hours 3, 15, and 20 GMT – Doha time being GMT + 3:00. In contrast, congested hours for departures are around hours 5, 17, and 22
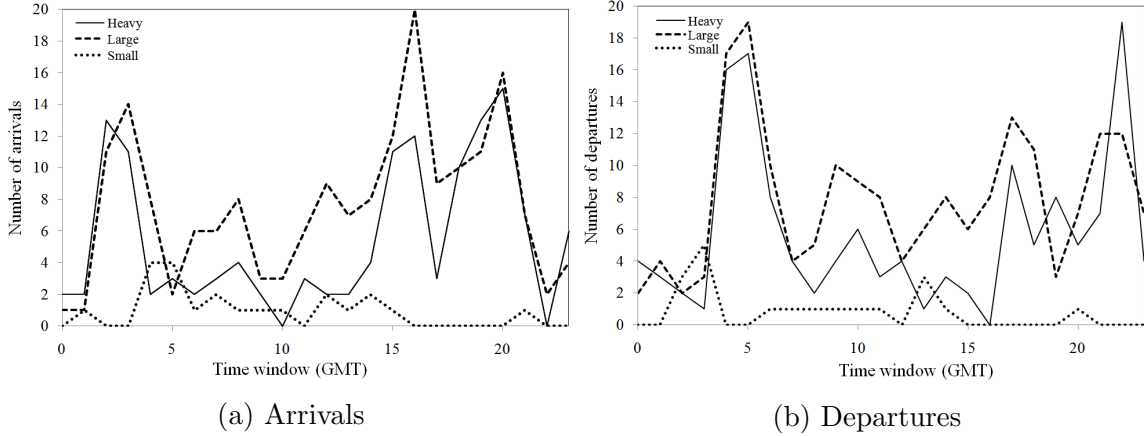
Figure 2.5: Projected Aircraft Patterns for Doha International Airport

GMT. There is approximately a three-hour difference between the busier hours for arrivals and departures that is reflective of common aircraft layovers at airports, as an arriving aircraft gets serviced and ready to depart again. Small aircraft are less present at DOH and have milder peaks of activity. It is worth noting that if aircraft operations were uniformly distributed throughout the day with a two-minute inter-operation time, then even a single runway would accommodate 720 aircraft. In practice, the capacity of the runways at Doha does not seem to be reached most of the day. However, certain time-windows of the day are particularly congested, require careful planning, and cause excess fuel and delay costs.

Aviation authorities enforce aircraft separation times between runway operations in order to obviate the dangers of wake turbulence. The magnitude of these separation times depends on the weight class of the leading/following aircraft and their operations types (landing or departure). Such separation times are typically *asymmetric*, due to the higher vulnerability of smaller aircraft to air turbulence. There exist different safety separation time standards, each resulting in a specific runway capacity utilization and airline fuel cost. We consider two different standards in our study, namely, the ICAO standard (currently adopted at DOH) and the one enforced

Departure → Departure Case				Departure → Arrival Case			
Leading \ Following	Heavy	Large	Small	Leading \ Following	Heavy	Large	Small
Heavy	90	120	120	Heavy	60	60	60
Large	60	60	60	Large	60	60	60
Small	60	60	60	Small	60	60	60
Arrival → Departure Case				Arrival → Arrival Case			
Leading \ Following	Heavy	Large	Small	Leading \ Following	Heavy	Large	Small
Heavy	75	75	75	Heavy	96	157	196
Large	75	75	75	Large	60	69	131
Small	75	75	75	Small	60	69	82

Table 2.2: Aircraft separation times (in seconds) following the FAA standard

by the FAA at airports in the United States and contrast their effects on runway operations if employed in Doha.

The ICAO standard classifies aircraft along three main weight classes (Heavy, Medium, and Light) based on their maximum takeoff weight (MTOW). It requires a minimum separation of 2 minutes between any pair of operation for any weight class unless a *light landing* follows a *heavy or medium landing*, in which case a 3-minute minimum separation time must be enforced. Likewise, FAA categorizes aircraft into similar weight classes (Heavy, Large, and Small). However, it introduces different separations based on minimum distances (in nautical miles) in compliance with the Instrument Flight Rules (IFR) that have to be maintained between aircraft operations. These nautical distances can be converted to minimum separation times in seconds assuming nominal aircraft speeds as in De Neufville and Odoni (2003), and are summarized in Table 2.2 (Lee, 2008) for different cases of Arrival/Departure considering runway occupancy times for different aircraft weight classes.

In addition to being asymmetric, the FAA separation times do not always satisfy the triangle inequality. In certain cases, the separation of consecutive aircraft is not sufficient to properly separate certain nonconsecutive aircraft in the sequence, as illustrated in Figure 2.6. In this example, the separation times of 60 seconds and 75 seconds between the consecutive operations do not introduce the 196 seconds necessary to separate the first and third operations. The need to separate all pairs of
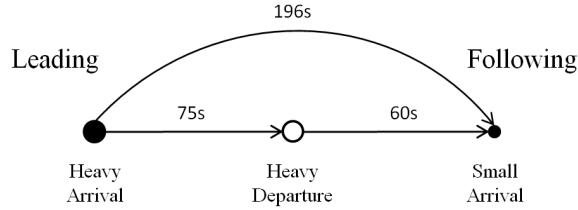
Figure 2.6: Non-triangular separation times in FAA standard

aircraft that share the same runway, whether consecutive or not, is readily enforced in Model RCM with the y-variables and the disjunctive Constraint (2.1d).

2.4.2 Results and Findings

Our empirical results are summarized in Tables 2.3 and 2.4 using alternative runway settings under ICAO and FAA separation time standards, respectively. Column 1 provides hourly time-windows of airport operations in Greenwich Mean Time (GMT). Columns 2-6 report the total excess fuel cost in US dollar (USD). An objective value of 0 reflects that all aircraft start at their ready-times and there are no deviations that result in added fuel costs. Column 2 reports fuel costs for a single runway setting (or closely parallel runways) operating under an FCFS sequencing policy, as currently implemented in DOH. Columns 3-6 report results for a setting with two parallel independent runways, as planned for HIA, under the three scheduling heuristic policies FCFS-SEG, FCFS-MIX, and FCFS-OPT and an optimal schedule (OPT).

Our proposed heuristic FCFS-OPT yields notable improvements in reducing the excess fuel cost over FCFS-SEG and FCFS-MIX, and provides near-optimal solutions that are very comparable to the optimal schedules produced by OPT. An examination of the solutions produced by FCFS-OPT and OPT approaches reveals the following: Although FCFS-OPT forces aircraft on the same runway to follow an FCFS order within the same stream of operations (departure/landing), it optimizes aircraft-runway assignments in a way that yields overall optimal or near-optimal solutions when compared to OPT results. It also optimizes the interweaving of the

32

Window	Single runway	Two runways			
(GMT)	FCFS	FCFS-SEG	FCFS-MIX	FCFS-OPT	OPT
0-1	0	0	0	0	0
1-2	206	206	69	69	69
2-3	34,355	16,816	4,513	3,849	3,849
3-4	126,516	29,878	852	754	754
4-5	145,979	13,864	2,391	1,886	1,886
5-6	235,131	33,901	2,289	2,206	2,206
6-7	205,463	15,954	2,047	1,228	1,228
7-8	142,652	938	3	3	3
8-9	129,623	1,352	219	219	219
9-10	55,666	1,234	172	172	172
10-11	26,327	317	162	162	162
11-12	16,867	1,020	206	206	206
12-13	4,020	1,336	716	716	716
13-14	4,803	1,683	523	304	304
14-15	12,321	5,632	2,908	2,183	1,963
15-16	16,920	5,724	1,606	1,547	1,547
16-17	121,249	36,369	4,196	3,319	3,319
17-18	112,657	2,028	746	646	646
18-19	187,234	6,155	2,046	1,890	1,890
19-20	295,523	21,057	4,582	3,977	3,977
20-21	448,154	21,083	3,509	3,005	3,005
21-22	344,028	15,071	4,225	3,171	3,124
22-23	230,337	11,950	921	871	871
23-24	251,911	2,570	321	321	321
Total fuel cost (USD)	3,147,942	246,139	39,221	32,707	32,440
Total delay (min)	31,327	2,516	476	511	511

Table 2.3: Fuel Costs under Alternative Runway Settings (ICAO Separation Standard)

Window	Single runway	Two runways			
(GMT)	FCFS	FCFS-SEG	FCFS-MIX	FCFS-OPT	OPT
0-1	0	0	0	0	0
1-2	172	172	34	34	34
2-3	10,152	8,512	2,727	2,671	2,671
3-4	12,853	5,105	316	306	306
4-5	6,825	2,971	862	862	862
5-6	5,121	3,169	1,141	1,141	1,141
6-7	4,838	1,772	301	301	301
7-8	531	346	0	0	0
8-9	1,017	653	124	124	124
9-10	1,090	469	77	77	77
10-11	452	199	97	97	97
11-12	1,032	586	129	129	129
12-13	1,498	885	394	394	394
13-14	1,019	673	87	87	87
14-15	4,642	3,513	1,184	972	972
15-16	5,803	4,450	508	508	508
16-17	14,345	11,646	1,745	1,745	1,745
17-18	2,877	904	241	200	200
18-19	7,003	4,577	1,050	1,050	1,050
19-20	16,402	13,018	3,138	3,138	3,138
20-21	13,237	9,898	2,241	2,241	2,241
21-22	13,702	4,757	411	411	411
22-23	3,135	2,776	584	584	584
23-24	1,753	1,077	140	140	140
Total fuel cost (USD)	129,497	82,128	17,531	17,211	17,211
Total delay (min)	1,262	696	205	209	209

Table 2.4: Fuel Costs under Alternative Runway Settings (FAA Separation Standard)

departure and arrival queues over the same runway. This highlights that aircraft-runway assignments are crucial and can yield excellent results, even when the aircraft sequence follows an FCFS policy within the same stream of operations.

In contrast, swapping aircraft positions or optimizing their sequencing within the same stream of operations does not result in notable savings. This underscores the importance of aircraft assignment decisions in reducing excess fuel costs, an aspect that is often overlooked, as more attention has been devoted to sequencing strategies. This also explains why FCFS-OPT dominates FCFS-MIX with respect to excess fuel cost, as the latter adopts myopic/greedy aircraft-runway assignments. A final note is in order to elucidate why delays under FCFS-MIX could be shorter than under FCFS-OPT. In our study, the excess fuel cost is the *unique* objective that is *optimized* in Model RCM and delays are only *recorded* in a post-solution analysis as a useful auxiliary metric (that is not optimized). As such, the greedy aircraft

Heuristic	Fuel cost		Delay		Shifted operation	Position Shifts
	Total (USD)	Saving %	Total (minutes)	Saving %		
FCFS-MIX	39,221	-	476	-	0	0
FCFS-OPT	32,707	16.61	511	-7.4	31.8%	2
OPT	32,440	17.29	511	-7.4	49.0%	2

Table 2.5: Benefits of FCFS-OPT over other Heuristics (under ICAO standard)

runway-assignment policy enforced by FCFS-MIX is dictated by the ready-times of aircraft and, therefore, can result is slightly better delays than under FCFS-OPT, while being suboptimal from an excess fuel cost viewpoint. As FCFS-OPT seeks to optimize the excess fuel cost, it performs limited position shifts that result in deviations from ready-times and occasionally slightly higher delays. To compare the trade-offs between optimizing total fuel cost and optimizing total delays we assess further computation to show the non-inferior solutions in Appendix B.

Table 2.5 summarizes results from our analysis of FCFS-MIX, FCFS-OPT, and OPT. Column 2 and 3 report fuel costs (USD) and anticipated savings for FCFS-OPT and OPT compared to FCFS-MIX. Likewise, Columns 4 and 5 summarize the associated delays and savings. Column 6 reports the percentage of the operations that are shifted from their initial FCFS position in the sequence, whereas the last column provides the number of aircraft position shifts in the FCFS-OPT and OPT solutions from the sequence produced with FCFS-MIX. Both the optimal sequence (OPT) and our proposed heuristic (FCFS-OPT) resulted in less than 2 aircraft position shifts at an average, per shifted operation (Figure 2.7). That is, optimal or near-optimal schedules can be achieved via very limited position shifts, which largely preserves the FCFS-MIX sequence. This is due to the fact that in presence of multiple runways assignment of operations has a crucial impact on the final excess fuel cost than that of the sequencing strategy. This indicates that, under data trends at Doha, our proposed FCFS-OPT heuristic not only ensures fairness amongst aircraft by exhibiting limited deviation from FCFS-MIX, but also empirically provides optimal or near-optimal results with respect to the fuel burn cost.
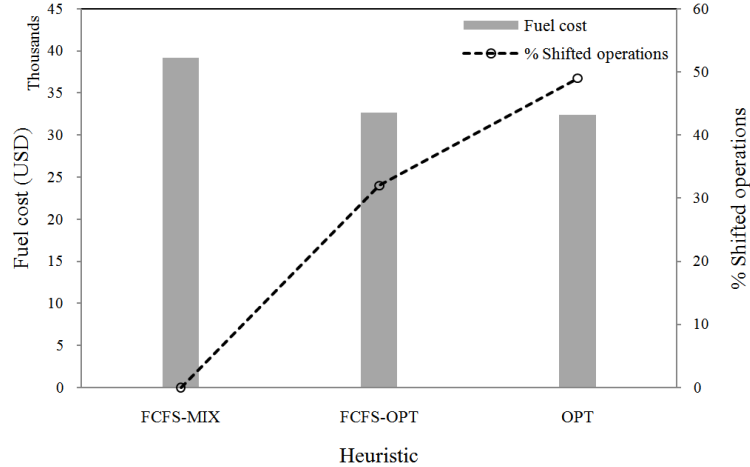
Figure 2.7: Comparison of Heuristic Results on DOH Projected Data

Further, we assessed the benefit of adopting two runways vs. a single runway. By transitioning operations from DOH to HIA, an anticipated savings of nearly 3 million dollars per day can be achieved. Our results also indicate that a segregated mode, as in FCFS-SEG, results in over 240 thousand dollars of excess fuel cost per day, as opposed to 39,221 dollars under a mixed mode. Under increasingly higher volumes of aircraft movements, especially when arrival and departure peaks are not occurring during the same time-windows, a mixed mode utilization of the runways can yield significant fuel savings.

We also examined in the anticipated gains accruing from the adoption of the FAA aircraft separation standard in lieu of the ICAO standard. Our results suggest that substantial reductions in fuel cost and average delays can be achieved using the FAA standard. Although using the FAA standard does not necessarily result in important fuel costs and delay reductions in every time-window of the day, it is overall very beneficial at the aggregate level as depicted in Figure 2.8. Limited savings with FAA usually occur when the mix of aircraft weight classes involves a significant proportion of small/large aircraft that follow heavy aircraft, which requires slightly larger separation times under the FAA standard.

36
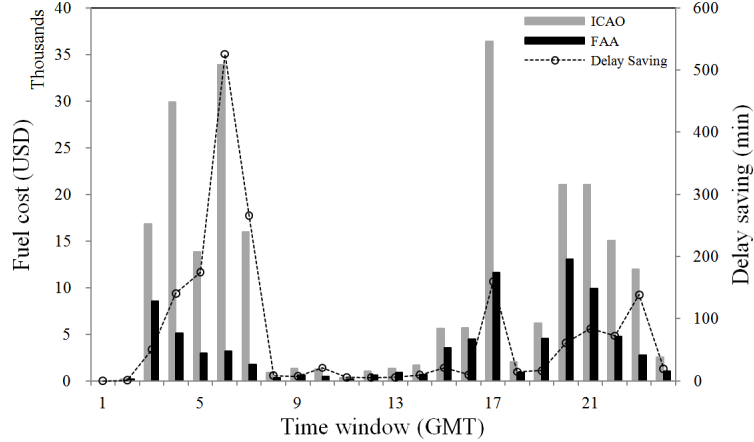
Figure 2.8: Benefits of FAA Standard over ICAO Standard (FCFS-SEG setting)

2.5. Conclusion

This chapter investigated a three-faceted approach for the assessment of runway management strategies, whereby an airport can strategically evaluate the combined effect of its runway physical configuration, a candidate aircraft sequencing policy, and a chosen aircraft separation standard, using optimization methodology. The chapter makes the following contributions:

- Using data from Qatar Civil Aviation Authority, our study quantifies the savings and demonstrates that the transition from a single runway with a nominal capacity of 30 arrivals per hour, as in the Doha International Airport, to two parallel independent runways with the nominal capacity of 60 arrivals per hour, as planned in the new Hamad International Airport (HIA), would achieve nearly $3 million savings per day in excess fuel burn cost. This, however, also revealed that the excess fuel cost or delays would not be completely eliminated, even under a two-runway configurations, whether used in segregated or mixed mode, because of the uneven distribution of operations throughout the day. It further highlights the necessity of examining enhanced sequencing policies and alternative aircraft separation times in order to better exploit the runway capacity.

37

Although the nominal runway capacity may not be reached during many hours of activity throughout the day, certain time-windows of the day are particularly congested and witness excess fuel cost and delays. Such busier hours typically can benefit from optimized aircraft assignment and sequencing procedures. In this regard, major airports can benefit from providing incentives to airlines to shift some of their operations to less congested hours of the day to more uniformly avail of the capacity of the airport.

- We developed an optimization-based heuristic which is based on the FCFS sequencing policy. We find that by slightly altering the FCFS sequence, the proposed heuristic not only preserves fairness among aircraft, but also consistently produces excellent (optimal or near optimal) solutions. Without deviating aircraft by not more than 2 positions from their FCFS sequence positions, the objective value produced by the proposed heuristic deviated by less than 1% from the optimal objective value found using a mixed-integer program.

- Our empirical results also indicate that international airports such as the Hamad International Airport can significantly benefit from using the FAA aircraft separation standard in lieu of the ICAO standard. In the specific case of HIA, this choice is expected to achieve nearly a 50% reduction in excess fuel cost.

Although illustrated with real data for Doha International Airport, the approach presented in the chapter and the proposed heuristic can be of general benefit to other airports, especially during busier hours of activity during the day. The anticipated savings in fuel costs can directly benefit airlines, airports, and governmental authorities that are concerned with environmental effects and emissions. We recommend for further investigation an analysis of the impact of alternative runway settings on additional airborne or ground-based operations related to taxiway routing, gate assignments, and workload at terminals.

CHAPTER 3

MULTIPLE-RUNWAY AIRCRAFT SEQUENCING PROBLEMS: ENHANCED FORMULATION AND ACCELERATED COLUMN GENERATION APPROACH

This chapter examines aircraft sequencing problems over multiple runways under mixed mode operations. Crafting valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints yields computational savings over a base mixed-integer formulation using a branch-and-bound/cut technique. To further enhance its computational tractability, the problem is alternatively reformulated as a set partitioning model with one side constraint that prompts the development of a specialized column generation approach. The latter is accelerated by incorporating several algorithmic features, including an interior point dual stabilization scheme, a complementary column generation routine, and a dynamic lower bounding feature. Empirical results using a set of computationally challenging simulated instances demonstrate the effectiveness and the relative merits of the strengthened mixed-integer formulation and the accelerated column generation approach.

3.1. Introduction & Motivation

Air traffic loads have been on the rise over the last several decades and are expected to double, and possibly triple in some regions, over the coming decade (Bennell et al., 2011). This growth in air traffic volumes has not been accompanied by appropriate capacity expansion in the air transport infrastructure. It is, therefore, predicted that flight delays costing multi-billion dollars will continue to negatively impact airline

companies and consumers. This motivates our research on developing efficient procedures to manage bottleneck operations and existing scarce resources at airports such as runways.

More specifically, this chapter studies modeling and solution methodology enhancements for an aircraft sequencing problem (ASP) over multiple *independent* runways, under mixed mode operations (both landings and departures). First, we examine a disjunctive-based 0-1 mixed-integer program (MIP) that simultaneously assigns aircraft to runways and ascertains optimal sequences for aircraft that are assigned to the same runway. The assignment and sequencing decisions are performed while complying with individual aircraft operation time-windows and minimal safety separation times amongst aircraft as imposed by aviation authorities to preclude wake vortex turbulence effects. Various preprocessing routines and symmetry-defeating constraints are proposed in order to enhance the computational tractability of this model. We also reformulate the problem as a set partitioning problem with a side constraint that prompts a specialized column generation approach. The latter is substantially accelerated using a combination of algorithmic schemes such as an interior point dual stabilization scheme, a complementary column generation feature, and a dynamic lower bounding feature in the subproblem that are tested in isolation and in a synergistic fashion.

The literature is replete with operations research approaches to aircraft sequencing problems, spanning complexity results, exact algorithms, and heuristics. The recent survey by Bennell et al. (2011) indicates that greater attention has been given to single-runway aircraft landing problems and that dynamic programming algorithms, Branch-and-Bound (B&B) algorithms, and metaheuristics are the most commonly used solution approaches. MIP models have been proposed in Abela et al. (1993), Ernst et al. (1999), and Beasley et al. (2000) for static single-runway aircraft sequencing problems. In Ernst et al. (1999), the multiple-runway case was also examined and

a B&B algorithm was proposed with root node preprocessing routines. Beasley et al. (2000) also extended their single-runway model to the case of multiple interdependent runways, with the objective of minimizing a weighted deviation from specified target landing times.

Also of interest in our study is the work by Wen et al. (2005), where the authors addressed the aircraft sequencing problem over multiple independent runways following the framework in Beasley et al. (2000) using a branch-and-price algorithm. As pointed out by the authors, although the column generation approach expectedly produced very tight lower bounds on the objective value, it was not tractable enough as it required computational times substantially greater than what is reported in the literature for the instances tested in Beasley et al. (2000). This is particularly impractical when it is noted that the aforementioned instances are mostly solvable in a fraction of a CPU second using recent versions of commercial solvers. It is also worthwhile to note that the multiple-runway aircraft sequencing problem under investigation bears similarities with parallel machine scheduling problems with time-windows and sequence-dependent processing times or the m-Asymmetric Traveling Salesman Problem with time-windows. Chen and Powell (1999) investigated parallel (identical, uniform, and unrelated) machine scheduling problems with the objective of minimizing the total weighted completion time and the weighted number of tardy jobs using a branch-and-price algorithm.

Beyond the specific confines of the application under investigation, there have been active efforts to mitigate two notorious drawbacks of column generation: a) tailing-off phenomena that are symptomatic of a slow convergence of simplex-based column generation approaches; and b) the difficulty of producing near-optimal integer solutions. Tailing-off effects have been combated using stabilization techniques, as discussed in du Merle et al. (1999) and Lübbecke and Desrosiers (2005) (also see encouraging results for the cutting stock problem in de Carvalho (2005) and Clautiaux

et al. (2011)). In particular, Rousseau et al. (2007) proposed an interior point dual stabilization (IPDS) technique that does not require elaborate parameter calibrations and that has been empirically observed to outperform the stabilization technique by Neame (1999) and to be comparably competitive to the box-penalty method in du Merle et al. (1999) for the vehicle routing problem with time-windows. Likewise, Gondzio et al. (2013) recently developed a primal-dual interior point method in order to overcome the unstable behavior of standard column generation approaches and to curtail the number of iterations performed. Furthermore, Subramanian and Sherali (2008) proposed a deflected subgradient scheme to mitigate dual noise and accelerate a column generation approach for large-scale airline crew planning problems. Such stabilization procedures tend to be successful in reducing the number of iterations performed and the overall computational effort in the linear programming phase, but without necessarily producing good quality integer solutions.

As far as producing optimal (or near-optimal) integer solutions was concerned, branch-and-price algorithms Barnhart et al. (1998) have typically been the method of choice. However, Ghoniem and Sherali (2009) introduced a complementary column generation (CCG) feature that consistently yielded optimal and near-optimal integer solutions to set partitioning problems that are solved by column generation without resorting to developing a branch-and-price algorithm. The CCG feature was shown to produce excellent duality-based gaps for bin packing problems, vehicle assembly-routing problems (Sherali and Ghoniem, 2009; Ghoniem and Sherali, 2009), and set packing problems (Ghoniem and Sherali, 2010). However, to the best of our knowledge, the IPDS technique in the spirit of Rousseau et al. (2007) and the CCG feature have not been empirically compared, nor has the synergistic benefit of jointly using them been examined. The present study precisely addresses this matter in the context of the multiple-runway aircraft sequencing problem. Furthermore, we

introduce a dynamic lower bounding (DLB) feature that can be applied to further accelerate the proposed column generation algorithm.

This chapter makes the following contributions. First, it demonstrates how valid inequalities, preprocessing schemes, and lexicographic symmetry-defeating constraints can greatly improve the computational tractability of the multiple-runway aircraft sequencing problem for difficult problem instances when used in isolation or in concert with each other. Second, the chapter demonstrates the relative merits and synergistic gains afforded by three column generation acceleration schemes. Third, our study empirically reveals which problem instance sizes in our test-bed were more efficiently solved using an enhanced MIP formulation or an accelerated column generation approach.

The remainder of this chapter is organized as follows. Section 3.2 develops an MIP formulation for the multiple-runway aircraft sequencing problem along with valid inequalities, preprocessing routines, and symmetry-defeating constraints. Thereafter, Section 3.3 introduces an alternative set partitioning reformulation of the problem and delineates a column generation approach that is enhanced with three algorithmic features. Section 3.4 reports our empirical results over a set of computationally challenging problem instances and highlights the relative merits and limitations of the proposed solution methodologies. Section 3.5 concludes this work with a summary of our findings and directions for future research.

3.2. Enhanced Mathematical Programming Formulation

This section provides a 0-1 MIP formulation for the multiple-runway aircraft sequencing problem along with modeling enhancements.

3.2.1 Notation and MIP Formulation

Given a set of m identical runways and n aircraft with input data on aircraft types (heavy/large/small), operation types (arrival/departure), ready-times and due-times, and the minimum separation times enforced by aviation authorities, the multiple-runway aircraft sequencing problem seeks to jointly assign the aircraft to runways and to determine the best aircraft sequence for each runway with respect to a chosen objective function. In doing so, time-window restrictions must be met for all aircraft and minimal safety separation times need to be enforced between any pair of aircraft that are assigned to the same runway. Consider the following notation:

Index Sets and Parameters:

- $M = \{1, \ldots, m\}$: A set of m identical runways.

- $J = \{1, \ldots, n\}$: A set of n aircraft (landings or departures).

- r_j : Ready-time for aircraft j, $\forall j \in J$.

- d_j : Due-time for aircraft j, $\forall j \in J$.

- $\mathcal{O}_j$: Operation type of aircraft j, being a landing or a departure, $\forall j \in J$.

- $\mathcal{C}_j$: Weight class of aircraft j, e.g., heavy, large, or small, $\forall j \in J$.

- w_j : Weight assigned to aircraft j based on its operation type and its weight class, $\forall j \in J$. In particular, higher priority has been assigned to landings over departures and to heavy aircraft over large and small ones. Moreover, in our test-bed $w_{j_1} = w_{j_2}$ if $O_{j_1} = O_{j_2}$ and $\mathcal{C}_{j_1} = \mathcal{C}_{j2}$.

- $p_{j_1 j_2}$: Minimum separation time required between aircraft j_1 and j_2 if they are assigned to the same runways and respectively the leading and the following

aircraft, $\forall j_1, j_2 \in J, j_1 \neq j_2$. This separation time is dictated by the operation types and weight classes of any pair of aircraft and are typically asymmetric.

Decision Variables:

- t_j : the start time of aircraft j, $\forall j \in J$.

- $z_{ij} \in \{0, 1\}$: $z_{ij} = 1$ if and only if aircraft j is assigned to runway i, $\forall i \in M, j \in J$.

- $y_{j_1 j_2} \in \{0, 1\}$: $y_{j_1 j_2} = 1$ if and only if aircraft j_1 and j_2 are assigned to the same runway and $t_{j_2} > t_{j_1}$, $\forall j_1, j_2 \in J, j_1 \neq j_2$.

The multiple-runway aircraft sequencing problem, denoted by **MRASP**, is formulated as the following 0-1 mixed-integer program:

$$\textbf{MRASP: Minimize } \sum_{j \in J} w_j t_j \tag{3.1a}$$

$$\sum_{i \in M} z_{ij} = 1, \quad \forall j \in J \tag{3.1b}$$

$$\sum_{j \in J} z_{ij} \leq \left\lceil \frac{n}{m} \right\rceil, \quad \forall i \in M \tag{3.1c}$$

$$r_j \leq t_j \leq d_j, \quad \forall j \in J \tag{3.1d}$$

$$t_{j_2} \geq t_{j_1} + p_{j_1 j_2} - (1 - y_{j_1 j_2})(d_{j_1} - r_{j_2} + p_{j_1 j_2}),$$
$$\forall j_1 \in J, j_2 \in J, j_1 \neq j_2 \tag{3.1e}$$

$$y_{j_1 j_2} + y_{j_2 j_1} \geq z_{ij_1} + z_{ij_2} - 1, \quad \forall i \in M, j_1 \in J, j_2 \in J, j_1 < j_2 \tag{3.1f}$$

$$y, z \text{ binary.} \tag{3.1g}$$

The objective function (3.1a) minimizes total weighted start times. Constraint (3.1b) ensures that every aircraft is assigned to exactly one of the m runways, whereas

Constraint (3.1c) bounds above the number of aircraft that are assigned to any runway. Time-window restrictions are introduced in Constraint (3.1d). Constraint (3.1e) guarantees proper separation between any pair of aircraft that are assigned to the same runway. Constraint (3.1f) activates the sequencing variables between any pair of aircraft that are assigned to the same runway. Constraint (3.1g) defines binary decision variables.

3.2.2 Enhancing Constraints for MRASP

In this section, we present valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints that were found to be computationally advantageous for MRASP.

3.2.2.0.1 Proposition 1 Constraint (3.2) is a valid inequality for MRASP:

$$y_{j_1 j_2} + y_{j_2 j_1} \leq 1 - \frac{1}{m-1} \left| \sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \right|, \quad \forall j_1 \in J, j_2 \in J, j_1 < j_2. \qquad (3.2)$$

Proof. The term $\left| \sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \right|$ expresses the absolute difference between the runway indices to which aircraft j_1 and j_2 are assigned. If $z_{ij_1} = z_{ij_2}$ for some runway i, then this term equals 0, and Constraint (3.2) in conjunction with Constraint (3.1f) would require $y_{j_1 j_2} + y_{j_2 j_1} = 1$. However, if $z_{ij_1} \neq z_{ij_2}, \forall i$, i.e., j_1 and j_2 are not assigned to the same runway, then $0 < \left| \sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \right| \leq m - 1$, i.e., $0 < \frac{1}{m-1} \left| \sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \right| \leq 1$ and Constraint (3.2) forces $y_{j_1 j_2} + y_{j_2 j_1}$ to equal 0. $\square$

Note that the Constraint (3.2) introduces nonlinearities in the model, which can be obviated by noting that

$$\left| \sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \right| = \max\{ \sum_{i \in M} i(z_{ij_1} - z_{ij_2}), -\sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \}$$

and, thereby, substituting Constraints (3.3a)-(3.3b) in lieu of Constraint (3.2):

$$\sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \leq (m-1)(1 - y_{j_1 j_2} - y_{j_2 j_1}), \quad \forall j_1 \in J, j_2 \in J, j_1 < j_2 \qquad (3.3a)$$

$$\sum_{i \in M} i(z_{ij_1} - z_{ij_2}) \geq (1-m)(1 - y_{j_1 j_2} - y_{j_2 j_1}), \quad \forall j_1 \in J, j_2 \in J, j_1 < j_2. \qquad (3.3b)$$

3.2.2.0.2 Proposition 2 By preprocessing, to ensure feasibility, we can enforce:

$$y_{j_2 j_1} = 0, \quad \forall i \in M, j_1 \in J, j_2 \in J, j_1 \neq j_2, r_{j_2} + p_{j_2 j_1} > d_{j_1}. \qquad (3.4)$$

Proof. If scheduling j_2 before j_1 would cause the latter to be infeasible ($r_{j_2} + p_{j_2 j_1} > d_{j_1}$), then j_2 cannot precede j_1, and hence $y_{j_2 j_1} = 0$. □

3.2.2.0.3 Proposition 3 By preprocessing, without loss of optimality, we can impose:

$$y_{j_2 j_1} = 0, \quad \forall i \in M, j_1 \in J, j_2 \in J, j_1 \neq j_2,$$
$$r_{j_1} < r_{j_2}, d_{j_1} \leq d_{j_2}, \mathcal{O}_{j_1} = \mathcal{O}_{j_2}, \mathcal{C}_{j_1} = \mathcal{C}_{j_2}. \qquad (3.5)$$

Proof. If $z_{ij_1} \neq z_{ij_2}, \forall i$, then this requirement holds. In addition, if aircraft j_1 and j_2 have the same weight class and the same operation type (being landings or departures), satisfy $r_{j_1} < r_{j_2}, d_{j_1} \leq d_{j_2}$, and are assigned to the same runway, then aircraft j_1 can be forced to precede j_2 without loss of optimality. In fact, in any optimal schedule where j_2 precedes j_1 under the aforementioned conditions, swapping the positions of j_1 and j_2 does not impact the separation times amongst aircraft or the objective value and, therefore, yields an alternative optimal solution. □

In the context of independent runways, there exists an inherent symmetry amongst aircraft, which introduces a significant computational burden on B&B/C solvers

(Sherali and Smith, 2001). To combat the symmetric reflections of feasible solutions, it is judicious to impose symmetry-defeating hierarchical constraints that aim at imparting distinct identities to the otherwise indistinguishable runways as follows:

$$\sum_{j \in J} a_j z_{i,j} \geq \sum_{j \in J} a_j z_{i+1,j}, \quad \forall i = 1, \ldots, m-1, \tag{3.6}$$

where a_j is a coefficient that can be specified to enforce different possible rank orderings of the runways. For example, by setting $a_j = 1$, Constraint (3.6) simply ranks the runways in the order of non-increasing number of aircraft assigned to the runways. However, the latter provides a weak differentiation amongst aircraft. The following lexicographic-based ordering provides a more useful device to defeat symmetry:

$$\sum_{j \in J} 2^{n-j} z_{i,j} \geq \sum_{j \in J} 2^{n-j} z_{i+1,j}, \quad \forall i = 1, \ldots, m-1. \tag{3.7}$$

Preliminary computational results suggest that the alternative formulation of Constraint (3.7) as in Constraint (3.8) (Ostrowski et al., 2010) was not advantageous for MRASP.

$$\sum_{v=i}^{\min(j,m)} z_{v,j} \leq \sum_{u=1}^{j-1} z_{i-1,u}, \quad \forall i = 2, \ldots, m, \, j \in J, j \geq i. \tag{3.8}$$

3.3. Accelerated Column Generation Approach

In this section, the MRASP is alternatively reformulated as a set partitioning model with a side constraint that is solved by column generation. We discuss different algorithmic features, including an interior point dual stabilization strategy, a complementary column generation scheme, and a dynamic lower bounding feature, which can be used in isolation or in concert with each other to accelerate the convergence of the column generation method.

3.3.1 Set Partitioning Reformulation

To formally introduce the proposed column generation approach, consider the following column construct, P^h, which is associated with a runway:

$$P^h = \begin{matrix} j=1 \\ j=2 \\ \vdots \\ j \\ \vdots \\ j=n \end{matrix} \begin{bmatrix} \\ \\ \\ (P_j^h) \\ \\ \\ \end{bmatrix} , \text{ where } P_j^h = \begin{cases} 1, & \text{if aircraft } j \text{ is included} \\ & \text{in column } h, \forall j \in J \\ 0, & \text{otherwise.} \end{cases}$$

Note that the column P^h is an n-column vector having 0-1 entries, where $P_j^h = 1$ equals 1 if and only if aircraft j is assigned to this column. Let c_h be the total cost for P^h, which reflects the cost in the optimized aircraft schedule associated with this particular runway, P^h. Thus, aircraft assignment decisions are captured by the 0-1 entries of a column, whereas the accompanying aircraft sequencing decisions are reflected in the column cost itself. We then define the decision variable x_h as

$$x_h = \begin{cases} 1 \text{ if column } h \text{ is selected,} \\ 0 \text{ otherwise, } \forall h = 1, ..., H, \end{cases}$$

where H is the total number of columns for the problem instance under investigation. Accordingly, the following set partitioning problem, denoted by **SPP**, provides an alternative formulation for the MRASP:

$$\textbf{SPP: Minimize } \sum_{h=1}^{H} c_h x_h \tag{3.9a}$$

$$\text{subject to } \sum_{h=1}^{H} P_j^h x_h = 1, \quad \forall j = 1, \ldots, n \tag{3.9b}$$

$$\sum_{h=1}^{H} x_h = m \tag{3.9c}$$

$$x \text{ binary.} \tag{3.9d}$$

The objective function (3.9a) minimizes the total schedule cost (the total weighted start times). Constraint (3.9b) achieves a set partitioning scheme for aircraft, guaranteeing that every aircraft is assigned to exactly one runway, whereas Constraint (3.9c) enforces an upper bound on the number of available runways. To alleviate the computational burden created by the huge number of possible columns, it is usually worthwhile to construct a restricted master program (RMP) that is associated with SPP. This RMP initially includes some selected set of $\hat{H}(\leq H)$ columns that yield a feasible solution to SPP, which get dynamically expanded by coordinating the linear programming (LP) relaxation of the RMP and a subproblem that identifies columns having a most negative reduced cost, until the LP relaxation of SPP is solved to optimality. A 0-1 solution to SPP would then need to be determined.

Consider the following notation for a formal statement of the subproblem:

- $z_j \in \{0, 1\}$: $z_j = 1$ if and only if aircraft j is selected in the column constructed by the subproblem, $\forall j \in J$.

- π: vector of dual variables associated with the set partitioning constraints in (3.9b), where $\pi = \bar{\pi}$ represents specific dual variable values obtained at a given iteration in the course of the column generation approach.

- π_0: dual variable associated with the upper bounding constraint (3.9c), where $\pi_0 = \bar{\pi}_0$ represents a specific dual variable value obtained at a given iteration in the course of the column generation approach.

The subproblem, denoted by $\mathbf{SP}(\bar{\pi}, \bar{\pi}_0)$, is defined as follows:

$$\mathbf{SP}(\bar{\pi}, \bar{\pi}_0)\text{: Minimize } RC \equiv \sum_{j \in J}(w_j t_j - \bar{\pi}_j z_j) - \bar{\pi}_0 \tag{3.10a}$$

$$\sum_{j \in J} z_j \leq \left\lceil \frac{n}{m} \right\rceil, \quad \forall i \in M \tag{3.10b}$$

$$r_j z_j \leq t_j \leq d_j z_j, \quad \forall j \in J \tag{3.10c}$$

$$t_{j_2} \geq t_{j_1} + p_{j_1 j_2} y_{j_1 j_2} - (1 - y_{j_1 j_2}) \max_{j \in J}\{d_j\},$$

$$\forall j_1 \in J, j_2 \in J, j_1 \neq j_2 \tag{3.10d}$$

$$y_{j_1 j_2} + y_{j_2 j_1} \geq z_{j_1} + z_{j_2} - 1, \quad \forall i \in M, j_1 \in J, j_2 \in J, j_1 < j_2 \tag{3.10e}$$

$$y_{j_1 j_2} + y_{j_2 j_1} \leq z_{j_1}, \quad \forall i \in M, j_1 \in J, j_2 \in J, j_1 \neq j_2 \tag{3.10f}$$

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2, r_{j_2} + p_{j_2 j_1} > d_{j_1} \tag{3.10g}$$

$$y_{j_2 j_1} = 0, \quad \forall j_1 \in J, j_2 \in J, j_1 \neq j_2,$$

$$r_{j_1} < r_{j_2}, d_{j_1} \leq d_{j_2}, \mathcal{O}_{j_1} = \mathcal{O}_{j_2}, \mathcal{C}_{j_1} = \mathcal{C}_{j_2} \tag{3.10h}$$

$$y, z \text{ binary.} \tag{3.10i}$$

3.3.2 Interior Point Dual Stabilization

Column generation techniques tend to exhibit slow convergence and long tailing-off effects that are largely due to degeneracy in the restricted master program. This issue can be particularly acute when multiple dual solutions exist for a given primal solution to the continuous relaxation of the RMP. Several stabilization techniques have been

proposed to mitigate this drawback by carefully guiding the selection of the dual variable values as these greatly impact the columns constructed in the subproblem and the overall column generation performance. Rousseau et al. (2007) proposed a so-called interior point stabilization technique that advocates, at every iteration, the construction of a vector of dual values as a convex combination of multiple optimal dual solutions. This generally tends to yield an interior point of the optimal dual polyhedron, and avoids the generation of an extreme point of this polyhedron that could typically exhibit unbalanced dual values.

We present the interior point stabilization technique in the context of the set partitioning model with a side constraint for the multiple-runway aircraft sequencing problem. To this end, we first reformulate the restricted set partitioning model (3.9) as the following restricted set covering problem with a side constraint (**SCP-P**):

$$\textbf{SCP-P}: \text{Minimize} \ \sum_{h=1}^{\hat{H}} c_h x_h \tag{3.11a}$$

$$\text{s.t.} \ \sum_{h=1}^{\hat{H}} P_j^h x_h \geq 1, \quad \forall\, j \in J \tag{3.11b}$$

$$- \sum_{h=1}^{\hat{H}} x_k \geq -m \tag{3.11c}$$

$$x \geq 0. \tag{3.11d}$$

At every iteration, upon solving SCP-P, the following sets K^* and J^* are determined:

- $K^* = \{\, k \in \{1, \ldots, \hat{H}\} \mid x_k > 0\}$.

- $J^* = \{\, j \in J \mid \bar{\pi}_j = 0\}$, for which Constraint (3.11b) is not binding.

Using complementary slackness, any dual vector π that satisfies Constraints (3.12b)-(3.12e) qualifies as a dual optimum at the current iteration. Hence, to generate different extreme points of the dual polyhedron (**SCP-D**), the objective function (3.12a) employs a vector u, where u_j is randomly generated between 0 and 1:

$$\textbf{SCP-D}(\bar{\pi}_0) : \text{Maximize } (\sum_{j \in J} u_j \pi_j) - \bar{\pi}_0 \tag{3.12a}$$

$$\text{s.t.} \sum_{j \in J} P_j^k \pi_j - \bar{\pi}_0 = c_k, \quad \forall\, k \in K^* \tag{3.12b}$$

$$\sum_{j \in J} P_j^k \pi_j - \bar{\pi}_0 \le c_k, \quad \forall\, k \in \{1, \ldots, \hat{H}\} \backslash K^* \tag{3.12c}$$

$$\pi_j = 0, \quad \forall j \in J^* \tag{3.12d}$$

$$\pi_j \ge 0, \quad \forall j \in J, \tag{3.12e}$$

where $\bar{\pi}_0$ is the dual variable value associated with Constraint (3.11c).

By solving SCP-D ρ times, a set of extreme points of the dual polyhedron is generated iteratively, denoted $\bar{\pi}^1, \ldots, \bar{\pi}^\rho$, and we construct the interior point dual solution $\bar{\pi} = \frac{1}{\rho} \sum_{k=1}^{\rho} \bar{\pi}^k$ as an average of the extreme points generated. Noting that $(\bar{\pi}, \bar{\pi}_0)$ is feasible to (3.12b)-(3.12e) and is hence dual optimal, this vector of dual values is passed on to the subproblem. In Rousseau et al. (2007), setting $\rho = 20$ empirically yielded attractive computational results for the vehicle routing with time-windows instances, whereas $\rho = 5$ yielded the best results for MRASP in our experience.

3.3.3 Complementary Column Generation

Upon solving the LP relaxation of SPP to optimality, a strong lower bound is typically produced. It is, however, more challenging to produce a high quality feasible

0-1 solution to SPP. Ghoniem and Sherali (2009) proposed the use of a complementary column generation feature that has been shown to produce excellent feasible 0-1 solutions to set partitioning problems with side constraints. We exploit this feature in the context of the multiple-runway aircraft sequencing problem. In every LP iteration, upon finding a column that prices out favorably in the subproblem, the CCG feature advocates the construction of an additional set of complementary columns by resolving the subproblem sequentially, whereby aircraft that appear in any column produced in the LP iteration get excluded from the aircraft set. Let $\mathcal{F}$ be a set of temporarily forbidden aircraft that is used to implement the CCG. The overall CCG approach is delineated as follows:

Algorithm 1: COMPLEMENTARY COLUMN GENERATION

1: Initialize RMP with $\hat{H} = m$ columns using a feasible solution to SPP.
2: **repeat**
3:     Solve the LP relaxation of RMP.
4:     Update $\bar{\pi}$ and $\bar{\pi}_0$.
5:     Solve $SP(\bar{\pi}, \bar{\pi}_0)$.
6:     Let $\bar{z}$ and $\bar{t}$ correspond to the constructed column and the associated start times.
7:     **if** $RC < 0$ **then**
8:         $\hat{H} \leftarrow \hat{H} + 1$, $P_j^{\hat{H}} = \bar{z}_j$, $c^{\hat{H}} = \sum_{j \in J} w_j \bar{t}_j$
9:         $\mathcal{F} \leftarrow \{j \in J : \bar{z}_j = 1\}$
10:        $flag = 0$
11:        Begin CCG
12:        **while** $J \neq \mathcal{F}$ and $flag = 0$ **do**
13:           Solve the $SP(\bar{\pi}, \pi_0)$ with the additional constraint $\sum_{j \in \mathcal{F}} z_j = 0$.
14:          **if** SP is feasible with $\bar{z}^{new}, \bar{t}^{new}$ **then**
15:            $\hat{H} \leftarrow \hat{H} + 1$, $P_j^{\hat{H}} = \bar{z}_j^{new}$
16:            $c^{\hat{H}} = \sum_{j \in J} w_j \bar{t}_j^{new}$, and $\mathcal{F} \leftarrow \mathcal{F} \cup \{j \in J : \bar{z}_j^{new} = 1\}$
17:          **else**
18:            $flag = 1$
19:        End CCG
20: **until** $RC < 0$
21: Solve the RMP as a 0-1 program.

54

3.3.4 Dynamic Lower Bounds

High quality solutions are mostly composed of balanced runways, where each runway is assigned a number of aircraft close to $\left\lceil \frac{n}{m} \right\rceil$ on average. However, the column generation approach may generate at early iterations very sparse columns having a number of aircraft significantly lower than $\left\lceil \frac{n}{m} \right\rceil$ (and sometimes close to 1), due to their attractive reduced costs. We, therefore, propose a dynamic lower bounding (DLB) scheme that can be triggered in addition to the CCG feature presented in Section 3.3.3. In every LP iteration, if the RMP contains less than δ columns, where δ is a user-specified threshold, then we do the following:

- Append Constraint (3.13) to the subproblem and solve it to generate a first column that prices out favorably:

$$\sum_{j \in J} z_j = \left\lfloor \frac{n}{m} \right\rfloor \tag{3.13}$$

  If no negative reduced-cost column is produced, the DLB routine is terminated.

- For the kth complementary column generated by the CCG feature, where k runs between $1, \ldots, m-2$, we impose the following lower bounding constraint in order to balance the assignment of the remaining aircraft across the next columns produced by the SP:

$$\sum_{j \in J \setminus \mathcal{F}} z_j \geq \left\lfloor \frac{|J \setminus \mathcal{F}|}{m-k} \right\rfloor - 1. \tag{3.14}$$

- For the last complementary column, we simply impose:

$$\sum_{j \in J \setminus \mathcal{F}} z_j = 1. \tag{3.15}$$

55

That is, in every LP iteration, the subproblem is solved iteratively (as for the CCG feature), with a lower bounding requirement to identify relatively dense columns and possibly create a full block of m columns. The DLB scheme is terminated when the RMP comprises a number of columns greater than or equal to some user-specified δ or when no negative reduced-cost first column could be produced in a given LP iteration. At such a point, the algorithm would have sufficiently progressed and the subproblem tends to generate attractive assignments of aircraft to runways until convergence to an optimal LP solution is obtained.

3.4.   Computational Study

In this section, we first discuss computational results obtained for the 0-1 MIP formulation with valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints, followed by an examination of the accelerated column generation approach.

3.4.1   Data Generation

We created a test-bed that includes 11 problem sizes characterized by the number of aircraft, n, and the number of runways, m: 15-aircraft instances with 2, 3, and 4 runways, 20-aircraft instances with $2, \ldots, 5$ runways, and 25-aircraft instances with $2, \ldots, 5$ runways. For each (n, m) combination, 5 instances were generated, resulting in a total of 55 instances which are available online at:

http://ahmed.ghoniem.info/download/MASP-SET.txt.

We also implemented the MIP formulation in Beasley et al. (2000) and tested it along with MRASP on the instances by Beasley (see Appendix A). The proposed MIP formulation and column generation approach were coded with AMPL and were solved using CPLEX 12.4 on a Windows 7 professional 64-bit operating system with

an Intel Core i7-2600 CPU with 3.40 GHz and 12 GB RAM desktop. A time limit of 3600 CPU seconds was imposed on all runs.

The ready-times were randomly generated using a discrete uniform distribution over the interval $(0, \gamma \frac{n}{m})$, where γ is a parameter that was randomly selected in the interval (30,90) to simulate the inter-arrival time between aircraft. The operation types (Arrival/Departure) were randomly assigned to aircraft using a discrete uniform distribution with equal likelihood $(\frac{1}{2}/\frac{1}{2})$. Likewise, aircraft weight classes (Heavy/Large/Small) were randomly assigned to aircraft using a discrete uniform distribution with equal likelihood $(\frac{1}{3}/\frac{1}{3}/\frac{1}{3})$. The weight w_j was introduced as a function of the aircraft operation type and its weight class. In our study, the greatest weight of 6 was assigned to heavy arrivals and the least weight of 1 was given to small departures. Every aircraft was prescribed a time-window of 600 seconds.

3.4.2 Results for the Enhanced MIP Model

Table 3.1 reports our computational results for the MIP model, MRASP, and its enhanced variants. The first two columns specify the number of aircraft, n, and the number of runways, m. The following MIP model variants considered in our computational study are reported in the third column of Table 3.1:

- MRASP is the base model introduced in Section 3.2.1, Equations (3.1a)-(3.1g).

- + VI refers to the base model augmented with Constraints (3.3a)-(3.3b).

- + PREP refers to the base model strengthened with the preprocessing restrictions in Constraints (3.4)-(3.5).

- + SYM refers to the base model with the lexicographic-based symmetry-defeating hierarchical constraint in (3.6).

- + VI/PREP/SYM combines all the aforementioned enhancements.

Table 3.1: Results on the Enhanced MIP Model

n	m	MIP Model	Solved		CPU (s)	% CPU Time Red.	B&B/C Nodes	% B&B/C Node Red.
15	2	MRASP	5/5		874.7	-	9,856,702	-
		+VI	5/5		46.5	94.69	624,811	93.66
		+PREP	5/5		45.8	94.77	762,149	92.27
		+SYM	5/5		713.7	18.41	8,523,046	13.53
		+(VI/PREP/SYM)	5/5		**4.0**	99.54	38,203	99.61
	3	MRASP	5/5		224.8	-	3,255,207	-
		+VI	5/5		180.3	19.78	1,500,565	53.90
		+PREP	5/5		165.4	26.41	2,191,359	32.68
		+SYM	5/5		444.2	-97.62	5,002,711	-53.68
		+(VI/PREP/SYM)	5/5		**21.6**	90.39	179,567	94.48
	4	MRASP	2/5	$\geq$	2,757.5	-	12,744,464	-
		+VI	1/5	$\geq$	2,970.1	-	7,771,502	-
		+PREP	3/5	$\geq$	2,281.4	-	14,056,413	-
		+SYM	1/5	$\geq$	2,948.3	-	11,700,049	-
		+(VI/PREP/SYM)	5/5		**297.4**	89.22	1,992,002	84.37
20	2	MRASP	5/5		432.8	-	3,817,577	-
		+VI	5/5		21.4	95.05	109,588	97.13
		+PREP	5/5		9.9	97.71	101,361	97.34
		+SYM	5/5		110.9	74.38	1,033,455	72.93
		+(VI/PREP/SYM)	5/5		**7.7**	98.23	49,260	98.71
	3	MRASP	4/5	$\geq$	1,624.3	-	11,549,197	-
		+VI	4/5	$\geq$	1,643.6	-	3,372,009	-
		+PREP	5/5		431.7	73.42	3,332,702	71.14
		+SYM	3/5	$\geq$	2,017.9	-	8,827,096	-
		+(VI/PREP/SYM)	5/5		**313.9**	80.68	1,083,759	90.62
	4	MRASP	0/5	$\geq$	3,600.0	-	13,909,720	-
		+VI	0/5	$\geq$	3,600.0	-	4,358,216	-
		+PREP	0/5	$\geq$	3,600.0	-	12,481,349	-
		+SYM	0/5	$\geq$	3,600.0	-	10,968,404	-
		+(VI/PREP/SYM)	1/5	$\geq$	**3,092.0**	-	8,360,935	-
	5	MRASP	0/5	$\geq$	3,600.0	-	9,144,660	-
		+VI	0/5	$\geq$	3,600.0	-	3,566,620	-
		+PREP	0/5	$\geq$	3,600.0	-	8,667,972	-
		+SYM	0/5	$\geq$	3,600.0	-	8,056,089	-
		+(VI/PREP/SYM)	2/5	$\geq$	**2,780.3**	-	4,145,831	-
25	2	MRASP	5/5		15.6	-	118,702	-
		+VI	5/5		10.8	30.79	16,417	86.17
		+PREP	5/5		**3.0**	80.44	16,141	86.40
		+SYM	5/5		12.3	21.06	103,374	12.91
		+(VI/PREP/SYM)	5/5		4.7	69.63	9,795	91.75
	3	MRASP	4/5	$\geq$	791.6	-	773,087	-
		+VI	3/5	$\geq$	1,652.3	-	1,015,697	-
		+PREP	5/5		**124.2**	84.31	1,067,518	-38.09
		+SYM	4/5	$\geq$	827.6	-	809,042	-
		+(VI/PREP/SYM)	5/5		810.7	-2.41	602,539	22.06
	4	MRASP	4/5	$\geq$	**934.1**	-	4,298,233	-
		+VI	2/5	$\geq$	2,363.6	-	1,164,986	-
		+PREP	4/5	$\geq$	1,343.9	-	3,808,334	-
		+SYM	2/5	$\geq$	2,179.5	-	6,046,367	-
		+(VI/PREP/SYM)	2/5	$\geq$	2,249.1	-	1,145,874	-
	5	MRASP	1/5	$\geq$	3,282.6	-	6,805,936	-
		+VI	0/5	$\geq$	3,600.0	-	1,253,355	-
		+PREP	1/5	$\geq$	**3,077.3**	-	6,045,729	-
		+SYM	0/5	$\geq$	3,600.0	-	5,856,473	-
		+(VI/PREP/SYM)	0/5	$\geq$	3,600.0	-	1,551,552	-

The fourth column in Table 3.1 reports the number of instances that were solved to optimality within a time limit of 3600 CPU seconds. Columns 5-8 in Table 3.1 summarize the average CPU time in seconds over five instances, the % CPU time savings achieved by each model variant over the base model, the average number of B&B/C nodes, and the % savings in the number of B&B/C nodes achieved by the enhanced model variants over the base model.

Tested in isolation, the preprocessing routines in Model MRASP + PREP appear to most markedly improve over MRASP, achieving an average computational savings over 70% and an accompanying 57% reduction in the number of B&B/C nodes. In contrast, Model MRASP + VI exhibited mixed results, ranging from 94.7% CPU time savings for $(n, m) = (15, 2)$ to worsening the performance for instances with 25 aircraft. MRASP + SYM also produced mixed results, achieving 74.4% CPU time savings for $(n, m) = (25, 2)$, but causing a substantial CPU time increase for $(n, m) = (15, 3)$.

Crafting valid inequalities, preprocessing routines, or lexicographic-based symmetry-defeating constraints in isolation did not consistently curtail the computational effort. However, our computational experience reveals that there is a remarkable synergy between the proposed modeling enhancements. In fact, Model MRASP + (VI/PREP/SYM) achieves an average computational savings of 75.1% over the base model for $(n, m) = (15, 2/3/4)$, $(20, 2/3)$, and $(25, 2/3)$. However, as the number of aircraft and the number of runways increase, even this enhanced formulation becomes ineffective, especially, for $(n, m) = (20, 4/5)$ and $(25, 4/5)$.

3.4.3 Results for the Accelerated Column Generation Approach

Table 3.2 reports our computational results for the column generation approach with different acceleration and enhancement schemes. Columns 1-2 specify the number of aircraft, n, and the number of runways, m. The third column reports the best

CPU time in seconds that is achieved by the MIP variants reported in Table 3.1. Column 4 lists the different column generation approaches in our study:

- Algorithm CG: Basic column generation approach (see Section 3.3.1).

- Algorithm SCG: Column generation with interior point dual stabilization scheme (see Section 3.3.2).

- Algorithm CCG: Column generation approach with the complementary column generation feature (see Section 3.3.3).

- Algorithm SCCG: Column generation with interior point dual stabilization and complementary column generation combined.

- Algorithm SCCG-DLB: SCCG with the dynamic lower bounding feature. Parameter δ (for DLB termination) was empirically set to 50, 100, and 150 for $n = 15$, 20, and 25, respectively (see Section 3.3.4).

Columns 5-6 show the average CPU time in seconds achieved by the different approaches and the average CPU time savings achieved by the enhanced CG algorithms over the basic CG approach. Column 7 reports the gap between the lower bound by solving the LP relaxation of the set partitioning formulation and the best IP solution obtained respectively by the different CG approaches. Column 8 shows the average reduction in the % gap achieved by the different CG approaches over the base approach. Likewise, Columns 9-10 summarize the number of LP iterations and the % reduction in the number of LP iterations.

A few observations are in order. First, instances with $(n, m) = (15, 2/3)$, $(20, 2)$, and $(25, 2/3)$ were efficiently solved using the enhanced MIP formulation and did not call for the use of the CG approach. In contrast, for $(n, m) = (15, 4)$, $(20, 3/4/5)$, and $(25, 4/5)$, the CG approach, especially with the proposed acceleration schemes, offered an attractive alternative. SCCG-DLB solved the instances in our test-bed

Table 3.2: Results on the Column Generation Approach with Acceleration Schemes

n	m	Best MIP Model CPU (s)	Approach	CPU (s)	% red. CPU Time	% GAP	% red. gap	LP Iter.	% red. LP Iter.
15	2	4.0	CG	176.6	-	4.72	-	83	-
			SCG	167.9	4.92	6.98	-48.00	83	-0.48
			CCG	125.8	28.79	0.06	98.72	67	18.80
			SCCG	149.2	15.55	0.08	98.35	73	11.81
			SCCG-DLB	133.2	24.61	0.04	99.09	51	38.31
	3	21.6	CG	33.0	-	22.74	-	61	-
			SCG	47.5	-43.72	22.51	1.04	53	12.50
			CCG	33.3	-0.90	0.47	97.94	41	32.24
			SCCG	37.6	-13.73	0.03	99.86	34	44.41
			SCCG-DLB	41.0	-24.12	0.15	99.33	33	46.38
	4	297.4	CG	22.1	-	2.51	-	51	-
			SCG	38.7	-75.20	0.49	80.64	45	12.55
			CCG	22.9	-3.47	0.10	95.95	25	51.37
			SCCG	30.8	-39.56	0.11	95.52	24	53.33
			SCCG-DLB	29.2	-32.17	0.17	93.32	22	57.65
20	2	7.7	CG	558.3	-	5.30	-	166	-
			SCG	360.2	35.49	5.30	0.00	169.6	-2.17
			CCG	349.5	37.41	0.00	100.00	137.4	17.23
			SCCG	320.6	42.57	0.00	100.00	142	14.46
			SCCG-DLB	315.2	43.54	0.00	100.00	103.8	37.47
	3	313.9	CG	267.5	-	17.88	-	104	-
			SCG	301.0	-12.54	17.88	0.00	100	3.86
			CCG	235.9	11.80	0.41	97.68	75	27.61
			SCCG	203.1	24.06	0.48	97.29	62	39.96
			SCCG-DLB	214.0	19.99	0.41	97.69	58	43.82
	4	≥ 3092.0	CG	127.7	-	14.45	-	86	-
			SCG	169.1	-32.43	13.31	7.86	82	5.10
			CCG	93.6	26.65	0.76	94.73	44	48.72
			SCCG	102.8	19.52	0.07	99.52	40	53.36
			SCCG-DLB	82.5	35.39	0.14	99.05	33	61.95
	5	≥ 2780.3	CG	51.8	-	0.05	-	79	-
			SCG	71.6	-38.32	8.15	-16,688.35	68	13.71
			CCG	47.2	8.82	0.10	-113.87	32	59.14
			SCCG	49.0	5.41	0.03	37.15	27	65.23
			SCCG-DLB	49.9	3.59	0.04	23.21	26	67.26
25	2	3.0	CG	765.5	-	20.15	-	273	-
			SCG	571.6	25.33	8.73	56.67	271.6	0.51
			CCG	608.4	20.53	5.28	73.80	247.4	9.38
			SCCG	584.4	23.65	5.43	73.06	250.4	8.28
			SCCG-DLB	461.3	39.74	0.00	100.00	165	39.56
	3	124.2	CG	847.9	-	20.27	-	178	-
			SCG	663.0	21.80	14.93	26.35	143	19.62
			CCG	539.2	36.40	5.06	75.06	112	37.00
			SCCG	436.3	48.54	5.02	75.25	93	48.09
			SCCG-DLB	461.9	45.52	0.19	99.04	83	53.70
	4	≥ 934.1	CG	514.0	-	16.43	-	146	-
			SCG	537.2	-4.52	10.83	34.08	132	9.99
			CCG	286.5	44.25	3.46	78.95	72	50.48
			SCCG	303.6	40.94	3.41	79.22	63	56.63
			SCCG-DLB	292.1	43.16	0.08	99.52	53	63.61
	5	≥ 3077.3	CG	245.1	-	17.96	-	125	-
			SCG	257.4	-5.02	4.34	75.83	107	14.58
			CCG	138.3	43.58	0.23	98.72	49	60.90
			SCCG	155.7	36.49	0.20	98.88	44	64.90
			SCCG-DLB	128.6	47.55	0.15	99.19	38	69.87

typically within 30 to 460 CPU seconds, producing optimal or very near-optimal solutions.

Although the SCG algorithm achieved 8.2% LP iteration savings over the CG approach, it did not consistently improve the CPU time or the duality gap at termination, yielding relatively poor IP solutions. In contrast, the CCG feature resulted in an overall 23.1% CPU time savings at an average over the basic CG, with 72.5% reduction in % gap. Moreover, blending the SCG algorithm with the CCG feature resulted in 18.5% and 86.7% reduction in CPU time and in % gap, respectively.

It is noteworthy that utilizing the DLB feature in the accelerated column generation scheme achieved an overall CPU time savings of 22.4% over the base column generation algorithm. It also achieved an average 91.8% savings in the % gap, reducing it from an average of 12% gap with the basic CG approach to a 0.12% overall gap, with an accompanying 52.7% reduction in the number of LP iterations. That is, all the instances were solved to optimality or near-optimality with a faster convergence (evidenced by shorter CPU times, fewer LP iterations, and an earlier generation of useful columns). Figures 3.1a and 3.1b illustrate how the accelerated SCCG-DLB approach curtails the tailing-off effect, produces attractive columns in early iterations, and reduces the number of LP iterations in comparison with the basic CG approach for a representative instance.

It is, therefore, our conclusion that, for the multiple-runway aircraft sequencing problem, the stabilization feature tends to accelerate the convergence of the CG without improving the quality of IP solutions produced. The complementary column generation feature, in contrast, played a decisive role in both accelerating the algorithmic convergence and producing excellent IP solutions. However, blending the stabilization scheme with the complementary feature in concert with the additional dynamic lower bounding technique consistently led to a faster convergence to optimal or near-optimal solutions.
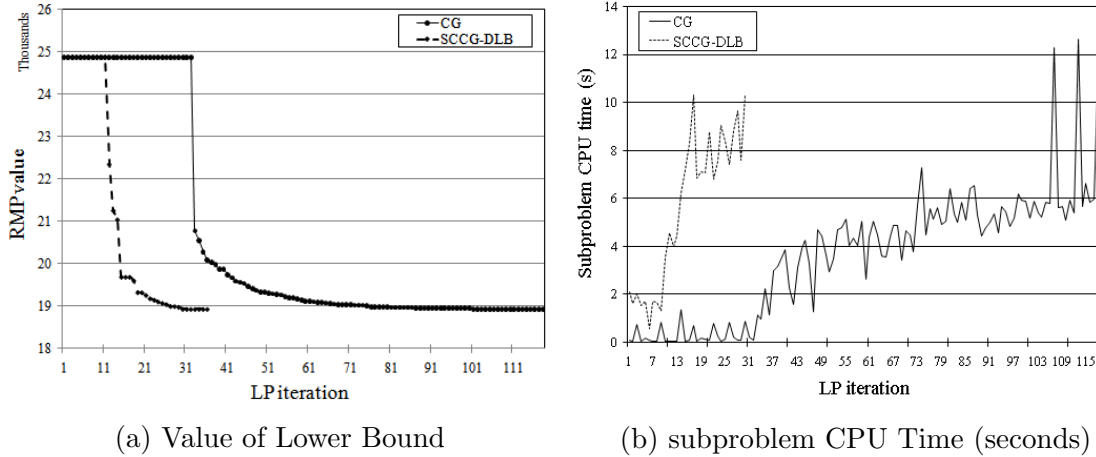
(a) Value of Lower Bound   (b) subproblem CPU Time (seconds)

Figure 3.1: CG acceleration for an instance of size $(n, m) = (25, 5)$

3.5. Conclusion and Directions for Future Research

This chapter examined the multiple-runway aircraft sequencing problem with both aircraft departures and arrivals. We investigated the computational tractability and the relative merits of a 0-1 mixed-integer program and a set partitioning formulation solved using column generation. Our empirical results indicate that the solution effort can be greatly reduced by adjoining valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints to the mixed-integer formulation. However, the aforementioned enhancements became less effective when the number of aircraft and runways increased, in which case the proposed column generation approach yielded attractive results. The individual and synergistic benefits of the acceleration schemes for the column generation approach were empirically investigated in our computational study. Our results suggest that the enhanced mixed-integer model failed to solve most of the instances with 20 and 25 aircraft with 4 and 5 runways within a time limit of 1 CPU hour, whereas the accelerated column generation approach consistently produced excellent solutions in manageable times.

From a computational point of view, we recommend for future research the investigation of tighter partial convex hull representations using the Reformulation-

Linearization Technique of Sherali and Adams (1990). This could be beneficial to further strengthen the mixed-integer formulation as well as the subproblem in the column generation approach. We also recommend solving the mixed-integer program that arises in the pricing subproblem using a dynammic programming procedure in lieu of resorting to solving it using a standard commercial solver such as CPLEX or GUROBI. This approach is detailed next in Chapter 4 and is demonstrated to yield very promising results. Other decomposition techniques, in the spirit of Benders decomposition or logic-based Benders decomposition, that decouple the assignment and sequencing decisions to tackle problem instances of even larger scale could be worthwhile. From an application point of view, we recommend for future research the development of integrated runway/taxiway models that extend the confines of classical models in the literature by taking into account such considerations as the location of runway exits, the network structure of taxiways, the physical constraints at holding areas, and the location of gates.

CHAPTER 4

AN EXACT DYNAMIC PROGRAMMING METHOD FOR SOLVING MULTIPLE-RUNWAY AIRCRAFT SEQUENCING PROBLEMS

In this chapter, the column generation approach proposed in Chapter 3 is further enhanced by devising a dynamic programming procedure for solving the pricing subproblem as an elementary shortest path problem with resource constraints. The results presented in Chapter 3 demonstrate the usefulness of employing complementary column generation and stabilization techniques to curtail the tailing off effect of column generation and to improve the ability to identify both tight lower and upper bounds on this class of problems. However, the computational effort involved in solving the column generation subproblem using commercial solvers was found to be relatively onerous and prohibits the implementation of a branch-and-price algorithm for multiple-runway aircraft sequencing problems. To overcome this difficulty, we develop a dynamic programming approach that is demonstrated to solve such pricing subproblems in manageable times, thereby greatly outperforming the use of commercial solvers to this end. Extending the work by Feillet et al. (2004), the proposed algorithm caters for non-triangular aircraft separation times and the need to separate consecutive as well as certain nonconsecutive aircraft. Our computational study was conducted using randomly generated computationally challenging instances that are often not solvable using CPLEX within a time limit of one CPU hour.

4.1. Introduction & Literature Review

The Multiple-Runway Aircraft Scheduling Problem (MRASP) has inherent similarities with m-Asymmetric Traveling Salesman Problems with Time-Windows or Vehicle Routing Problems with Time Windows (VRPTW). The VRPTW seeks to identify a set of minimum cost routes, originating and terminating at a depot, which service a set of costumers within the allowable service times, given a fleet of (capacitated) vehicles. In MRASP, runways can be metaphorically viewed as vehicles and aircraft as costumers. MRASP can be modeled as a 0-1 MIP (as in Beasley et al. 2000) or alternatively as a set partitioning problem (SPP). In the latter, a column represents of a subset of aircraft that are assigned to a runway with an associated objective cost that reflects their optimal sequencing over this runway.

The VRPTW has been widely investigated in the literature using exact as well as heuristic solution approaches. In a survey by Laporte (1992), exact approaches for the VRPTW are classified into three main categories: (i) tailored tree search methods; (ii) dynamic programming; and (iii) solving integer linear programs using optimization solvers. Solomon and Desrosiers (1988), in another extensive survey, categorize time-window-constrained routing and scheduling problems as follows: (i) single and multiple traveling salesman problems; (ii) shortest path problems; (iii) minimum spanning trees; (iv) VRPs; (v) pickup-delivery problems; and (vi) so-called shoreline problems. Optimization-based approaches for such problems mainly employ principles of implicit enumeration that are based on dynamic programing and branch-and-bound algorithms (Desrochers et al. 1988). Branch-and-bound approaches either rely on state space relaxations to compute lower bounds or solving the continuous relaxation of set partitioning models by column generation. In the latter case, so-called branch-and-price algorithms typically employ a dynamic programming procedure for single vehicle problems to solve the column generation pricing subproblem.

66

Single vehicle routing problems with time-windows generalize Shortest Path Problems with Time windows (SPPTW) and are commonly solved using dynamic programming procedures having pseudo-polynomial complexity (Feillet et al. 2004). A shortest path is determined via a recursive procedure, starting from the origin node and ending at the destination node, or vice versa. As the algorithm progresses each partial path is assigned a label that encapsulates pertinent information related to its cost, nodes it includes, and its resource consumption (e.g., time, capacity, etc.). What distinguishes a DP variant for SPPTW from another is the way the set of eligible nodes to be visited from a current node is formed and the order in which nodes are identified and selected for examination. Optimal shortest path algorithms are usually classified into *label correcting* algorithms and *label setting* algorithms (Fu et al. 2006).

This chapter makes the following contributions. First, it proposes a dynamic programming algorithm for solving elementary shortest path problems with non-triangular aircraft separation times. Second, it demonstrates the usefulness of the proposed DP approach over a set of computationally challenging problem instances and demonstrates that it greatly outperforms column generation approaches where the pricing problem is solved as a mixed-integer program. Third, it highlights algorithmic features that contribute to enhancing the computational performance of the proposed DP.

The remainder of this chapter is organized as follows. In Section 4.2 we present the set partitioning formulation used for solving the Multiple-Runway Aircraft Sequencing Problem. Section 4.3 presents the dynamic programing scheme for solving the column generation pricing subproblem. This section further proposes additional enhancements to a dynamic programing scheme to accelerate the computational speed. In Section 4.4 we demonstrate our computational results from solving the MRASP with a branch-and-bound method and comparing the results with our proposed dy-

namic programing algorithm with the enhancements. Section 4.5 summarizes our key findings and the directions for future study.

4.2. Set Partitioning Formulation of MRASP

In this section MRASP is presented as a set partitioning model with a convexity constraint. Consider a column construct, Q^h, which is associated with a runway. The column Q^h is an n-column vector having 0-1 entries, where $Q_j^h = 1$ equals 1 if and only if aircraft j is assigned to this column. Let κ_h be the total cost for Q^h, which reflects the cost in the optimized aircraft schedule associated with this particular runway, Q^h. Thus, aircraft assignment decisions are captured by the 0-1 entries of a column, whereas the accompanying aircraft sequencing decisions are reflected in the column cost itself. We use the following set partitioning problem, denoted by **SPP**. Consider the following notation for a formal statement of the master problem:

- Q^h: n-column with 0-1 entries that represents a subset of aircraft assigned to the same runway.

- κ_h: Cost associated with column Q^h. Here, it represents the total cost resulting from sequencing the aircraft included in column Q^h.

- x_h: Binary variable such that $x_h = 1$ if and only if column h is selected.

- π: Vector of dual variables associated with the set partitioning constraints in (4.1b), where $\pi = \bar{\pi}$ represents specific dual variable values obtained at a given iteration in the course of the column generation approach.

- π_0: Dual variable associated with constraint (4.1c), where $\pi_0 = \bar{\pi}_0$ represents a specific dual variable value obtained at a given iteration in the course of the column generation approach.

68

$$\textbf{SPP}\text{: Minimize } \sum_{h=1}^{H} \kappa_h x_h \tag{4.1a}$$

$$\text{subject to } \sum_{h=1}^{H} Q_j^h x_h = 1, \quad \forall j = 1, \ldots, n \tag{4.1b}$$

$$\sum_{h=1}^{H} x_h = m \tag{4.1c}$$

$$x \text{ binary.} \tag{4.1d}$$

The objective function (4.1a) minimizes the total schedule cost (the total weighted start times). Constraint (4.1b) achieves a set partitioning scheme for aircraft, guaranteeing that every aircraft is assigned to exactly one runway. Constraint (4.1c) is a convexity constraint that enforces the number of available runways.

To circumvent a complete enumeration of a possibly exponential number of columns, each corresponding to a subset of aircraft that can be feasibly assigned to a runway, column generation approaches can be used to heuristically solve Model SPP or can be embedded in a branch-and-price algorithm to solve it to optimality. Our computational experience in Chapter 3, however, indicates that solving the pricing subproblem as an MIP can be computationally onerous and can inhibit the development of branch-and-price algorithm. To overcome this shortcoming, we propose next a label correcting dynamic programing scheme to solve the pricing subproblem while taking account non-triangular aircraft separation times.

4.3. Solving the Subproblem via Dynamic Programming

In this section, we present the proposed DP approach for solving the column generation pricing subproblem of model SPP. Let V denote the set of landing/departing aircraft under scrutiny. In the sequel, each aircraft is referred to as a node and every pair of nodes is connected via a directed arc whose cost, $\delta(u, v), \forall u, v \in V, u \neq v$,

equals the minimum separation time between the corresponding two aircraft. As discussed in Subsection 2.4.1, these separation times are non-triangular under the FAA standard. Further, each node is characterized by a time-window during which the start-time of of its aircraft, t_v, $v \in V$, has to be scheduled. That is, $r_v \leq t_v \leq d_v$, $\forall v \in V$, where r_v and d_v are the ready time and the deadline for node v, respectively. We also define weight w_v associated with each node and consider cost of visiting a node as $w_v t_v$.

Label correcting algorithms extend the paths to possible adjacent nodes by either reaching forward or backtracking, and labels are updated accordingly. Dominance rules are commonly employed eliminate inferior paths in order to reduce the number of extended paths that have to be considered as the algorithm progresses. Desrochers (1988) introduced a label reaching algorithm based on the Bellman-Ford-Moore shortest path algorithm and demonstrated its usefulness for VRPTW (Desrochers et al. 1992). Feillet et al. (2004) extended the algorithm for the elementary shortest path problems with resource constraints (ESPPRC). In the elementary shortest path problem, each node appears on a path at most once. In such algorithms, nodes and the labels are iteratively examined until no new label is available. It should be noted that such algorithms are based on the assumption that triangular inequality holds for arc costs. Considering a network $G \equiv (V, E)$ with arc costs $\delta : E \rightarrow R$ for any edge $(u, v) \in E$, triangle inequality dictates that $\delta(s, v) \leq \delta(s, u) + \delta(u, v)$, for any triplet of nodes in the network. However, this condition does not hold in MRASP due to non-triangular aircraft separation times.

In Subsection 4.3.1, we first introduce a path extension scheme. This is followed by a discussion of dominance rules in Subsection 4.3.2. The overall DP scheme is summarized in Subsection 4.3.3, whereas Subsection 4.3.4 discusses a base and an enhanced DP implementation variants.

4.3.1 Path Extension Scheme

Let path P_v be the path extended from dummy origin s to node v. We set the ready-time of the dummy node s as $r_s = 0$, its arc cost to any other node in the network as $\delta(s, v) = 0, \forall v \in V$, and its dual variable value to $\pi_s = \bar{\pi}_0$ (i.e. the dual value associated with Constraint (4.1c)). Any path P_v is characterized by a cost c_v and its resource consumption t_v which, in the context of MRASP, corresponds to the cumulative time along the path (i.e., the earliest time at aircraft/node v can start). This is information is encapsulated in $P_v(c_v, t_v, R_v)$, where R_v denotes the set of reachable nodes from v. With this notation, for the case of the dummy node $P_s = (-\bar{\pi}_0, 0, V)$. When node u is examined, all non-dominated paths from u will be extended to all adjacent reachable nodes, v, and the labels of the v nodes will be updated accordingly.

Algorithm 2: EXTEND(P_u, v)

1: $t_v = \max \left\{ r_v, t_{\tau(P_u, l)} + \delta(\tau(P_u, l), v) : l = 0, \ldots, min\{\theta, |P_u|\} \right\}$
2: $c_v = c_u + w_v t_v - \bar{\pi}_v$
3: UPDATE-R(P_v)
4: $P_v \leftarrow (c_v, t_v, R_v)$

With the use of dominance rule the labels are eliminated at destinations. The dominance rule in Feillet et al. (2004) works as follows. Let $P_v^*(c_v^*, t_v^*, R_v^*)$ and $P_v'(c_v', t_v', R_v')$ be two distinct paths extended to v. P_v^* dominates P_v' if and only if $c_v^* \leq c_v', t_v^* \leq t_v'$, and $R_v' \subseteq R_v^*$. The optimal solution will be reached by extending the non-dominated paths to reachable nodes.

To determine an optimal solution with non-triangular arc costs, the attributes of the labels, the label updating rule, and the dominance rule need to be adjusted. We define the *degree of non-triangularity* for any node u in the graph, θ, as the maximum number of predecessors along a path extended to u for which nonconsecutive separation needs to be enforced. In general, θ can vary between 0 and $|V| - 1$. The case of $\theta = 0$ presents itself in a network where the triangular inequality holds. The

worst case of $\theta = |V| - 1$ arises when there exists a path through all nodes in V, ending in u, such that the consecutive separation between the $|V| - 1$ consecutive nodes preceding u is not sufficient to properly separate the first node and node u. An examination of the FAA aircraft separation time standard in Table 2.2 indicates that $\theta = 3$. For example, this arises in the following sequence of four aircraft: Heavy arrival, Large departure, Large departure, and Small arrival, where the consecutive separation times amount to $75 + 60 + 60 = 195$ s, whereas 196 s are required between the first and fourth aircraft.

Algorithm EXTEND(P_u, v) updates the labels as we extend the paths from node u to node v. Attribute $\tau(P_u, l)$ records the lth parent of node u in the path P_u with $\tau(P_u, 0) \equiv u$. Since the separation times are non-triangular, we need to enforce separation times between consecutive as well as non-consecutive nodes. While updating the time t_v we take the maximum time that will appropriately separate v from its $1, \ldots, \theta^{th}$ predecessors. UPDATE-R(P_v) identifies the adjacent nodes $j \notin P_v$ that are feasible to extend the label, $t_j \leq d_j$ (i.e. extending the label is not causing any time window violation) and $w_j r_j - \bar{\pi}_j < 0$ so that they can improve the path P_v. Let $\bar{\mathcal{P}}_v$ be the set of non-dominated labels already extended to node v.

4.3.2 Dominance Rules

Algorithm DOMINANCE(P_v^*, P_v', λ) checks for domination of labels when new labels are extended to each node. A few modifications are required to Feillet et al. (2004) dominance rule. Ω is a temporary set storing pair of labels at each node. ENQUEUE and DEQUEUE add and remove the pair of labels from the set Ω. If path P_v^* is not dominating path P_v' the algorithm will return FALSE, thus the path P_v' will be added to the labels at node v. Otherwise we need to check the dominance of P_v^* if both paths extend to the possible adjacent reachable nodes for the next θ stages. If all of the possible extensions for the next θ stages were dominant then we conclude that

path P'_v is dominated by P^*_v and therefore will not be extended to node v. Let λ be the degree of non-triangularity of the network that we defined earlier as θ.

Algorithm 3: DOMINANCE (P^*_v, P'_v, λ)

1: **if** $(c'_v < c^*_v$ or $t'_v < t^*_v$ or $R'_v \nsubseteq R^*_v)$ **then**
2:     return FALSE
3: ENQUEUE $(\Omega, (P^*_v, P'_v))$
4: **while** $\lambda > 0$ and $\Omega \neq \emptyset$ **do**
5:     DEQUEUE(Ω)
6:     **for** $j \in R^*_v \cap R'_v$ **do**
7:         EXTEND(P^*_v, j)
8:         EXTEND(P'_v, j)
9:         DOMINANCE $(P^*_j, P'_j, \lambda - 1)$
10: return TRUE

The DOMINANCE(P^*_v, P'_v, λ) algorithm will solve the shortest path problem for the networks with non-triangular distances in the general case. However, the complexity of the algorithm *exponentially* increases as the degree of non-triangularity of the network increases. For the case that $\theta = |V| - 1$ the algorithm will eventually is required to search a complete enumeration of the nodes at each iteration of label extension.

To resolve this issue we avail of the fact that the FAA separation times have a standardized format. We extract the exceptions that the triangle inequality are violated. Defining the standard exceptions, we redefine the DOMINANCE algorithm according to the exceptions. With regard to the separation times there will be three main exceptions that the separations violate the triangle inequality. These exceptions are depicted in Figures 4.1 and 4.2.

We explain EXCEPTION1 with an example depicted in Figure 4.3. Here, $t^*_v < t'_v$ and assume that $c^*_v < c'_v$, and $R'_v \nsubseteq R^*_v$. If we assume that the triangle inequality holds, according to the dominance rule proposed by Feillet et al. (2004) we can conclude that P^*_v dominates P'_v and therefore P'_v is not extended to v. However for a reachable node j, under FAA separation times we will have $t^*_j = max\{r_j, t_v + \delta(v, j), t_{v_1} + \delta(v_1, j), t_{v_2} +$

73

(a) EXCEPTION1   (b) EXCEPTION2
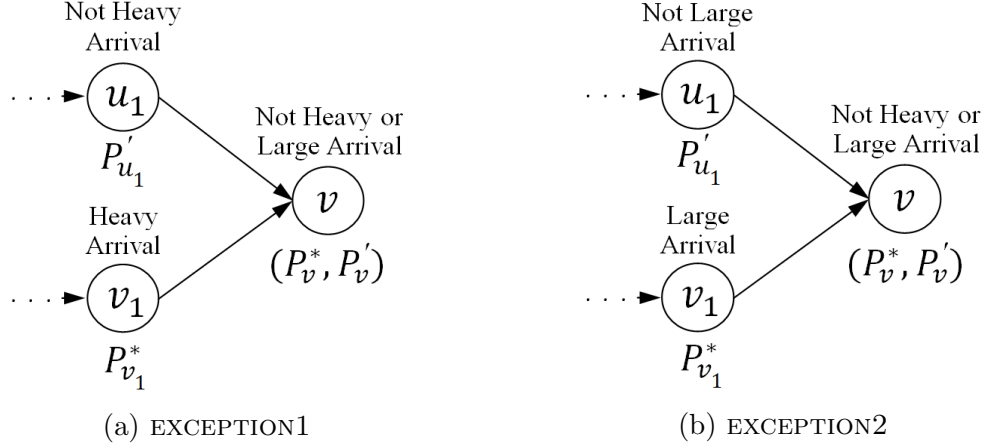
Figure 4.1: Exceptions 1 and 2 in dominance rule
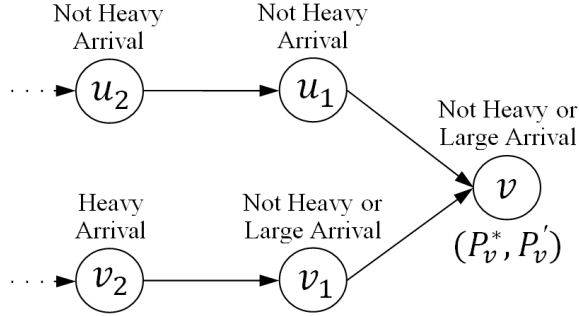
Figure 4.2: EXCEPTION3 in dominance rule

$\delta(v_2, j)\} = 996$ while $t'_j = max\{r_j, t_v + \delta(v, j), t_{u_1} + \delta(u_1, j), t_{u_2} + \delta(u_2, j)\} = 960$. Due to the non-triangular separation times and the fact that all of the separation times (consecutive and non-consecutive) are required to be conserved, we encounter a contradiction where $t'_v < t^*_v$ and therefore, P^*_j will no longer dominate P'_v at j. Node v_1 causes a 61 unit increase in t^*_j. We will refer to this increase as PUSH.

Let v° be the node in P^*_v that can cause the PUSH in t^*_j. For EXCEPTION1 and EXCEPTION2 this node is v_1 and in EXCEPTION3 this node is v_2 (see Figures 4.1 and 4.2). For node j we assume that it will be a node that could potentially cause the largest PUSH in a general case. The worst case PUSH occurs when node j is a Small Arrival and we have $\delta(v, j) = 60$. The value of PUSH can be derived by

Figure 4.3: Example from EXCEPTION1

$\delta(v^\circ, j) - ((t_v^* + 60) - t_{v^\circ}^*)$. If PUSH $\leq (t_v' - t_v^*)$ then we have $t_v^* < t_v'$ and therefore P_v^* dominates P_v'. Otherwise, no conclusion can be made and the label P_v' will be added to $\bar{\mathcal{P}}_v$. If label P_v^* dominates P_v' in the worst case scenario it will dominate in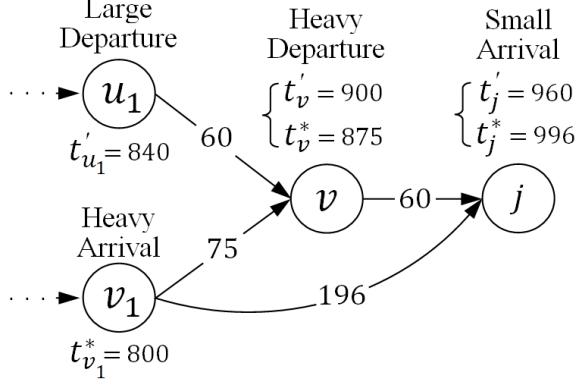 other cases. Algorithm CHECK-DOMINANCE will determine our adjusted dominance rule among pairs of labels using the exception rules.

Using the CHECK-DOMINANCE algorithm we update the set of non-dominated labels $\bar{\mathcal{P}}_v$ at node v. Let the set L_v be the set of newly extended labels to node v at each iteration. UPDATE-$\bar{\mathcal{P}}(\bar{\mathcal{P}}_v, L_v)$ adds L_v to $\bar{\mathcal{P}}_v$ while removing all non-dominated labels. For each newly generated path P_v' from L_v and each path P_v^* from $\bar{\mathcal{P}}_v$, the algorithm will remove the non-dominated labels from both sets. At the end, all the non-dominated members of L_v and $\bar{\mathcal{P}}_v$ will constitute the updated set $\bar{\mathcal{P}}_v$.

Algorithm 4: CHECK-DOMINANCE(P_v^*, P_v')

1: **if** $(c_v^* \leq c_v'$ and $t_v^* \leq t_v'$ and $R_v' \subseteq R_v^*)$ **then**
2:    **if** any of the exceptions apply **then**
3:       identify the exception node v°
4:       $j \leftarrow$ Small Arrival
5:       PUSH $= \delta(v^\circ, j) - ((t_v^* + 60) - t_{v^\circ}^*)$
6:       **if** PUSH $> (t_v' - t_v^*)$ **then**
7:          return False
8: return True

Algorithm 5: UPDATE-$\bar{\mathcal{P}}(\bar{\mathcal{P}}_v, L_v)$

1: **for** $P_v' \in L_v$ **do**
2:     **for** $P_v^* \in \bar{P}_v$ **do**
3:        **if** CHECK-DOMINANCE(P_v^*, P_v') **then**
4:           $L_v = L_v \setminus \{P_v'\}$
5:           Break
6:        **if** CHECK-DOMINANCE(P_v', P_v^*) **then**
7:           $\bar{\mathcal{P}}_v = \bar{\mathcal{P}}_v \setminus \{P_v^*\}$
8: $\bar{\mathcal{P}}_v = \bar{\mathcal{P}}_v \cup L_v$

4.3.3   Overall DP Procedure

To demonstrate the algorithm for non-triangular shortest path problem with time windows (NTSPPTW) we use the following sets. Let set A be the set of nodes with negative reduced cost. Set E is the set of nodes that have unextended labels and need treatment. We define NTSPPTW as follows.

Algorithm 6: NTSPPTW

1: **for** $v = 1$ to n **do**
2:     $P_v = $ EXTEND(P_s, v)
3:     **if** $r_v w_v - \bar{\pi}_v < 0$ **then**
4:        Add v to A and E
5:        Add P_v to $\bar{P}_v$
6: **while** $E \neq \emptyset$ **do**
7:     Select a node u from E and $E = E \setminus \{u\}$
8:     **for all** $v \in A$ **do**
9:        **for all** $P_u \in \bar{P}_u$ **do**
10:           **if** $v \in R_u$ **then**
11:              $P_v = $ EXTEND(P_u, v)
12:              $L_v = L_v \cup \{P_v\}$
13:          UPDATE-$\bar{\mathcal{P}}(\bar{\mathcal{P}}_v, L_v)$
14:        **if** at least one path was added to $\bar{\mathcal{P}}_v$ **then**
15:           $E = E \cup \{v\}$

4.3.4   Enhanced vs. Base DP

In our computational study, we contrast two implementations of the proposed DP approach. The first, simply referred to as DP, is delineated in Algorithm 6: NTSPPWT. The second is an enhanced DP (EDP) procedure with the following details that aim at accelerating its computational performance:

1. *Multiple columns*: At each iteration of DP for solving the subproblem, DP produces many labels along with the optimal label with the minimum reduced cost. Here, instead of adjoining the best column to the RMP, EDP augments the RMP with a set of up to k columns.

2. *CPU time threshold*: At the early stages of DP, RMP is not producing good quality dual values and normally the labels that are produced by DP are not good quality columns for RMP in return. Therefore we set a time limit ϵ for the computation time at each iteration of DP. If at the time ϵ, DP has labels with negative reduced costs it terminates and returns the labels, otherwise it will continue until termination by finding the optimal label.

3. *Dominance rule*: The dominance rule in Algorithm 4 plays an important role in eliminating the inferior labels and therefore reducing the size of the problem and accelerating the computational speed. We can use a heuristic version of the dominance rule before reaching the termination in column generation. To this end we follow this procedure:
   - Relax the condition $R'_v \subseteq R^*_v$ in Algorithm 4;
   - If DP did not find labels with negative reduced cost at termination, then fix the condition and re-run DP;
   - If no label with negative reduced cost is produced, then terminate the column generation;
   - Otherwise, for next iteration run the relaxed DP.

4.4. Computational Results

In this section we discuss the computational results on Mixed-Integer Programing (MIP) approach where MRASP is solved by branch-and-bound/cut approach with the solver CPLEX 12.4 and the column generation approach where the subproblem is solved by our proposed DP and EDP. For the MIP approach we used a version of the model MRASP introduced in Chapter 3 where the constraint (3.1c) is relaxed. The instances used for these series of computations are the same test-bed introduced in Chapter 3, Subsection 3.4.1 including 55 instances. The minimum separation times used as $\delta(u, v)$ for our computations are the FAA separation standard introduced in Subsection 2.4.1, Table 2.2. The column generation approach, DP, and enhancements of DP were coded with C++. The MIP approach was tested using CPLEX 12.4. All the computations were conducted on a Windows 7 professional 64-bit operating system with an Intel Core i7-2600 CPU with 3.40 GHz and 12 GB RAM desktop. A time limit of 3600 CPU seconds was imposed on all runs.

Table 4.1 reports our computational results for the MIP approach, DP, and EDP. The first two columns specify the number of aircraft, n, and the number of runways, m. For each problem size there are 5 randomly generated instances. We report the average values for each problem size. Column 3 specifies the method used to solve the instances. The following method variants considered in our computational study:

- MIP refers to a variation of the base MIP model introduced in Chapter 3, Subsection 3.2.1 where the Constraint (3.1c) is relaxed. Equations: (3.1a, 3.1b, 3.1d, 3.1e, 3.1f, and 3.1g). This MIP model is solved by a branch-and-bound/cut method with the solver CPLEX 12.4.

- DP refers to the Algorithm 6: NTSPPWT. Column generation terminates when DP was unable to produce any label with negative reduced cost.

- EDP refers to the DP algorithm with additional enhancements.

We summarize the computational results for MIP, DP, and EDP in Table 4.1. Column 4 reports the number of solved instances within the 1 hour time threshold. Column 5 reports the average CPU time in seconds. Column 6 demonstrates the average percentage reduction in CPU time comparing DP and EDP. Column 7 reports the average percentage gap after termination. The results for DP and EDP are reported for the root node. Column 8 summarizes the average percentage reduction in the percentage gap comparing DP and EDP.

As it was tested in Chapter 3, Subsection 3.4.1 the MIP generally fails to efficiently solve MRASP. Even with the enhancements and the additional convexity Constraint (3.1c) the results in Table 3.1 showed that a MIP approach even with enhancements is unable to solve the problem instances to optimality within the time limit. Here in Table 4.1 we also test MRASP in isolation without the convexity constraint. Besides the problem size (m, n)=(25,2) the MIP failed to solve all of the instances in the specific problem sizes to optimality. The percentage gaps that are reported by the solver at termination are also reported in the table. for instance, for the problem size (20,5) the average percentage gap is at least 24.2% since the solver was teminated at 1 hour time limit.

Under the proposed DP algorithm all of the instances for different problem sizes were solved within the time limit until the column generation terminated with the true lower bound. The CPU times are substantially lower than MIP approach. For the case of (15,2) the average CPU time is 36.4 seconds compare to the 1897.3 seconds for MIP with just 3 instances solved to optimality. Two observations are worth mentioning here. First, the DP has shown very good reductions in average CPU times but the percentage gaps at termination are higher than the gaps achieved by MIP. For example for the same problem size (15,2) the percentage gap for DP is 39.5% compare to MIP with 4.9%. Although DP is effectively finding the lower bounds at very reasonable times compare to MIP, at termination the column generation faces

Table 4.1: Results on the Dynamic Programing Approach

n	m	method	Solved	CPU time (s)		% red. time	% gap		% red. gap	# of columns
15	2	MIP	3/5	$\geq$	1897.3		$\geq$	4.9		
		DP	5/5		36.4			29.5		84
		EDP	5/5		0.6	98.5		0.7	97.6	747
	3	MIP	1/5	$\geq$	2941.6		$\geq$	11.6		
		DP	5/5		1.2			26.1		62
		EDP	5/5		0.2	82.4		0.1	99.5	448
	4	MIP	0/5	$\geq$	3600.0		$\geq$	16.6		
		DP	5/5		39.7			38.6		70
		EDP	5/5		0.1	99.7		0.0	100.0	391
20	2	MIP	4/5	$\geq$	1065.3		$\geq$	1.7		
		DP	5/5		597.9			3.7		161
		EDP	5/5		6.0	99.0		0.1	97.7	1261
	3	MIP	0/5	$\geq$	3600.0		$\geq$	7.9		
		DP	5/5		386.0			15.3		113
		EDP	5/5		0.6	99.9		0.0	99.7	743
	4	MIP	0/5	$\geq$	3600.0		$\geq$	16.6		
		DP	5/5		17.2			13.1		96
		EDP	5/5		0.3	98.0		0.2	98.8	600
	5	MIP	0/5	$\geq$	3600.0		$\geq$	24.2		
		DP	5/5		6.6			62.1		105
		EDP	5/5		0.3	95.6		0.0	100.0	632
25	2	MIP	5/5		29.2			0.0		
		DP	5/5		413.0			9.0		242
		EDP	5/5		1.8	99.6		8.7	3.5	1927
	3	MIP	3/5	$\geq$	2146.6		$\geq$	1.8		
		DP	5/5		36.3			25.1		189
		EDP	5/5		1.1	96.9		3.7	85.1	1337
	4	MIP	0/5	$\geq$	3600.0		$\geq$	3.8		
		DP	5/5		76.0			20.2		174
		EDP	5/5		0.9	98.8		0.0	99.8	1107
	5	MIP	1/5	$\geq$	2913.0		$\geq$	7.4		
		DP	5/5		49.5			34.8		153
		EDP	5/5		0.7	98.7		0.0	99.9	870

difficulties to find good quality integer solutions and therefore resulting in large gaps at termination. The second observation is that DP tends to perform faster at problem sizes with larger number of runways. For the case of (20,2) the algorithm terminates at 597.9 seconds on average. As the number of runways increases we see for the problem size (20,5) DP solves all of teh instances at an average CPU time of 6.6 seconds.

In contrast, adding the enhancements to DP shows substantial improvements in reducing CPU time and duality gap. Adding the enhancements, EDP shows an overall 97% reduction in the CPU time and an overall reduction of the duality gap by 89%. All of the problem instances were solved in a fraction of seconds. The worst case performance of EDP was on problem size (20,2) with an average CPU time of only 6 seconds and the average duality gap of 0.1%. Out of 55 instances, 42 instances were solved to optimality at the root node. RMP was unable to find an optimal integer solution for only 13 instances at termination of column generation approach. The enhancements to DP resulted in producing significantly larger number of labels compare to the base DP. EDP on average was able to provide RMP with 6 times more number of labels relative to DP. This increase in number of the labels reduced the total number of column generation iterations and therefore the total computational time. It also provided the RMP with many more number of columns and led to producing good quality integer solutions that would result in optimal or very near-optimal solutions.

It can be concluded that Multiple-Runway Aircraft Sequencing Problems can be efficiently solved by the dynamic programing method. The base dynamic programing approach enables the column generation to converge at a very fast rate but the quality of the integer solutions were not appealing. Adding the enhancements to the basic DP can substantially reduce the computation time and the duality gap. The combination of enhancements on DP, show that there is no need for producing opti-

mal labels at early stages of column generation and only returning a batch of labels with negative reduced cost can potentially provide the restricted master problem with sufficient information to produce good quality dual values. This is also due to the fact that the enhancements would enable the dynamic programing algorithm to produce significantly larger number of the columns and therefore reducing the column generation convergence time and increasing the quality of the integer solution.

4.5. Conclusion and Remarks

This chapter, proposed a methodology to efficiently solve the Multiple-Runway Aircraft Sequencing Problem. The algorithm is based on a dynamic programing approach to solve the shortest path problem within the column generation scheme. The main difficulty of the Multiple-Runway Aircraft Sequencing Problem is that minimum safety distances between aircraft operations must be preserved within the consecutive and non-consecutive operations considering the fact that these distances do not hold the triangle inequality. The proposed dynamic programing method finds an exact solution to MRASP while considering non-triangular distances and other side constraints. This chapter further introduced enhancements to the base dynamic programing approach to accelerate the computational speed.

Our empirical results reveal that Multiple-Runway Aircraft Sequencing Problems can be efficiently solved by dynamic programing method. Branch-and-bound/cut methods failed to solve different problem sized to optimality. In contrast the proposed dynamic programing algorithm solved all of the instances in our test-bed within a reasonable time. The only drawback for the base dynamic programing approach was the poor duality gap at convergence. This issue was addressed by the introduction of enhancements to the base dynamic programing algorithm. The enhancements enabled the algorithm to produce significantly larger number of columns in less time and therefore reducing the total number of the iterations, reducing the total amount

of computational time and substantially improving the quality of the integer solutions at the convergence. In fact, the enhanced dynamic programing approach was able to solve most of the instances to optimality at the root node. Although the proposed algorithm converges with optimal or very near-optimal solutions, we recommend for future research the implementation of a branch-and-price approach to solve this class of problems to optimality.

CHAPTER 5

CONCLUSION AND DIRECTIONS FOR FUTURE WORK

This dissertation presents three essays on runway operations management. The first essay proposes a three-faceted approach that examines the impact of runway physical configurations, runway scheduling strategies, and runway safety regulations on runway capacity utilization using optimization methodology. This work is grounded in the analysis of data on Doha International Airport which operates a single runway and will be replaced, during 2013, by two parallel, independent runways in the New Doha International Airport. The second essay proposes an in-depth study of multiple runway aircraft sequencing problems using mixed-integer programming models and column generation algorithms. The third essay designs and proposes a dynamic programing scheme to efficiently solve Multiple-Runway Aircraft Sequencing Problem. The proposed dynamic programing method finds an exact solution to the problem while considering the specific problem attributes and other side constraints. This chapter summarizes our findings and identifies directions for future research.

5.1. Summary, Findings and Insights

Essay one examines a three-faceted approach for runway capacity management, based on the runway configuration, a chosen sequencing policy, and an aircraft separation standard. In this context, we propose optimization-based heuristics that yield optimal or near-optimal schedules and assess their benefits under alternative runway settings. This integrated approach is applied, in collaboration with *Qatar Civil Aviation Authority*, to investigating the transition from the (Old) Doha International

Airport to the New Doha International Airport. Our computational study of alternative runway settings uses optimization methodology along with tailored preprocessing routines. The main findings on this essay can be summarizes as the the following:

- Using data from Qatar Civil Aviation Authority, this essay demonstrates that the transition from a single runway with a nominal capacity of 30 arrivals per hour, as in the Doha International Airport, to two parallel independent runways with the nominal capacity of 60 arrivals per hour, as planned in the new Hamad International Airport (HIA), would achieve nearly \$3 million savings per day in excess fuel burn cost.

- The excess fuel cost or delays would not be completely eliminated, even under a two-runway configurations. This further highlights the necessity of examining enhanced sequencing policies and alternative aircraft separation times in order to better exploit the runway capacity during busier hours.

- The study introduced a mixed-integer mathematical programing model for Multiple-Runway Aircraft Sequencing Problem and developed an optimization-based heuristic which is based on the FCFS sequencing policy. We find that by slightly altering the FCFS sequence, the proposed heuristic not only preserves fairness among aircraft, but also consistently produces excellent (optimal or near optimal) solutions. Without deviating aircraft by not more than 2 positions from their FCFS sequence positions, the objective value produced by the proposed heuristic deviated by less than 1% from the optimal objective value found using a mixed-integer program.

- Our empirical results also indicate that international airports such as the Hamad International Airport can significantly benefit from using the FAA aircraft separation standard in lieu of the ICAO standard. In the specific case of HIA, this choice is expected to achieve nearly a 50% reduction in excess fuel cost.

Essay two examines aircraft sequencing problems over multiple runways under mixed mode operations. Crafting valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints yields computational savings over a base mixed-integer formulation using a branch-and-bound/cut technique. To further enhance its computational tractability, the problem is alternatively reformulated as a set partitioning model with one convexity constraint that prompts the development of a specialized column generation approach. The latter is accelerated by incorporating several algorithmic features, including an interior point dual stabilization scheme, a complementary column generation routine, and a dynamic lower bounding feature. Empirical results using a set of computationally challenging simulated instances demonstrate the effectiveness and the relative merits of the strengthened mixed-integer formulation and the accelerated column generation approach. The main findings on this essay can be summarizes as the the following:

- We investigated the computational tractability and the relative merits of a 0-1 mixed-integer program and a set partitioning formulation solved using column generation.

- Our empirical results indicate that the solution effort can be greatly reduced by adjoining valid inequalities, preprocessing routines, and symmetry-defeating hierarchical constraints to the mixed-integer formulation. However, the aforementioned enhancements became less effective when the number of aircraft and runways increased. Our results suggest that the enhanced mixed-integer model failed to solve most of the instances with 20 and 25 aircraft with 4 and 5 runways within a time limit of 1 CPU hour.

- The proposed column generation approach yielded attractive results in cases where the branch-and-bound/cut failed to efficiently solve the 0-1 mixed-integer program to optimality within a reasonable time. Additional acceleration schemes

introduced to the base column generation approach to further enhance the computational efficacy. An interior point dual stabilization algorithm, a complementary feature and a dynamic lower bounding technique introduced to the base column generation.

- The individual and synergistic benefits of the acceleration schemes for the column generation approach were empirically investigated in our computational study. The accelerated column generation approach consistently produced excellent solutions in manageable times.

- For the multiple-runway aircraft sequencing problem, the stabilization feature tends to accelerate the convergence of the column generation without improving the quality of IP solutions produced. The complementary column generation feature, in contrast, played a decisive role in both accelerating the algorithmic convergence and producing excellent IP solutions. However, blending the stabilization scheme with the complementary feature in concert with the additional dynamic lower bounding technique consistently led to a faster convergence to optimal or near-optimal solutions.

Essay three, proposed a methodology to efficiently solve the Multiple-Runway Aircraft Sequencing Problem. The algorithm is based on a dynamic programing approach to solve the shortest path problem within the column generation scheme. The main difficulty of the Multiple-Runway Aircraft Sequencing Problem is that minimum safety distances between aircraft operations must be preserved within the consecutive and non-consecutive operations considering the fact that these distances do not hold the triangle inequality. The proposed dynamic programing method finds an exact solution to MRASP while considering non-triangular distances and other side constraints. This essay further introduced enhancements to the base dynamic

87

programing approach to accelerate the computational speed. The main findings on this essay can be summarizes as the the following:

- Our empirical results reveal that Multiple-Runway Aircraft Sequencing Problems can be efficiently solved by dynamic programing method. Branch-and-bound method failed to solve different problem sized to optimality as it was also tested and shown in essay two.

- The proposed dynamic programing algorithm solved all of the instances in our test-bed within a reasonable time. The base dynamic programing approach enables the column generation to converge at a very fast rate. The only drawback for the base dynamic programing approach was the poor duality gap at convergence.

- We further introduced enhancements to the base dynamic programing algorithm. The enhancements enabled the algorithm to produce significantly larger number of columns in less time and therefore reducing the total number of the iterations, reducing the total amount of computational time and substantially improving the quality of the integer solutions at the convergence. In fact, the enhanced dynamic programing approach was able to solve most of the instances to optimality at the root node.

- The enhancements enabled the dynamic programing algorithm to produce significantly larger number of the columns and therefore reducing the column generation convergence time and increasing the quality of the integer solution.

- The combination of enhancements on DP, show that there is no need for producing optimal labels at early stages of column generation and only returning a batch of labels with negative reduced cost can potentially provide the restricted master problem with sufficient information to produce good quality dual values.

5.2. Future Research Directions and Recommendations

Although illustrated with real data for Doha International Airport, the approach presented in the essay one and the proposed heuristic can be of general benefit to other airports, especially during busier hours of activity during the day. The anticipated savings in fuel costs can directly benefit airlines, airports, and governmental authorities that are concerned with environmental effects and emissions. We recommend for further investigation an analysis of the impact of alternative runway settings on additional airborne or ground-based operations related to taxiway routing, gate assignments, and workload at terminals.

From a computational point of view, we recommend for future research the investigation of tighter partial convex hull representations using the Reformulation-Linearization Technique of Sherali and Adams (1990). This could be beneficial to further strengthen the mixed-integer formulation as well as the subproblem in the column generation approach. Other decomposition techniques, in the spirit of Benders decomposition or logic-based Benders decomposition, that decouple the assignment and sequencing decisions to tackle problem instances of even larger scale could be worthwhile. From an application point of view, we recommend for future research the development of integrated runway/taxiway models that extend the confines of classical models in the literature by taking into account such considerations as the location of runway exits, the network structure of taxiways, the physical constraints at holding areas, and the location of gates.

On the last essay, we recommend for future research the necessity of implementing a branch-and-price approach to find the final optimal solution. It is also necessary to test the proposed algorithm on larger problem sizes with larger number of aircraft and runways. From an application point of view we recommend to test the algorithm on the instances generated based on real life examples such as the data shown in Chapter 2.

APPENDIX A

FUEL CONSUMPTION ESTIMATES

Fuel costs are calculated based on fuel burn/minute for an aircraft, which depends on the aircraft operation and its weight class. We employed the base fuel burn of the aircraft models categorized by Cook et al. (2004) and used estimates for average fuel burn (gallons per block hour of operation) for the existing aircraft models operating in DOH. The following table reports the average estimates of fuel burn (gal/hour) for aircraft weight categories based on the operation type at DOH.

	Heavy	Large	Small
Arrival	5,043	2,063	206
Departure	1,614	658	66

Table A.1: Fuel Burn Consumption (gal/hour)

APPENDIX B

TRADE-OFFS BETWEEN TOTAL DELAYS AND FUEL CONSUMPTION

In Chapter 2 we find the optimal solutions with respect to fuel cost function for ICAO and FAA separation standards. Optimizing fuel costs will cause sub-optimal solutions in terms of total delays. Here we solve the instances to optimality with the objective of minimizing total delays and record the value of fuel cost for the optimal solution with respect to total delays. Table B.1 and B.2 summarize the results. Columns 2 and three in each table shows the values of the respective function while optimizing the selected objective function.

Objective function	Resulted value			
	Delay (min)		Fuel cost (USD)	
Delay (min)	475		50,071	(54.3% increase)
Fuel cost (USD)	511	(7.6% increase)	32,440	

Table B.1: Trade-offs with ICAO standard

Objective function	Resulted value			
	Delay (min)		Fuel cost (USD)	
Delay (min)	171		25,100	(45.8% increase)
Fuel cost (USD)	209	(22.1% increase)	17,211	

Table B.2: Trade-offs with FAA standard

For example, in Table B.1 optimizing total delays will result to 475 minutes of delay while the fuel cost for this solution is 50,071 USD. Optimizing delays caused 54.3% increase in the fuel cost compare to optimal fuel costs. On the other hand,

optimal fuel costs are 32,440 USD which will enforce just 7.6% increase in the delays compare to optimal delays. This shows that optimal fuel cost results in solutions that are very close to optimal delays. In Table B.2 we summarize similar results under FAA standard. With FAA separations, due to more refined separation times, the optimal solutions result in smaller values in general. However, the optimal fuel costs have much more impact on delays (22.1% increase compare to optimal delays). Figure B.1 depicts the relative trade-offs between the two objective functions under ICAO and FAA standards.
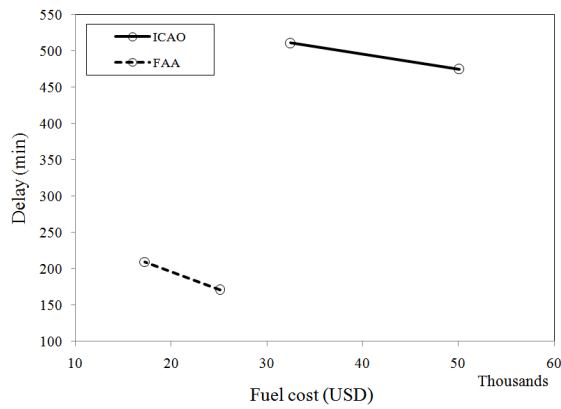

Figure B.1: Trade-offs between Total Fuel Cost and Total Delay Cost Functions

APPENDIX C

RESULTS ON BENCHMARK INSTANCES

We report here results for benchmark instances used in Beasley et al. (2000) with the objective of minimizing the total weighted deviations from target times. Columns 4-6 provide results as reported in Beasley et al. (2000). Columns 7-8 present the results we obtained using these instances with the model in Beasley et al. (2000) and CPLEX 12.4. The last two columns report the results obtained using an adaptation of MRASP to reflect the constraints and the objective function in Beasley et al. (2000). It is worth noting that our model does not involve the auxiliary binary variables introduced in Beasley et al. (2000) that indicate whether or not a pair of aircraft is assigned to the same runway.

				Beasley et. al (2000)		Beasley Model		MRASP	
					Tree		B&B		B&B
Instance	n	m	Optimal	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes
1	10	1	700	0.4	49	0.11	44	0.28	31
		2	90	0.6	91	0.11	11	0.13	27
		3	0	-	-	0.10	0	0.10	0
2	15	1	1480	5.2	454	0.14	361	0.12	580
		2	210	1.8	115	0.27	77	0.25	4
		3	0	-	-	0.11	0	0.11	0
3	20	1	820	2.7	42	0.13	83	0.12	127
		2	60	3.8	142	0.13	57	0.28	56
		3	0	-	-	0.13	0	0.13	0
4	20	1	2520	220.4	20002	1.10	11926	1.45	15015
		2	640	1919.9	193319	5.16	101737	1.21	8729
		3	130	2299.2	39901	0.35	2578	0.67	1927
		4	0	-	-	0.14	0	0.14	0
5	20	1	3100	922	50745	2.45	33617	4.14	41737
		2	650	11510.4	282160	2.40	27288	1.02	7538
		3	170	1655.3	20035	0.46	2841	0.35	1271
		4	0	-	-	0.14	0	0.15	0
6	30	1	24442	33.1	10806	0.08	0	0.08	0
		2	554	1568.1	25316	0.21	361	0.28	84
		3	0	-	-	0.12	0	0.11	0
7	44	1	1550	10.6	2192	0.17	1512	0.14	1326
		2	0	-	-	0.12	0	0.10	0
8	50	1	1950	111.9	1114	0.35	71	0.50	142
		2	135	3450.6	9020	0.64	520	1.69	1742
		3	0	-	-	0.41	0	0.35	0

Table C.1: Benchmark Instances with Weighted Deviations from Target Times

BIBLIOGRAPHY

Abela, J., D. Abramson, M. Krishnamoorthy, A. De Silva, G. Mills. 1993. Computing optimal schedules for landing aircraft. In: Proceedings of the 12th National ASOR Conference 71-90.

Anagnostakis, I., J. P. Clarke. 2002. Runway operations planning: a two-stage heuristic algorithm. In: AIAA aircraft technology integration and operations forum Los Angeles CA USA.

Atkin, J. A. D., E. K. Burke, J. S. Greenwood, D. Reeson. 2004. A metaheuristic approach to aircraft departure scheduling at London heathrow airport. In: Electronic proceedings of the 9th international conference on computer-aided scheduling of public transport San Diego CA USA.

Atkin, J. A. D., E. K. Burke, J. S. Greenwood, D. Reeson. 2007. Hybrid metaheuristics to aid runway scheduling at London heathrow airport. Transportation Science 41 90106.

Atkin, J. A. D., E. K. Burke, J. S. Greenwood, D. Reeson. 2008. A meta-heuristic approach to departure scheduling at London Heathrow airport. Computer Aided Systems of Public Transport 235-252.

Barnhart, C., E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, P.H. Vance. 1998. Branch-and-price: column generation for huge integer programs. Operations Research **46** 316-329.

Beasley, J. E., J. Sonander, P. Havelock. 2001. Scheduling aircraft landings at London Heathrow using a population heuristic. Journal of the Operational Research Society **52** 483-493.

Beasley, J. E., M. Krishnamoorthy, Y. M. Sharaiha, D. Abramson. 2000. Scheduling aircraft landingsthe static case. Transportation science **34**(2) 180-197.

Bencheikh, G., J. Boukachour, A. E. H. Alaoui, F. E. Khoukhi. 2009. Hybrid method for aircraft landing scheduling based on a job shop formulation. International Journal of Computer Science and Network Security **9** 7888.

Bennell, J. A., M. Mesgarpour, C. N. Potts. 2011. Airport runway scheduling. 4OR: A quarterly Journal of Operations Research **9**(2) 115-138.

Bertsimas, D., M. Frankovich, A. Odoni. 2011. Optimal Selection of Airport Runway Configurations. Operations research **59**(6) 1407-1419.

Bianco, L., P. Dell'Olmo, S. Giordani. 2006. Scheduling models for air traffic control in terminal areas. Journal of Scheduling **9** 223-253.

Boysen, N., M. Fliedner. 2011. Scheduling aircraft landings to balance workload of ground staff. Computers & Industrial Engineering **60**(2) 206-217.

Brentnall, A. R. 2006. Aircraft arrival management. PhD thesis University of Southampton UK.

Brinton, C. R. 1992. An implicit enumeration algorithm for arrival aircraft scheduling. In: Proceedings of the IEEE/AIAA 11th digital avionics systems conference Seattle WA USA.

Capri, S., M. Ignaccolo. 2004. Genetic algorithms for solving the aircraft-sequencing problem: the introduction of departures into the dynamic model. Journal of Air Transport Management **10** 345351.

Chen, Z-L., W. B. Powell. 1999. Solving parallel machine scheduling problems by column generation, INFORMS Journal on Computing **11**(1) 78-94.

Ciesielski, V., P. Scerri. 1997. An anytime algorithm for scheduling of aircraft landing times using genetic algorithms. Australian Journal of Intelligent Information Processing Systems **4** 206213.

Clautiaux, F., C. Alves, J. Valério de Carvalho, J. Rietz. 2011. New stabilization procedures for the cutting stock problem. INFORMS Journal on Computing **23**(4) 530-545.

Cook, A., G. Tanner, S. Anderson. 2004. Evaluating the true cost to airlines of one minute of airborne or ground delay. EUROCONTROL. URL: http://www.eurocontrol.int.

de Carvalho, J.M.V. 2005. Using extra dual cuts to accelerate column generation. INFORMS Journal on Computing 17(2) 175-182.

de Neufville, R., A. R. Odoni. 2003. Airport systems: planning, design, and management. McGraw-Hill New York.

Dear, R. G., 1976. The dynamic scheduling of aircraft in the near terminal area. Flight Transportation Laboratory, Massachusetts Institute of Technology, Technical report R76-9.

Dear, R. G., Y. S. Sherif. 1991. An algorithm for computer assisted sequencing and scheduling of terminal area operation. Transport Research Part A **25** 129139.

Dell'Olmo, P., G. Lulli. 2003. A dynamic programming approach for the airport capacity allocation problem. IMA Journal of Management Mathematics **14**(3) 235-249.

Desrochers, M. 1986. An algorithm for the shortest path problem with resource constraints. Montreal : Universite de Montreal, Centre de recherche sur les transports.

Desrochers, M., J. Desrosiers, M. Solomon. 1992. A new optimization algorithm for the vehicle routing problem with time windows. Operations research **40**(2) 342-354.

Desrochers, M., J. K. Lenstra, M. W. Savelsbergh, F. Soumis. 1988. Vehicle routing with time windows: Optimization and approximation. Vehicle routing: Methods and studies 16 65-84.

du Merle, O., D. Villeneuve, J. Desrosiers, P. Hansen. 1999. Stabilized column generation. Discrete Mathematics **194** 229-237.

Ernst A. T., M. Krishnamoorthy, R. H. Storer. 1999. Heuristic and exact algorithms for scheduling aircraft landings. Networks **34** 229-241.

Feillet, D., P. Dejax, M. Gendreau, C. Gueguen. 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. Networks **44**(3) 216-229.

Fu, L., D. Sun, L. R. Rilett. 2006. Heuristic shortest path algorithms for transportation applications: state of the art. Computers & Operations Research **33**(11) 3324-3343.

Ghoniem, A., H. D. Sherali, H. Baik. 2012. Enhanced models for a mixed arrival-departure aircraft sequencing problem. *Manuscript*, Isenberg School of Management, University of Massachusetts Amherst.

Ghoniem, A., H. D. Sherali. 2009. Complementary column generation and bounding approaches for set partitioning formulations. Optimization Letters **3**(1) 123-136.

Ghoniem, A., H. D. Sherali. 2011. Set partitioning and packing versus assignment formulations for subassembly matching problems. Journal of the Operational Research Society **62** 2023-2033.

Gilbo, E. P., 1993. Airport capacity: Representation, estimation, optimization. Control Systems Technology IEEE Transactions on **1**(3) 144-154.

Gondzio, J., P. González-Brevis, P. Munari. 2013. New developments in the primal-dual column generation technique. European Journal of Operational Research 224(1) 41-51.

Heblij, S. J., R. A. Wijnen. 2008. Development of a runway allocation optimisation model for airport strategic planning. Transportation Planning and Technology **31**(2) 201-214.

Hu, X. B., W. H. Chen. 2005. Receding horizon control for aircraft arrival sequencing and scheduling. IEEE Trans Intell Transport Syst **6** 189197

Idris, H. R. 2001. Observation and analysis of departure operations at Boston Logan international airport. PhD thesis Department of Mechanical Engineering MIT USA.

Idris, H. R., B. Delcaire, I. Anagnostakis, W. D. Hall, J. P. Clarke, R. J. Hansman, E. Feron, A. R. Odoni. 1998. Observations of departure processes at logan airport to support the development of departure planning tools. The 2nd USA/Europe Air Traffic Management R&D Seminar Orlando.

Idris, H. R., I. Anagnostakis, B. Delcaire, R. J. Hansman, J. P. Clarke, E. Feron, A. R. Odoni. 1999. Observations of departure processes at Logan airport to support the development of departure planning tools. Air Traffic Control **7** 229-257.

Lübbecke, M. E., J. Desrosiers. 2005. Selected topics in column generation. Operations Research **53**(6) 1007-1023.

Laporte, G. 1992. The vehicle routing problem: An overview of exact and approximate algorithms. European Journal of Operational Research **59**(3) 345-358.

Lee, H., 2008. Trade-off evaluation of scheduling algorithms for terminal-area air traffic control. Doctoral dissertation, Massachusetts Institute of Technology.

Lee, H., H. Balakrishnan. 2008. Fuel cost, delay and throughput tradeoffs in runway scheduling. In Proceedings of American control conference IEEE 2449-2454.

Luenberger, R. A. 1998. A traveling-salesman-based approach to aircraft scheduling in the terminal area. Technical report NASA Technical Memorandum 100062.

Neame, P.J. 1999. Nonsmooth dual methods in integer programming Ph.D. Thesis, University of Melbourne, Victoria, Australia.

Newell, G. F., 1979. Airport capacity and delays. Transportation Science **13**(3) 201-241.

Ostrowski, J., M. F. Anjos, A. Vannelli. 2010. Symmetry in scheduling problems. URL: http://www.optimization-online.org/DB_FILE/2010/11/2824.pdf.

Pinol, H., J. E. Beasley. 2006. Scatter search and bionomic algorithms for the aircraft landing problem. European Journal of Operational Research **171** 439462.

Psaraftis, H. N. 1978. A dynamic programming approach to the aircraft sequencing problem. Technical report, R78-4, Flight Transportation Laboratory, MIT, USA.

Pujet, N., B. Delcaire, E. Feron. 1999. Input-output modeling and control of the departure process of congested airports. Proceedings of AIAA guidance navigation and control conference and exhibit Portland OR USA.

Randall, M. C. 2002. Scheduling aircraft landings using ant colony optimisation. In: Proceedings of The IASTED international conference artificial intelligence and soft computing, Banff, Canada.

Rousseau, L-M., M. Gendreau, D. Feillet. 2007. Interior point stabilization for column generation. Operations Research Letters **35** 660-668.

Sölveling, G., S. Solak, J. P. B. Clarke, E. L. Johnson. 2011. Scheduling of runway operations for reduced environmental impact. Transportation Research Part D: Transport and Environment **16**(2) 110-120.

Sherali, H. D., A. Ghoniem. 2009. Joint vehicle assembly-routing problems: An integrated modeling and optimization approach, Networks **53**(3) 249-265.

Sherali, H. D., J. C. Smith. 2001. Improving discrete model representations via symmetry considerations. Management Science **47** 1396-1407.

Sherali, H. D., W. P. Adams. 1990. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. SIAM Journal of Discrete Mathematics **3** 411-430.

Sherali, H. D., W. P. Adams. 1999. A reformulation-linearization technique for solving discrete and continuous nonconvex problems. Kluwer Academic Publishers, Dordrecht.

Solomon, M. M., J. Desrosiers. 1988. Survey Paper-Time Window Constrained Routing and Scheduling Problems. Transportation science **22**(1) 1-13.

Stevens, G. 1995. An approach to scheduling aircraft landing times using genetic algorithms. Honours thesis, Department of Computer Science, RMIT University, Melbourne, Australia.

Subramanian, S., H. D. Sherali. 2008. An effective deflected subgradient optimization scheme for implementing column generation for large-scale airline crew scheduling problems, INFORMS Journal on Computing **20**(4) 565-578.

Venkatakrishnan, C.S., A. Barnett, A. R. Odoni. 1993. Landings at Logan Airport: Describing and increasing airport capacity. Transportation Science **27**(3) 211-227.

Wells, A. T. 1992. Airport planning & management.

Wen, M., J. Larsen, J. Clausen. 2005. An exact algorithm for aircraft landing problem. IMM-Technical Report-2005-12 Informatics and Mathematical Modeling Technical University of Denmark.